



# Big Data Technologies for Near-Real-Time Results:

## Balanced Infrastructure that Reduced Workload Completion Time from Four Hours to Seven Minutes<sup>1</sup>



Improvements in widely available compute, storage, and network components are rapidly improving the ability of commercial, academic, and government organizations to handle big data effectively. Intel has demonstrated dramatic results from optimized, balanced Apache Hadoop\* clusters that include the latest Intel® Xeon® processors, solid-state local storage, and 10 Gigabit Intel® Ethernet Converged Network Adapters.

In fact, in tests conducted by Intel, upgrading these components and using the Intel® Distribution for Apache Hadoop\* software (Intel® Distribution) reduced the time required to sort a terabyte of data using the TeraSort benchmark workload from approximately four hours to approximately seven minutes.<sup>1,2</sup> These results represent significant progress toward near-real-time data analytics at significantly lower cost than was previously possible with proprietary hardware and software implementations. The dramatic cost savings and improved efficiency are vital aspects of enabling the full potential of big data technologies.

This paper demonstrates how those results were achieved, as guidance for IT decision makers and others as they consider where to invest for optimal results from Hadoop\* environments. It also introduces some of the benefits available from the pre-optimized, turnkey Intel Distribution for Apache Hadoop software. Using the guidelines presented here, organizations can work toward optimal performance within their budgetary requirements for specific workloads and circumstances.

Balancing the compute, storage, and network resources in an Apache Hadoop\* cluster enabled the full benefit of the latest Intel® processors, solid-state drives, 10 Gigabit Intel® Ethernet Converged Network Adapters, and the Intel® Distribution for Apache Hadoop\* software. Building a balanced infrastructure from these components enabled reducing the time required for Intel to complete a TeraSort benchmark test workload from approximately four hours to approximately seven minutes, a reduction of roughly 97 percent.<sup>1</sup> Results such as this from big data technologies pave the way for low-cost, near-real-time data analytics that will help businesses respond almost instantly to changing market conditions and unlock more value from their assets.

## Table of Contents

<b>1 Evolving Tools to Manage Big Data</b> .....	<b>2</b>
1.1 Introducing Hadoop*: A Robust Framework for Generating Value from Big Data .....	2
1.2 The Industry Ecosystem of Big Data Technologies .....	2
<b>2 Establishing Balance Among Critical Components</b> .....	<b>3</b>
2.1 Advancements in Compute Resources .....	3
2.2 Advancements in Storage Technology .....	4
2.3 Advancements in Networking .....	4
2.4 An Optimized Hadoop Distribution from Intel .....	4
<b>3 The Role of 10GbE and Other Factors in Accelerating Hadoop Workflows</b> .....	<b>5</b>
3.1 Optimizing for the Import Stage .....	6
3.2 Optimizing for the Processing Stage: The Path from Four Hours to Seven Minutes .....	6
3.3 Optimizing for the Export Stage .....	8
<b>4 Description of the Research Environment</b> .....	<b>8</b>
<b>5 Tuning and Optimization Considerations</b> .....	<b>9</b>
5.1 Networking, OS, and Driver Optimizations .....	9
5.2 Hadoop Configuration Parameters .....	9
5.3 Future Enhancements .....	10
<b>6 Conclusion</b> .....	<b>11</b>

## 1 Evolving Tools to Manage Big Data

Enormous data stores have become a fact of life for organizations of all types and sizes. The ability to manipulate, transform, and drive benefit from that data—which is often **unstructured big data**—is quickly becoming the norm, and the tools and techniques for doing so are becoming more commonplace. Frameworks such as Hadoop are widely used, and IT organizations are increasingly building their own computing environments for handling big data.

### 1.1 Introducing Hadoop\*: A Robust Framework for Generating Value from Big Data

Hadoop is an open-source software framework written in Java\* and based on Google’s MapReduce\* and distributed file system work. It is built to support distributed applications by analyzing very large bodies of data using clusters of servers, transforming it into a form that is more usable by those applications. Hadoop is designed to be deployed on commonly available, general-purpose infrastructure.

Tasks that the framework is particularly well suited for include indexing and sorting large data sets, data mining, log analytics, and image manipulation. Key parts of the Hadoop framework include the following:

- **Hadoop Distributed File System (HDFS\*)** achieves fault tolerance and high performance by breaking data into blocks and spreading them across large numbers of worker nodes.
- **Hadoop’s MapReduce Engine** accepts jobs from applications and divides those jobs into tasks that it assigns to various worker nodes.

### 1.2 The Industry Ecosystem of Big Data Technologies

A large ecosystem of solutions—of which Hadoop is just one part—has been designed to derive maximum value from big data. Another key component is NoSQL (“Not only SQL”) databases, which exist as an alternative (or complement) to the more common, table-based relational database management systems (RDBMSs). Unlike RDBMSs, NoSQL databases are not based primarily on tables. That characteristic makes them somewhat less efficient than RDBMSs for functionality that depends on the relationships between data elements, but more streamlined for handling large amounts of data where those relationships are not centrally important.

While structured data can be stored in NoSQL databases, these systems are especially well suited for handling unstructured data and in particular to providing high scalability and performance for retrieving and appending that data in large quantities. As big data technologies have grown to hold a more prominent role in the data center, open source solutions such as Hadoop have become increasingly important and well-developed, through work by a combination of commercial and non-commercial entities. High-profile NoSQL databases in the Hadoop ecosystem include **Apache Cassandra\***, **HBase\***, and **MongoDB\***.

In addition to Hadoop, other examples of big data technologies range from the simple-to-use, open source **Disco Project\***, for which developers write jobs using Python\* scripting, to the **SAP HANA\* real-time data platform**, an enterprise-scale environment focused on business intelligence and related usages. All of these industry innovations are being developed and optimized for Intel Xeon processor-based platforms. This paper focuses on Hadoop as an example of building systems for processing big data because of its widespread and fast-growing presence in commercial, research, and academic environments.

## 2 Establishing Balance Among Critical Components

While Hadoop clusters are typically built from generally available, mainstream components, that fact doesn't diminish the challenges associated with choosing and matching those components for maximum benefit. The primary consideration is to create a balance in the environment between compute, storage, and network resources, as depicted in Figure 1.

Before turning to specific strategies for deciding on combinations of components for your cluster, it is useful to consider the state of generally available technology in each category, as shown in Table 1. After establishing which types of resources are desirable, the discussion will focus on how a Hadoop cluster takes advantage of those resources, before describing the role of 10 Gigabit Ethernet (10GbE) networking in delivering the benefits of each.

### 2.1 Advancements in Compute Resources

Platform architecture introduced with the Intel® Xeon® processor E5 family makes better use of resources throughout the solution stack, compared to previous-generation platforms. For example, increased core count, from six cores (12 hardware threads) to eight cores (16 hardware threads) per socket, enhances the ability to handle a higher degree of parallelism, which is particularly beneficial for data-intensive Hadoop workloads. Intel® Data Direct I/O Technology (Intel® DDIO) is a new feature in the Intel Xeon processor E5 family that allows Intel® Ethernet Controllers and adapters to talk directly with the processor cache instead of the main memory, helping deliver increased bandwidth and lower latency that are particularly beneficial when processing large data sets.

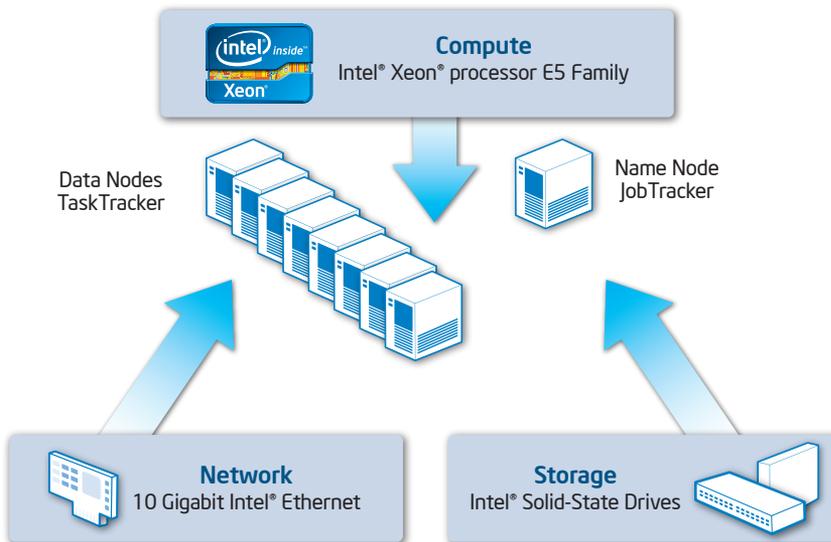


Figure 1. Balance among compute, network, and storage resources for optimal Apache Hadoop\* results.

Table 1. Upgrades across the Hadoop\* solution stack for a high-performance balanced infrastructure.

	COMPUTE	STORAGE	NETWORKING
<b>Good</b>	Earlier-generation Intel® Xeon® processors	Conventional spinning hard drives	Intel® Ethernet Gigabit Server Adapters
<b>Better</b>	Intel® Xeon® processor E5 family	Tiered storage (conventional plus solid-state drives)	10 Gigabit Intel® Ethernet Converged Network Adapters
<b>Best</b>	Intel Xeon processor E5 family	All solid-state drives	10 Gigabit Intel Ethernet Converged Network Adapters

**2.2 Advancements in Storage Technology**

Solid-state drives (SSDs) represent a major shift in persistent storage for mainstream client and server computers. Eliminating the electro-mechanical parts of conventional hard disk drives (HDDs), such as spinning disks and read/write heads, is a key factor in providing dramatically improved data-access time and reduced latency.

The Intel® Solid-State Drive (Intel® SSD) 520 Series used in the testing described in this paper is available in a wide range of capacities, offering built-in data-protection features and exceptional performance improvements over conventional HDDs.<sup>3</sup> Built with compute-quality Intel 25-nanometer NAND Flash Memory, the Intel SSD 520 Series offers random read performance of up to 50,000 input/output operations per second (IOPS)<sup>4</sup> and sequential read performance of up to 550 megabytes per second (MB/s).<sup>5</sup>

As a transitional step between conventional hard drives and SSDs, it has become increasingly common for organizations to provision servers with both types of drives on the same machine. In this scenario, SSDs function as high-speed data cache devices, reducing the need for reads from and writes to the conventional HDDs, improving overall performance.

**2.3 Advancements in Networking**

Regular advances in the wire speeds of networking components have been ongoing for many years, together with complementary technologies that add value such as increasing throughput, improving cost-effectiveness, and enhancing flexibility. The main consideration in the present study is the transition from Gigabit Ethernet (1GbE) to 10GbE.

Intel® Ethernet Controllers and Converged Network Adapters have been instrumental in driving down the cost of 10GbE networking. In turn, increasing deployment of virtualization and bandwidth-hungry applications such as data analytics and video on demand has led to more widespread adoption of 10GbE, establishing a virtuous cycle where cost benefits and broadening mainstream adoption continue to support each other. Intel Ethernet software drivers have been optimized for big data implementations; for example, they are engineered to minimize I/O interference to Hadoop data processing.

The Intel® Ethernet Converged Network Adapter X540 is a low-cost, low-power 10GBASE-T solution that provides backward compatibility with existing 1000BASE-T networks using Category 6 and Category 6A copper cabling. The Intel® Ethernet Controller X540 lowers both initial cost and power requirements

by integrating the MAC and PHY into a single-chip solution. The Intel® Ethernet Converged Network Adapter X520 offers SFP+ connectivity for 10GbE using either copper or fiber-optic networking.

**2.4 An Optimized Hadoop Distribution from Intel**

The Intel Distribution helps streamline and improve the implementation of Hadoop on Intel® architecture-based infrastructure. It is the only distribution built from the silicon up to enable the widest range of data analysis on Hadoop, and it is the first with hardware-enhanced performance and security capabilities. The solution includes the Hadoop framework, MapReduce, Hadoop Distributed File System (HDFS), and other related components to support both batch processing and near-real-time analytics, including the Hive\* data warehouse infrastructure, Pig\* data flow language, and HBase database. The components included in the Intel Distribution are shown in Figure 2.

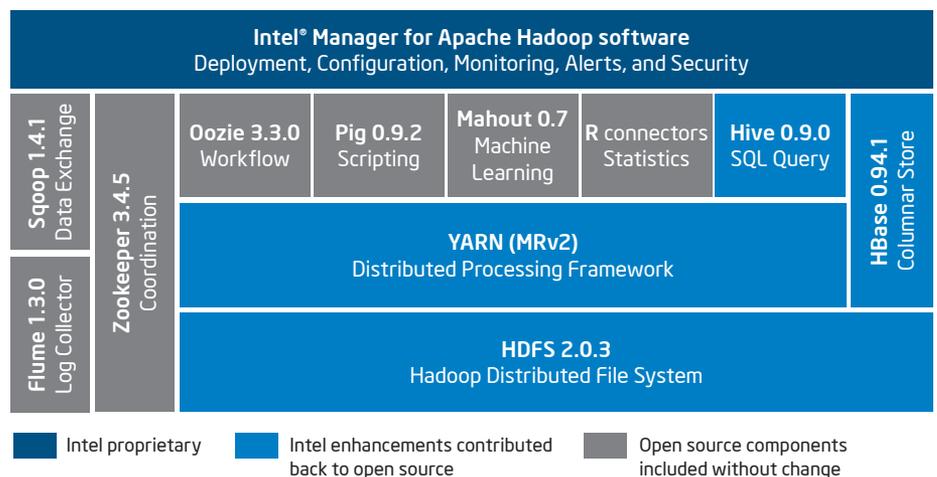


Figure 2. Components in the Intel® Distribution for Apache Hadoop\* Software.

The Intel Distribution provides tuning and optimization guidelines based on real-world experience, as well as wizards and other automated deployment tools. The Intel® Manager for Apache Hadoop\* software provides automated installation on Hadoop cluster nodes, as well as real-time management, monitoring, and diagnostic capabilities using powerful, intuitive dashboards, as shown in Figure 3. The Intel Distribution also offers customers extensive training resources, as well as assistance with system design, deployment, customization, and tuning. Enterprise support is also available on a 24/7 basis to meet the needs of organizations and people whose success depends on high degrees of cluster uptime.

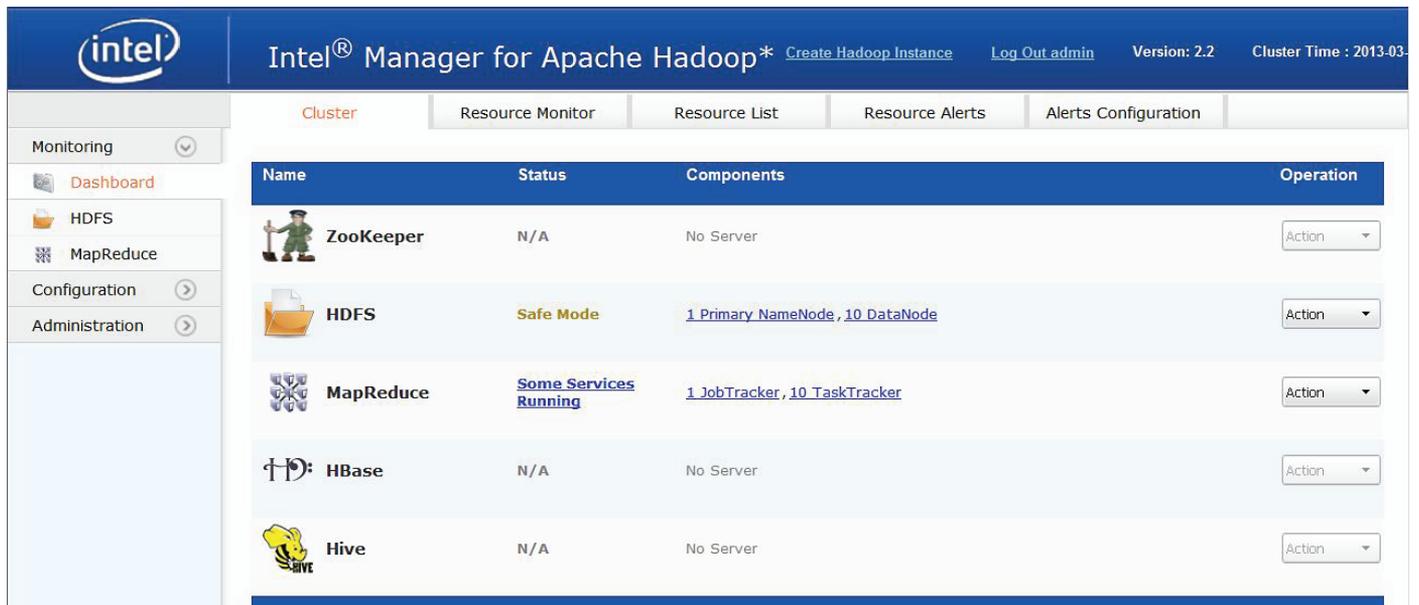


Figure 3. Dashboards to simplify setup and configuration, reducing deployment time.

### 3 The Role of 10GbE and Other Factors in Accelerating Hadoop Workflows

Improvements in the ability to manage big data continue to bring us closer to achieving real-time analytics in mainstream environments. Using common data center hardware and software, we are improving on solutions that take hours or even days to derive value out of source data.

Improving results in this context requires consideration of the entire workflow, including the point where data enters the system, the mechanisms for processing it, and the tasks associated with exporting the processed data back out of the system. Accordingly, the Hadoop workflow can be considered as consisting of three stages:

1. **Import.** The first step of using Hadoop to derive an answer from a large data set is to get the data from an application into HDFS. Data can be imported in either a streamed or a batch modality.
2. **Processing.** Once the data has been imported into HDFS, Hadoop manipulates that data to extract value from it. The MapReduce engine accepts jobs from applications through its JobTracker node, which divides the work into smaller tasks that it assigns to TaskTracker nodes. Typical operations include sorting, searching, or analytics. TeraSort is a standard Hadoop benchmark based on a data-sorting workload.<sup>1</sup> In our test environment, we use TeraGen to actually generate the data set.
3. **Export.** After the operations have been completed on the data in the processing stage, the results are made available to applications.

This model demonstrates the value of 10GbE throughout the Hadoop workflow, although the relative demands of these three stages compared to one another vary significantly according to the nature of the work being performed. To give a simple example, a workload that includes large-scale data compression might be expected to have a larger workload burden for importing data than for exporting it back out in its compressed form. Similarly, some tasks are more computationally intensive than others, even though the amounts of data they work on could be similar. The tuning and optimization guidelines and support services associated with the Intel Distribution can be beneficial in identifying the best approach for specific needs.

### 3.1 Optimizing for the Import Stage

The import stage within this model consists simply of putting data into HDFS for processing. That process will always occur at least once, and in some cases—particularly where MapReduce is sold as a service—many imports could be necessary. This stage and the Hadoop replication factor place intense network-performance demands on the system, in terms of both networking and storage I/O.

10GbE networking is instrumental in supporting these demands as data is imported into the system. Migrating from 1GbE to 10GbE produces up to a 4x performance improvement in import operations using conventional HDDs with parallel writes and up to 6x improvement using SSDs.<sup>1</sup> The greater improvement with SSDs can be attributed to faster writes into the storage subsystem using non-volatile memory.

### 3.2 Optimizing for the Processing Stage: The Path from Four Hours to Seven Minutes

Intel testing used a 1 terabyte (TB) TeraSort workload distributed over 10 data nodes with one name node. To gauge the benefit of upgrading various resources, results were gathered before and after migrating the cluster from the Intel® Xeon® processor 5600 series to the Intel® Xeon® processor E5-2600 product family, followed by an upgrade from conventional HDDs to SSDs, and finally from 1GbE to 10GbE.

These hardware upgrades resulted in a reduction in processing time from approximately four hours to about 12 minutes. Implementing the Intel

Distribution reduced that processing time further to approximately seven minutes. Altogether, this performance increase reflects nearly a 97-percent reduction in the time required for the processing stage.

The first hardware change tested was an upgrade from the Intel® Xeon® processor X5690 to the Intel® Xeon® processor E5-2690. As illustrated in Figure 4, the processor upgrade reduced the time required to sort the 1 TB data set approximately in half, from 250 minutes to 125 minutes.<sup>6</sup>

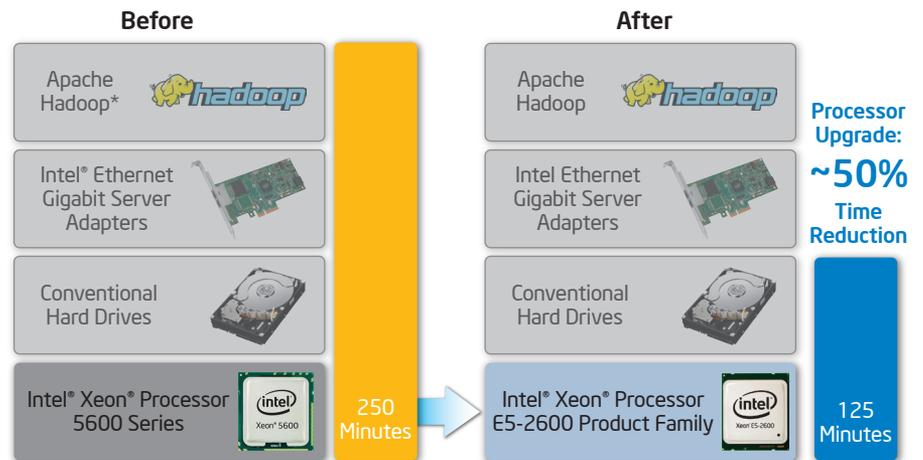


Figure 4. Processor upgrade to Intel® Xeon® processor E5-2600 product family: processing-stage speed improvement of approximately 50 percent.<sup>1</sup>

Working with this massive data set, the ability to access non-sequential data quickly is a key performance consideration. Therefore, to reduce any existing storage bottleneck, the next upgrade tested was to replace the conventional HDDs with SSDs, taking advantage of their dramatically higher random read times. Building on the previous performance improvement from the processor upgrade, shifting from conventional HDDs to the Intel SSD 520 Series reduced the time to complete the workload by approximately another 80 percent—from about 125 minutes to about 23 minutes, as shown in Figure 5.<sup>7</sup>

For those customers interested in combining conventional HDDs and SSDs in the same server, Intel offers Intel® Cache Acceleration Software. This tiered storage model provides some of the performance benefits of SSDs at a lower acquisition cost, but in addition to the performance differences compared to an SSD-only configuration, this approach also sacrifices some of the reliability benefits available from SSDs. Nevertheless, this storage model offers customers another option as they move toward full adoption of SSDs to dramatically reduce the time to get from data to insight.

Testing has also revealed that when five SSDs are installed per task node, the Hadoop framework is able to run enough simultaneous Map tasks to generate parallel I/O to each SSD and utilize the processors at almost 100 percent.<sup>8</sup> This state allows for highly optimized performance of Map tasks. Setting `io.sort.mb` and `io.sort.record.percent` flags appropriately avoided intermediate Map output spills and excessive disk reads and writes. Each Map task processing a 128 MB block of data is completed in less than 10 seconds and generates 128 MB of output. Running 32 Map tasks in parallel enables each task node to generate more than 5 Gb/s.<sup>8</sup>

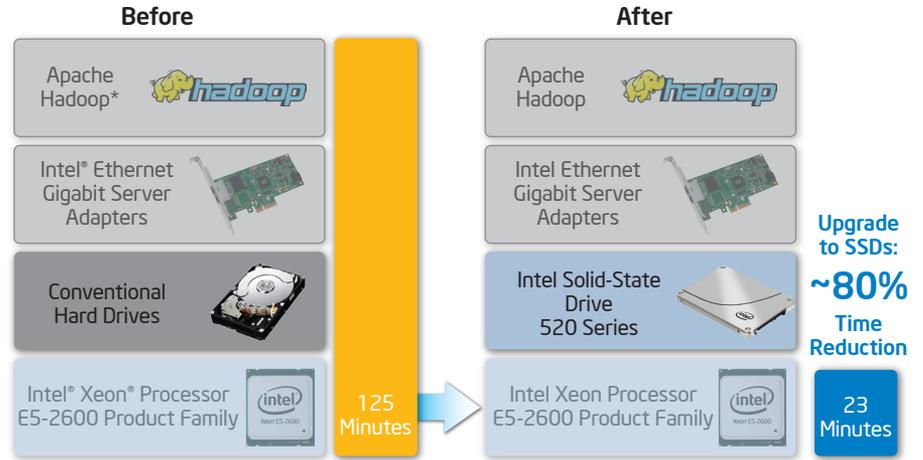


Figure 5. Storage upgrade to solid-state drives: processing-stage speed improvement of approximately 80 percent.<sup>7</sup>

The large scale and distributed nature of Hadoop workloads makes network I/O a vital aspect of overall performance at every stage in the workflow, and 10GbE is a cost-effective, scalable solution that helps reduce wait times for data. Having a high bandwidth 10GbE network not only allows for data to be imported and exported from the cluster quickly, but it also improves the shuffling phase of the TeraSort workload to be accelerated. Using 10GbE links between Map and Reduce nodes allows the Reduce nodes to fetch data rapidly, helping improve overall job-execution time and cluster performance.

Upgrading the cluster hardware from 1GbE to 10GbE to build on the upgrades of the processor and storage reduced processing time on the test workload by up to an additional 50 percent, from about 23 minutes to about 12 minutes, as shown in Figure 6.<sup>9</sup> Using 10GbE interconnects and SSDs supported running more than 100 concurrent reducer tasks on the 10-node test cluster, exhibiting good job scaling and high resource utilization.<sup>5</sup>

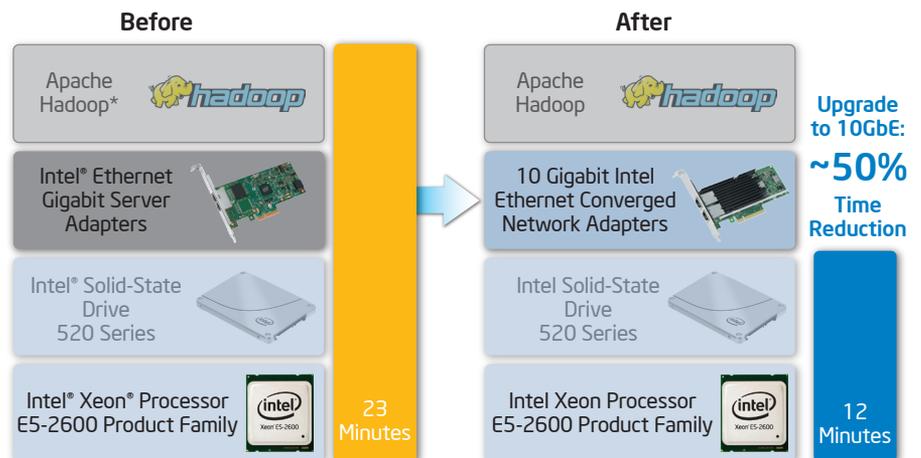


Figure 6. Networking upgrade to 10 Gigabit Ethernet: processing-stage speed improvement of approximately 50 percent.<sup>9</sup>

In addition to the hardware upgrades already described, the Intel Distribution provides an intuitive interface and many under-the-hood optimizations, including advanced data compression, dynamic replica selection for HDFS, and MapReduce speedup. This additional engineering helps deliver performance gains and, together with trusted, enterprise-grade support from Intel, helps customers more rapidly deploy and successfully maintain a Hadoop environment. Implementing the Intel Distribution on top of the hardware upgrades already made reduced workload-completion time requirements by up to approximately another 40 percent—from about 12 minutes to about 7 minutes, as shown in Figure 7.<sup>10</sup>

### 3.3 Optimizing for the Export Stage

Similar to the import stage, 10GbE dramatically benefits performance of extracting data from the system after processing, particularly in conjunction with upgrading from conventional HDDs to SSDs. Initial testing with conventional HDDs revealed a significant bottleneck in the export stage of the Hadoop workflow due to random disk seeks. Replacing the local storage with SSDs eliminated this shortcoming, enabling results in keeping with those we saw in the

import stage. Using SSDs, we again saw up to approximately a 6x performance improvement from 10GbE relative to 1GbE, dramatically improving the time requirements for the overall operation across the workflow. As described above, using SSDs and 10GbE also enhances the benefit from high levels of processor resources, emphasizing the benefit of balancing resources in the Hadoop cluster.

### 4 Description of the Research Environment

The Hadoop test bed used to generate the results reported in this paper included one head node (name node, job tracker), 10 workers (data nodes, task trackers), and a Cisco Nexus\* 5020 10 Gigabit switch. The various baseline and upgraded worker-node components that are compared in the testing are detailed in Table 2.<sup>2</sup>

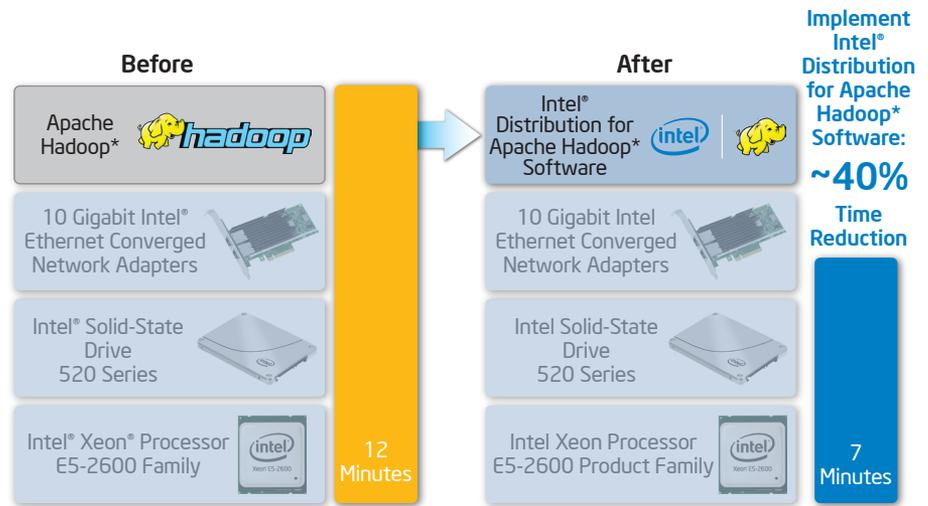


Figure 7. Implementation of Intel® Distribution for Apache Hadoop\* Software: processing-stage speed improvement of approximately 40 percent.<sup>10</sup>

Table 2. Worker-node components compared in the test environment.

	PROCESSOR AND BASE SYSTEM COMPARISON	STORAGE COMPARISON	NETWORK ADAPTER COMPARISON	SOFTWARE COMPARISON
<b>Baseline Components</b>	SuperMicro SYS-1026T-URF 1U servers with two Intel® Xeon® processors X5690 @ 3.47 GHz, 48 GB RAM	700 GB 7200 RPM SATA hard drives	Intel® Ethernet Server Adapter I350-T2 (Gigabit Ethernet)	Apache Hadoop* 1.0.3
<b>Upgraded Components</b>	Dell PowerEdge* R720 2U servers with two Intel® Xeon® processors E5-2690 @ 2.90 GHz, 128 GB RAM	Intel® Solid-State Drive 520 Series	Intel® Ethernet Converged Network Adapter X520-DA2 (10 Gigabit Ethernet)	Intel® Distribution for Apache Hadoop* software 2.1.1

## 5 Tuning and Optimization Considerations

In addition to upgrading network components and considering the use of the Intel Distribution, organizations working to maximize the value they get from big data technologies must consider configurations and settings in the networking stack and the Hadoop software environment itself. While the potential scope of configurations and settings to be considered is daunting, best practices suggest that engineers should pay particular attention to the considerations listed in this section for optimal value.

### 5.1 Networking, OS, and Driver Optimizations

In the OS and networking stack, the number of open files, open network connections, and running processes at any time should be adjusted according to the needs of the specific workload. The Intel 10GbE Linux\* drivers should also be optimized by adjusting the number of RSS queues (two was the optimal number in the testing described in this paper), and the number of context switches should be reduced by adjusting interrupt throttling. In the OS and the networking and TCP/IP stack, the following settings, optimizations, and practices are of particular note:

- **Increase the limit of open files in the OS.** Hadoop inherently opens large numbers of files, so increasing the limit of concurrent open files reduces job failures; Intel testing found 32K to be sufficient. Increasing the limit of concurrent processes also helps reduce job failures.

- **Increase the number of outstanding connections and outstanding SYN requests.** Hadoop's HDFS and MapReduce engine open large numbers of short-lived TCP/IP connections. In Intel setup configurations, this setting is increased to 3,240,000, reducing the wait time for HDFS and MapReduce communications.
- **If resource sharing beyond the Hadoop workload is not required, consider increasing TCP/IP maximum window size and scaling up to 16 MB.** Where possible, this approach helps to maximize the value of the 10GbE investment.
- **If enough system memory is available, increase the TCP/IP send and receive buffer sizes.** This change may improve network throughput. In the Intel test cluster, the maximum values were set to 16 MB.
- **Disable TCP/IP selective ACK when bandwidth is readily available.** When selective ACKs are enabled, the response to client requests can be delayed, degrading performance in terms of job execution and completion time; disabling them helps improve overall server response time and Hadoop job performance.
- **Use JBOD for storage.** Hadoop has built-in load balancing and uses efficient round-robin among the available HDFS and MapReduce JBOD Disks. Using RAID with SSDs and fast storage limits storage throughput and overall job performance. Instead, use *Disks in JBOD* mode for HDFS and MapReduce, as both include built-in functionality for load balancing across multiple JBOD disks.

### 5.2 Hadoop Configuration Parameters

Within the more than 200 configuration parameters available within the Hadoop stack, starting with the following subset of tuning considerations can help engineering teams apply their efforts efficiently:

- **Memory configurations for Java virtual machine (JVM) tasks.** Each Map and Reduce task runs in an independent JVM instance. One can specify the amount of memory each Map and Reduce task can allocate, using the configuration parameters *mapred.map.child.java.opts* and *mapred.reduce.child.java.opts* for Map and Reduce tasks, respectively. The Intel test cluster set the Map task heap to 512 MB and the Reduce task heap to 1.5 GB.
  - **Memory requirements for Map tasks** will depend how much output is being generated from each Map. A 128 MB block size with sort applications requires about 200 MB to store intermediate records without any spill. This memory can be managed using configuration parameters that follow the examples *io.sort.mb=200mb*, *io.sort.record.percent=.15*, and *io.sort.spill.percent=1.0*.
  - **Memory use for Reduce tasks** can also be tuned. Intel testing suggests that the default settings for most parameters are optimal, although *mapred.job.reduce.input.buffer.percent* can be changed to 0.7 so the reducer does not need to empty out the memory before it starts the final merge.

- **Number of concurrent Map and Reduce tasks.** On Intel Xeon processors, the optimal number of Map tasks tends to be the number of logical cores, and the number of reducers should be set equal to the number of physical cores. These settings can be configured using the *mapred.tasktracker.map.tasks.maximum* and *mapred.tasktracker.reduce.tasks.maximum* parameters.
- **NameNode and DataNodes request handler count and thread count.** If enough memory and compute resources are available on the NameNode, the number of thread handles in the NameNode can be increased to 100 or more to enable support for larger numbers of concurrent requests. Increasing the handler count for DataNodes can also be beneficial, particularly when SSDs or fast storage are being used.
- **Reducing network latency for IPC communications between nodes.** Set *ipc.server.tcpnodelay* and *ipc.client.tcpnodelay* to 'true'.
- **Heartbeat frequency between job and task trackers.** The default heartbeat frequency is 3 seconds. For smaller jobs where each Map task completes more quickly, this setting could delay task-completion notifications and scheduling of new tasks. Setting *mapreduce.tasktracker.outofband.heartbeat* to send immediate indications for job completions can improve job performance. Tuning heartbeat frequency using *mapreduce.tasktracker.outofband.heartbeat.damper* parameters can also provide some benefit.
- **Speculative task executions.** Hadoop may schedule the same tasks on multiple nodes as a safeguard against node failure or delayed execution. This practice is useful when it takes advantage of unused resources, but not when the cluster is running at capacity. Therefore, disabling speculative task executions is sometimes advisable, particularly in the presence of high-performance processing, storage, and network resources.
- **Intermediate Map output compression.** Enabling compression for intermediate Map output can help improve performance on a cluster that is bound by storage or network performance. Note that, when using 10GbE and SSDs, compressing the output may provide little if any improvement.
- **HDFS block size.** The optimal size of HDFS blocks is workload-specific, but larger block sizes often do not generate better performance, because they may cause an extra load on memory and tend to cause intermediate spills in the Map phase. At the same time, smaller block sizes create extra overhead for smaller and more parallel tasks. In Intel testing, the 128 MB block size gave the best overall performance for the TeraSort benchmark.

### 5.3 Future Enhancements

As described in this paper, Intel has been able to produce enormous performance advantages using the latest Intel Xeon processors, SSDs, and 10 Gigabit Intel Ethernet Converged Network Adapters, as well as the Intel® Distribution for Apache Hadoop\* software. Ongoing advances with all of these components are expected to produce further performance benefits in big data implementations.

A wide variety of software enhancements are under consideration for Hadoop and other big data technologies. One example is the possibility of accelerating the transport layer by replacing HTTP-based inter-node communication with other, more optimized options, improving overall throughput without adding physical resources. This and other software enhancements are an ongoing area of research at Intel.

Virtualized, pre-built Hadoop clusters for business analytics and other big data usages represent another promising area of research and development, with the potential for a number of benefits, such as the following:

- **Reducing the complexity** of implementing Hadoop environments
- **Encapsulating best practices** for configuring virtualized resources, without manual tuning
- **Enabling multi-purpose environments** built on big data technologies

### 6 Conclusion

The ability to store and analyze huge amounts of unstructured data promises ongoing opportunities for businesses, academic institutions, and government organizations. Intel has shown through this research that significant performance gains are possible from Hadoop through a balanced infrastructure based on well-selected hardware components and the use of the Intel® Distribution for Apache Hadoop\* software.

The results featured in this paper are part of a large and growing body of research being conducted at Intel and elsewhere in the industry to identify best practices for building and operating Hadoop clusters and other big data solutions, as well as for developing and tuning software to run optimally in those environments. That progress continues to help guide the computing industry toward simplified, low-cost implementations to drive the future of pervasive real-time analytics.

Learn more by visiting the following pages:

[hadoop.intel.com](http://hadoop.intel.com)

[www.intel.com/bigdata](http://www.intel.com/bigdata)

[www.intel.com/go/ethernet](http://www.intel.com/go/ethernet)

[www.intel.com/xeonE5](http://www.intel.com/xeonE5)

[www.intel.com/storage](http://www.intel.com/storage)

<sup>1</sup> TeraSort Benchmarks conducted by Intel in December 2012. Custom settings: `mapred.reduce.tasks=100` and `mapred.job.reuse.jvm.num.tasks=-1`. For more information: <http://hadoop.apache.org/docs/current/api/org/apache/hadoop/examples/terasort/package-summary.html>.

<sup>2</sup> Cluster configuration: One head node (name node, job tracker), 10 workers (data nodes, task trackers), Cisco Nexus 5020 10 Gigabit switch. Baseline worker node: SuperMicro SYS-1026T-URF 1U servers with two Intel® Xeon® processors X5690 @ 3.47 GHz, 48 GB RAM, 700 GB 7200 RPM SATA hard drives, Intel® Ethernet Server Adapter I350-T2, Apache Hadoop® 1.0.3, Red Hat Enterprise Linux® 6.3, Oracle Java® 1.7.0\_05.

Upgraded processor and base system in worker node: Dell PowerEdge® R720 2U servers with two Intel® Xeon® processors E5-2690 @ 2.90 GHz, 128 GB RAM. Upgraded storage in worker node: Intel® Solid-State Drive 520 Series. Upgraded network adapter in worker node: Intel® Ethernet Converged Network Adapter X520-DA2. Upgraded software in worker node: Intel® Distribution for Apache Hadoop® software 2.1.1.

<sup>3</sup> The Intel® Solid-State Drive 520 Series is currently not validated for data center usage.

<sup>4</sup> Solid-state drive performance varies by capacity.

<sup>5</sup> Performance measured using Iometer® with Queue Depth 32. Tests conducted in December 2012.

<sup>6</sup> Baseline worker node: SuperMicro SYS-1026T-URF 1U servers with two Intel® Xeon® processors X5690 @ 3.47 GHz, 48 GB RAM, 700 GB 7200 RPM SATA hard drives, Intel® Ethernet Server Adapter I350-T2, Apache Hadoop® 1.0.3, Red Hat Enterprise Linux® 6.3, Oracle Java® 1.7.0\_05.

Upgraded processor and base system in worker node: Dell PowerEdge® R720 2U servers with two Intel® Xeon® processors E5-2690 @ 2.90 GHz, 128 GB RAM, 700 GB 7200 RPM SATA hard drives.

<sup>7</sup> Baseline storage: 700 GB 7200 RPM SATA hard drives, upgraded storage: Intel® Solid-State Drive 520 Series.

<sup>8</sup> Source: Intel internal testing, December 2012.

<sup>9</sup> Baseline network adapter: Intel® Ethernet Server Adapter I350-T2, upgraded network adapter: Intel® Ethernet Converged Network Adapter X520-DA2.

<sup>10</sup> Upgraded software in worker node: Intel® Distribution for Apache Hadoop® software 2.1.1.

Software and workloads used in performance tests may have been optimized for performance only on Intel® microprocessors. Performance tests, such as SYSmark® and MobileMark®, 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. For more information go to [www.intel.com/performance](http://www.intel.com/performance).

Results are based on Intel internal testing, using third party benchmark test data and software. Intel does not control or audit the design or implementation of third party benchmark data, software or Web sites referenced in this document. Intel encourages all of its customers to visit the referenced Web sites or others where similar performance benchmark data are reported and confirm whether the referenced benchmark data are accurate and reflect performance of systems available for purchase.

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel.

Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors.

Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information. The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order. Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or by visiting Intel's Web Site [www.intel.com/](http://www.intel.com/).

Copyright © 2013 Intel Corporation. All rights reserved. Intel, the Intel logo, and Xeon are trademarks of Intel Corporation in the U.S. and other countries.

\*Other names and brands may be claimed as the property of others.

Printed in USA

0313/ME/MESH/PDF

 Please Recycle

328340-001US

