White Paper
Big Data

Deploying Hadoop on SUSE® Linux Enterprise Server

Big Data Best Practices

Table of Contents

Executive Summary

Big data technologies are becoming increasingly important in almost all areas of today's IT world. According to a recent Gartner survey, 64 percent of organizations are investing or planning to invest in big data technology.[1]

As IT architects begin implementing big data initiatives it is crucial they choose the right components.

One of the most important and most used technologies in big data is Apache Hadoop. Hadoop is an open source big data framework that combines all required technology components to provide a fully functional big data infrastructure called a Hadoop cluster. From an operating system (OS) standpoint, a Hadoop cluster is a very special workload with specific requirements for the hardware and operating system.

This white paper describes how to effectively use SUSE Linux Enterprise Server 11 SP3 for Hadoop. This paper gives guidance on installing Hadoop on SUSE Linux Enterprise Server and on configuring and optimizing the operating system for Hadoop. It also includes a comparison between virtualized and bare-metal Hadoop workloads.

Scope of This Document

Planning, installing, configuring, optimizing, operating and maintaining Hadoop clusters are all non-trivial tasks that require a lot of special knowledge. This paper briefly discusses many of these tasks, such as selecting proper hardware and networking options, but they are largely outside its scope. This document focuses on the Linux operating system in a Hadoop big data environment. The intended audience for this document is IT architects who scope, plan and execute the deployment of a Hadoop cluster based on SUSE Linux Enterprise Server. The target audience also includes system administrators who maintain, operate and possibly extend existing Hadoop clusters.

Hadoop and the Operating System

Apache Hadoop is an open source software project that enables the distributed processing and storage of large data sets across clusters of commodity servers. It consists of two major parts: MapReduce and the Hadoop Distributed File System (HDFS). From an OS standpoint, a Hadoop cluster is a very special workload with specific requirements for the hardware and OS.

The OS resides between the Hadoop software layers and the hardware layers in the vertical Hadoop stack. Tasks like memory management, CPU and process scheduling, I/O routing and scheduling, networking, security and virtualization are all handled by the OS.

Learn more about SUSE and big data at:
www.suse.com/partners/big-data/

1 **www.gartner.com/newsroom/id/2593815**

Also, the installation and patching of Hadoop and other software components are core tasks of the operating system. In the context of a big Hadoop cluster, management tools can automate many of these tasks and may provide complete lifecycle management (installation, patching, updating, monitoring) for Hadoop clusters of any size.

Most Hadoop deployments today run on Linux servers[2]. SUSE Linux Enterprise Server offers the full power of Linux and all its components as an industry-proven and stable enterprise-class Linux distribution.

SUSE Linux Enterprise Server is already widely used in the big data space by many large companies, and not just for Hadoop. For example, SUSE Linux Enterprise Server is the foundation for Hadoop in the Teradata Aster Big Analytics Appliance and the recommended and supported operating system for the SAP HANA in-memory database. Many other big data databases and applications are certified for SUSE Linux Enterprise Server.

Comparing Bare-Metal and Virtualized Hadoop Workloads

If you want maximum performance from your Hadoop deployment, we recommend a dedicated Hadoop infrastructure. The Hadoop distributions reach the highest possible performance levels when running on bare-metal installations. This infrastructure should be relatively static with the occasional addition of nodes to handle increased data or processing needs.

However, there are many reasons why a virtualized Hadoop deployment might be a better choice. One benefit is the ability to use existing infrastructure during off hours, thus maximizing return on IT investment. In a development environment where you are evaluating Hadoop for business benefit or in an environment where ingest rates are low and data mining is only run periodically, virtualizing some or all of the infrastructure would be a good choice.

Also, Hadoop operates on the premise that you are bringing the computing power to where your data resides, rather than transporting your data to a central computing area. If your data resides in the cloud, it will likely make sense to build Hadoop in the cloud as well.

In a virtualized Hadoop deployment it is important to tune file systems, schedulers, drivers and so on to achieve maximum performance. When building a virtualized Hadoop infrastructure, we recommend you configure SUSE Linux Enterprise Server in the following way:

- **Use the NOOP scheduler.** *Doing otherwise means there are two I/O schedulers (the virtual and physical) trying to manage I/O.*
- **Use Virtio drivers within KVM.** *Virtio is a virtualization standard for network and disk device drivers in which the guest's device driver "knows" it is running in a virtual environment, and cooperates with the hypervisor. This enables virtual machine guests to get higher-performance network and disk operations.*
- **Turn off write barriers, atime and diratime in the virtual guest.** *By doing this, Linux will not write back file access times. This speeds up reads. You do not need Linux to record these times as Hadoop keeps its own atime attributes for its own blocks.*
- **Provision the virtual machine only for the amount of memory physically available on the node.** *In other words, don't use the over-commit options of modern hypervisors that allow virtual machines to claim more memory than is available for either balloon or swapping. This prevents contention for resources and provides the best possible memory performance for the virtual machines.*

SUSE Linux Enterprise Server is the foundation for Hadoop in the Teradata Aster Big Analytics Appliance and the recommended and supported operating system for the SAP HANA in-memory database.

2 **www.linuxfoundation.org/news-media/announcements/2012/01/linux-foundation-releases-enterprise-linux-user-report**

Proposed Architecture Using SUSE Linux Enterprise Server

Architectural Overview

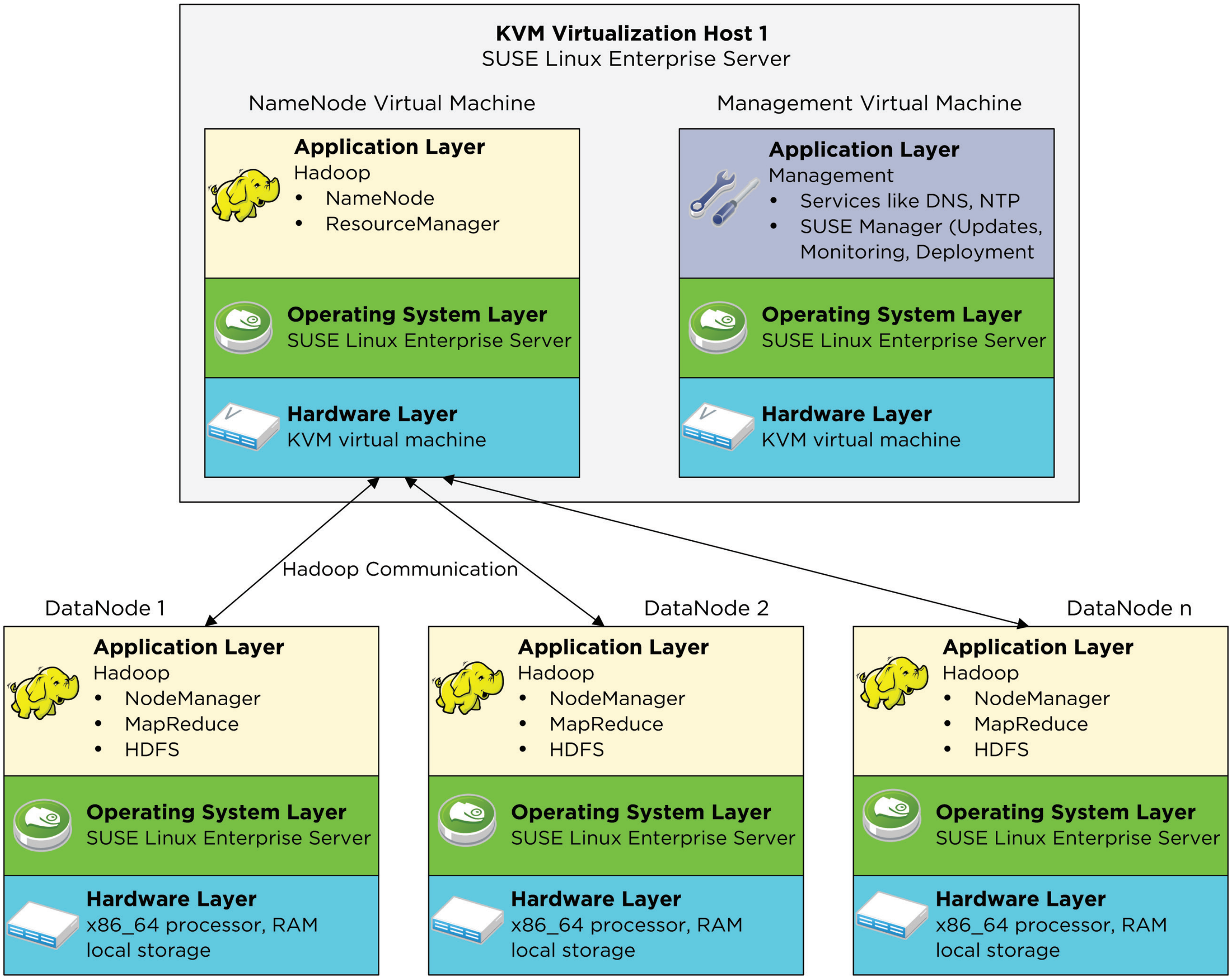

Figure 1. *For simplicity, this diagram shows only three DataNodes. Your Hadoop deployment will likely have far more DataNodes.*

Description of Hadoop Nodes and Services

Different kinds of node and service resources make up a Hadoop cluster. For a typical Hadoop implementation there are two types of nodes. The NameNode[3] is a distinct system, usually containing redundant components to protect against failures. It performs the critical function of providing access to the various cluster services, data, and compute resources on behalf of client system requests. While the NameNode does not store any of the cluster's data itself, this mapping function is critical to the operation of the cluster and is a single point of failure. For this reason, Hadoop deployments often use a secondary NameNode, since it can snapshot or checkpoint the namespace of files and locations for use as a recovery image.

The DataNode[4], or scale-out portion of the node count, provides computation power and manages its own storage elements. The DataNode executes the desired analytic compute tasks, leveraging this localized data access. A portion of the cluster's data blocks are located on each DataNode and are replicated across multiple DataNodes, with HDFS providing resiliency across these multiple nodes per HDFS configuration parameters.

From a logical architecture perspective, there are three main service daemons[5]—ResourceManager, NodeManager and Application Manager. The ResourceManager has a pluggable Scheduler, and considers resource requirements to allocate to the application. The per-node NodeManager provides, monitors and reports the memory, CPU, disk, network and so on for each node to the ResourceManager, and also launches the analytic applications. The per-application ApplicationManager interacts with the Scheduler by negotiating appropriate resources, along with tracking and monitoring the progress of the MapReduce operations.

3 **http://wiki.apache.org/hadoop/NameNode**
4 **http://wiki.apache.org/hadoop/DataNode**
5 **http://hadoop.apache.org/docs/r2.3.0/hadoop-yarn/hadoop-yarn-site/YARN.html**

Infrastructure Considerations

Before planning your Hadoop cluster, you need to consider and make decisions about your infrastructure. That includes hardware, network, network services and software infrastructure as well as the facility where you will locate the cluster. This is largely outside the scope of this document, but some general considerations follow.

Racks, Power and Cabling

If possible, spread cluster nodes over at least two different racks. By spreading cluster nodes over multiple racks, the outage of a whole rack will not lead to an outage of a cluster.

Hadoop nodes usually have high loads, causing very high power consumption per node. Make sure your power supply and cooling systems are adequate.

You should reduce the cabling in each rack to a minimum. In an ideal case, you will have three cables per node: one Ethernet cable and two power cables.

Networking

You networking choices will be determined by the size of your Hadoop cluster, as well as many other factors.

Be sure not to under-engineer your network. Hadoop is I/O bound, not processor bound, and will thus benefit from greater networking speeds. While Hadoop is quite resilient for most types of node failures or outages, failures are automatically handled by replication of the data to other systems, causing very high network traffic until completed. When in doubt, you should opt for more and higher-speed interfaces to have this fail-over scenario covered.

We recommend you separate the Ethernet network segment from any other network using a router or any other gateway. This ensures that broadcast frames stay within the Hadoop network.

More importantly, the network separation makes it easier to implement your own infrastructure services, like PXE boot with DHCP. This simplifies the cluster deployment.

Similarly, you should use exactly one separated IP network for the Hadoop cluster. Always use IPv4, not IPv6, in your setups. This simplifies the setup and ensures compatibility with all Hadoop components.

Network Architecture View

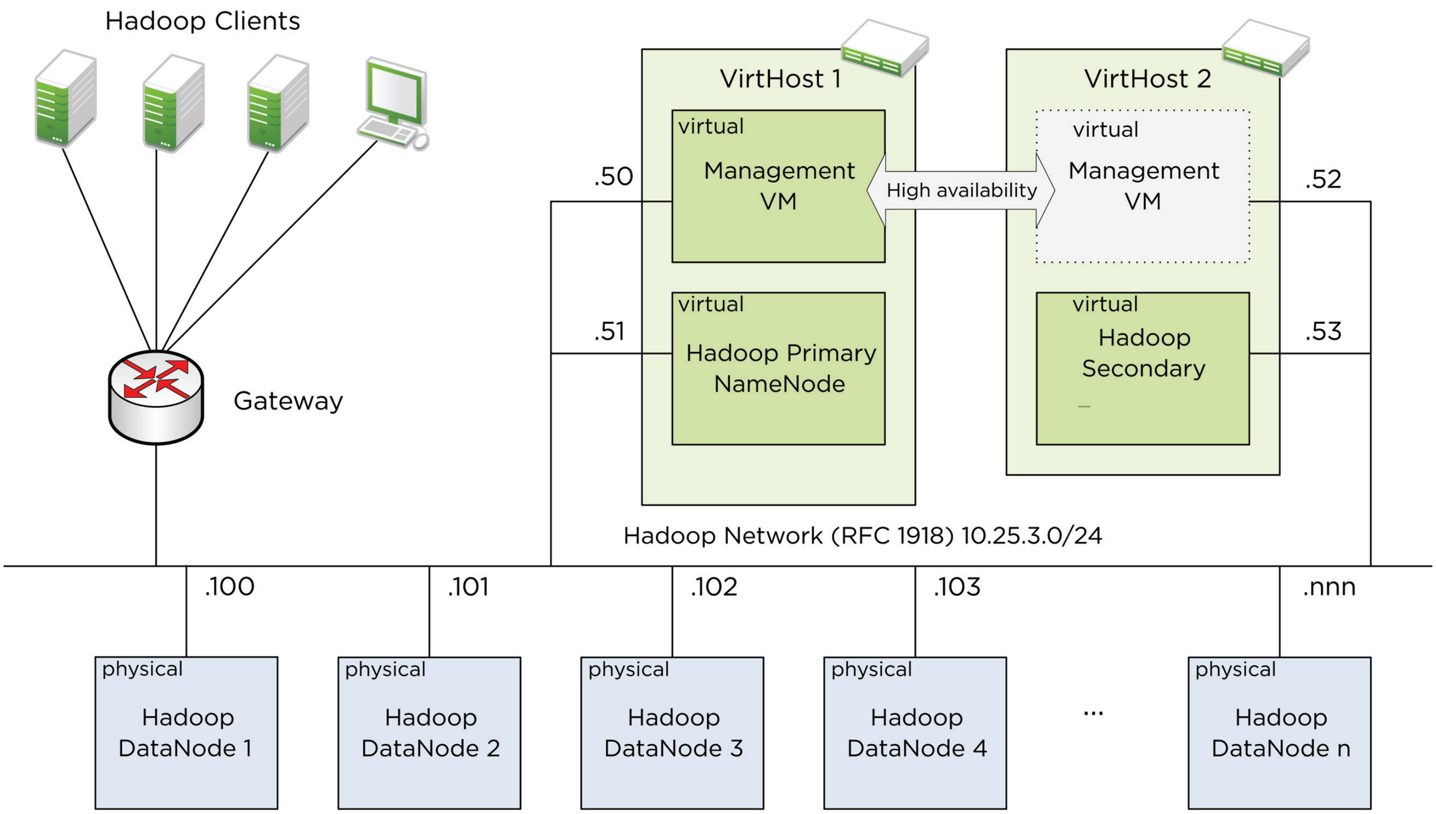

Figure 2. *This diagram shows the network architecture of a typical Hadoop deployment.*

Server and Storage Sizing Recommendations

Complete hardware sizing is beyond the scope of this document, but in general we recommend you maintain a one-to-one relationship for CPU core, memory channel, network interface controller and disk spindle. This will optimize your deployment platform, especially for the DataNodes.

For the purpose of the tests that informed this paper, we built our Hadoop deployment using IBM x3650 M4 servers, each with two Intel Xeon E5-2620 2.0 GHz 6-core processors, 64 GB RAM and three 300 GB hard drives.

Physical Architecture View

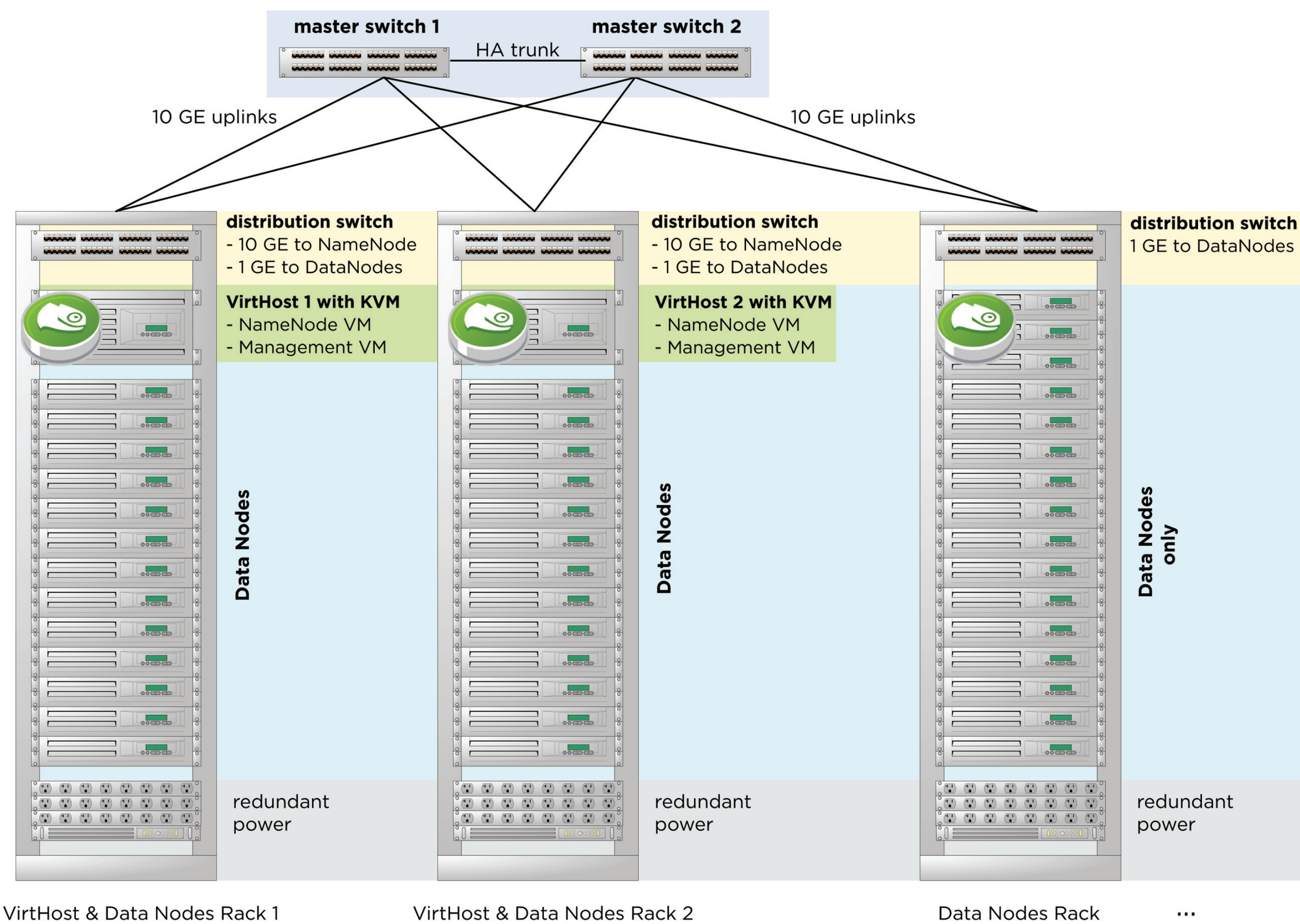

Figure 3. *This is an example of what a Hadoop deployment might look like in a rack infrastructure.*

FIRMWARE AND DEVICE DRIVERS

Due to the very close interaction between Hadoop, the Linux operating system and the underlying system platform, we recommended you ensure all firmware versions of the components in use match the latest stable releases provided by the manufacturer.

NOTE: *At this time there are no known firmware settings, such as in the BIOS, that need to be manipulated to specific values. As Hadoop solutions mature and become more popular, such settings may become available from the various vendors.*

Likewise, in the area of device drivers, you should ensure that you've deployed the latest available versions. For most industry standard x86_64 servers these versions are already included in the SUSE Linux Enterprise Server distribution.

PROCESSOR

Almost any modern x86_64 processor can deliver the necessary performance and desired characteristics for a Hadoop deployment. Often processors in the 2 to 2.5 GHz range provide the best overall total cost of ownership, given the number of nodes involved in a typical Hadoop deployment.

You should not take scaling, in terms of socket and core counts, to extreme limits. For most of the nodes the goal is to move the compute processes to the node containing the data. You don't need higher socket count systems, and in fact two-socket systems are actually the most economical and preferred way to obtain the highest throughput of Hadoop processes within the network and storage constraints. Similarly, you do not need the highest core count per socket. Instead, focus on balancing and matching the available I/O paths to the core count.

MEMORY

For the various node types in the Hadoop deployment we recommend:

- **NameNode:** *64–128 GB (about 1 GB of memory per one million blocks stored in HDFS should provide room to grow the cluster towards 100 nodes)*
- **DataNode:** *24–512 GB*

NETWORK INTERFACE

Your networking interfaces should support 1 GB/s Ethernet or higher. You may also want to include multiple interfaces, so you can bond them together to provide redundancy in the case of the NameNode or to increase throughput for the DataNode.

STORAGE

In Hadoop, most data is replicated. By default there will be three copies made across your instance. When purchasing storage elements, budget for this amount of duplication.

Also, remember the primary principle of Hadoop is to bring the computing power to where the data resides. Hence, all of the DataNodes systems will benefit from having a full complement of local disk storage, while the NameNode requires fewer local drives. We recommend the following:

- **NameNode:** *At least one drive for the operating system (but preferably mirrored across a pair of drives in a RAID1 fashion for redundancy).*
- **DataNodes:** *At least one drive for the operating system. Optimally one drive spindle per processor socket core configured individually as JBOD (Just a Bunch Of Disks).*

For performance reasons, having the fastest local disk drives possible is a good investment. You should opt for the highest rotational speed possible, preferably using SCSI or serial attached SCSI (SAS) drives. Also ensure that the storage drive controller resident in the system itself provides direct hardware access to the drives. Avoid drive controllers that emulate RAID or logical volume access via software RAID functionality.

SUMMARY

With SUSE Linux Enterprise Server at the core of your deployment, you will have a wide range of SUSE YES certified systems to choose from. After the initial procurement and setup of your Hadoop hardware deployment, we highly recommended that you perform integrity checks or benchmarks on each of the components. That way you can determine if any of the processor, memory, network or storage components are limiting performance.

Installation and Configuration of the Operating System

When you consider that many Hadoop deployments have hundreds of nodes and that some very large deployments have as many as four thousand nodes, you will understand the importance of streamlining the operating system installation for each of your Hadoop nodes.

Normal SUSE Linux Enterprise Server installation is quite easy using YaST and can be automated through AutoYaST. For installing the OS on all your Hadoop nodes, however, we recommend SUSE Manager. Each extension to Hadoop, such as Zookeeper and Hive, has its own control panel, and the enterprise vendors which offer Hadoop distributions each offer their own unique cluster management software. However, these control options do not touch the OS layer.

SUSE Manager helps you perform important OS layer actions such as:

- *Auto installation*
- *Patch management*
- *Configuration management*
- *Systems and remote management*
- *Monitoring*
- *Optimizing SUSE Linux Enterprise Server*

These capabilities make SUSE Manager a perfect complement to the monitoring and management capabilities provided in the cluster management software.

Below are the basic steps to creating your cluster with SUSE Manager. For detailed information on SUSE Manager, see the documentation[6].

1. Create the OS distribution that you will distribute to your nodes. You can set up an AutoYaST profile in SUSE Manager that determines how the distribution will be installed. See the next section for details of optimizing your SUSE Linux Enterprise Server distribution.
2. Deploy the distribution using SUSE Manager.
3. Deploy the NameNode and ResourceManager to the master node. To do this, go into the Hadoop configuration files and edit the IP address of the server so that it matches that of your intended master node, then deploy the configuration files using SUSE Manager.
4. Now push the Hadoop configuration files to all your worker nodes.
5. Connect to the NameNode server. You can verify that the NameNode configuration has been deployed using:
   ```
   cat/etc/hadoop/conf/core-site.xml
   ```
   Now format HDFS as the HDFS user:
   ```
   Sudo -u hdfs hdfs namenode -format
   ```
   Start the NameNode server:
   ```
   for x in 'cd /etc/init.d ; ls hadoop-hdfs-*' ;
   do sudo service $x start ; done
   ```
6. Create the file directories. We recommend this step because otherwise the Hadoop application you run might take ownership of the directory space.
   ```
   Sudo -u hdfs hadoop fs -mkdir /tmp

   Sudo -u hdfs hadoop fs -mkdir /user/hdfs

   Sudo -u hdfs hadoop fs -mkdir -p /var/lib/hadoop-
   hdfs/cache/mapred/mapred/stagin

   Sudo -u hdfs hadoop fs -chmod 1777 /var/lib/hadoop-
   hdfs/cache/mapred/mapred/stag

   Sudo -u hdfs hadoop fs -chown -R mapred /bar/lib/
   hadoop-hdfs/cache/mapred
   ```
7. Start MapReduce on the NameNode:
   ```
   for x in 'cd /etc/init.d ; ls hadoop-
   0.20-mapreduce-*' ; do sudo service $x start ; done
   ```

6 **www.suse.com/documentation/suse_manager/**

8. Your NameNode is now set up. Your final task is to start HDFS and MapReduce on your worker nodes:

```
for x in 'cd /etc/init.d ; ls hadoop-hdfs-*' ;
do sudo service $x start ; done

for x in 'cd /etc/init.d ; ls hadoop-0.20-
mapreduce-*' ; do sudo service $x start ; done
```

This is where SUSE Manager will prove very valuable. You can use a remote command (bash) in SUSE Manager to perform these commands on all your worker nodes at once, greatly speeding up your Hadoop cluster implementation.

This process only needs to happen once, when originally setting up your nodes, and can even be further automated using bootstrap scripts in SUSE Manager.

If you are creating your Hadoop deployment in the cloud, SUSE Cloud can work with SUSE Manager to help provide you with a similar ease of install.

Basic Optimization of the Operating System

To optimize your operating system you should start with the latest available Linux distribution from SUSE, which is SUSE Linux Enterprise Server 11 SP3. Following are general recommendations for optimizing SUSE Linux Enterprise Server for Hadoop.

Install SUSE Subscription Management Tool (SMT) or SUSE Manager

These tools can help you provide updates and patches for the whole Hadoop cluster. The Subscription Management Tool for SUSE Linux Enterprise Server helps customers manage their SUSE Linux Enterprise software updates while maintaining corporate firewall policy or regulatory compliance requirements. The Subscription Management Tool is a package proxy system that integrates with SUSE Customer Center and provides key SUSE Customer Center capabilities locally at the customer site, thus maintaining all the capabilities of the SUSE Customer Center while allowing a more secure, centralized deployment.

Update to the Latest Patch Level

As a general recommendation, we recommend updating to the latest patch level. These updates provide timely resolutions to improve stability and optimize performance of the SUSE Linux Enterprise platform. The SUSE Quality Assurance group rigorously tests maintenance updates and service packs prior to release. Through service packs and software updates, SUSE ensures that security threats are identified and resolved quickly and properly and our customer requests for new hardware enablement and enhanced features work flawlessly and are delivered promptly.

Use the NOOP Scheduler

cfq is the default scheduler for SUSE Linux Enterprise Server. Using it in this case would mean there are two I/O schedulers (the virtual and physical) trying to manage I/O. NOOP can be set for each block device that is part of your HDFS implementation as follows:

```
echo "noop >/sys/block/[DEVICE]/queue/scheduler"
```

Adjust vm.swappiness Value

vm.swappiness controls how aggressively the kernel swaps memory pages to disk. It can be set to a value between 0 – 100; the higher the value, the more aggressive the kernel is in seeking out inactive memory pages and swapping them to disk.

For Hadoop, you do not want any memory pages going to disk. Set vm.swappiness as follows:

```
sysctl -w vm.swappiness=0
```

and

```
echo "vm.swappiness=0" >> /etc/sysctl.conf
```

Choosing and Mounting the File System

SUSE Linux Enterprise Server supports both Ext3 and XFS. Over several benchmarking tests XFS performed better than Ext3. We recommend using XFS, as we have seen the best results and reliability using that file system. You should create the file system using the default options but mount it using the noatime, inode64 and nobarrier options. Linux records when files are created and last modified. With noatime, the file access times aren't written back; this speeds up reads. Hadoop keeps it own atime attributes for its own blocks. You can see more about the inode option at: **http://xfs.org/index.php/XFS_FAQ#Q:_What_is_the_inode64_mount_option_for.3F**

You should also use battery-backed disk controller caches and enable write-back caching and read-ahead caching. To maximize storage performance and still provide the highest storage integrity, we recommend using the nobarrier option with XFS. However this comes with some risks in the event of sudden power loss if controller caches are not protected with a battery backup unit. In general, for maximum XFS performance with high reliability we recommend:

- *Using the nobarrier mount option*
- *Setting the storage controller cache to writeback* if and only if *battery backup is present*
- *Disabling individual drive caches*

Tune ulimit settings

A busy Hadoop daemon will need to open a lot of files. The open fd ulimit in Linux defaults to 1024, which might be too low. The recommended value in the community is 16384.

Set net.core.somaxconn Kernel Parameter to 1024

somaxconn is the maximum number of pending connection requests queued for any listening socket. Add `net.core.somaxconn=1024` to `/etc/sysctl.conf`. This value is highly recommended by the community. It is especially useful for when DataNodes fail and Hadoop is re-balancing multiple blocks.

Use Jumbo Frames

This reduces the CPU overhead associated with network traffic and allows for higher overall utilization for data payloads per packet.

I/O Performance

To maximize performance for a Hadoop workload you should make sure to maximize your system's I/O performance. You can see general I/O tuning guidelines for SUSE Linux Enterprise Server in the SUSE Linux Enterprise Server documentation at: **http://doc.opensuse.org/products/draft/SLES/SLES-tuning_sd_draft/cha.tuning.io.html**

Recommendations for Configuring Aggregated Links

To maximize network performance and availability within a Hadoop cluster, you can aggregate multiple network interface cards into a single logical, bonded device as long as they are present in the nodes and cabled into the network infrastructure. Depending upon your network switching infrastructure constraints, you will want to enable one of the modes besides "active-backup" since active backup would use only one of the bonded slaves and the others would sit idle until the active one fails. You can see general setup guidelines for SUSE Linux Enterprise Server in the SUSE Linux Enterprise Server documentation at: **www.suse.com/documentation/sles11/book_sle_admin/data/sec_bond.html**

Installation and Configuration of Hadoop on SUSE

Providing a complete walk-through of the Hadoop installation process is outside the scope of this paper. Since Hadoop is an open source project, vendors such as Cloudera, Hortonworks and Intel provide their own Hadoop distribution. Each vendor also offers their own installation guides. Below are the names of those guides as well as important things to keep in mind when installing each distribution.

Cloudera

Cloudera provides the *CDH4 Installation Guide*. You can follow this guide exactly for your Cloudera Hadoop deployment on SUSE Linux Enterprise Server.

Hortonworks

Hortonworks provides the *Hortonworks Data Platform Cluster Planning Guide*, as well as general information on their Hortonworks Data Platform 2.0 page. Refer to the planning guide to ensure you have the proper specifications for your system.

You will begin by installing the Ambari server. If you are using SUSE Linux Enterprise Server 11 SP3, you will have to adjust some package names once the installer finishes.

```
# sed -i "s/apache2-mod_php5/apache2-mod_php53/g" /
var/lib/ambari-server/resources/stacks/HDP/2.0._/
services/GANGLIA/metainfo.xml
```

```
# sed -i "s/php5-json/php53-json/g" /var/lib/ambari-
server/resources/stacks/HDP/2.0._/services/NAGIOS/
metainfo.xml
```

You can now open a browser, log in to the Ambari server and proceed with the installation as described in the Hortonworks guide.

Intel

Intel provides the *Installation Guide for Intel Distribution for Apache Hadoop Software*, as well as an optimization and tuning guide.

The Intel distribution requires a number of different specific software packages and services. An extensive number of ports need to be available and accessible. Please see Intel's installation guide for details. Also note that Intel's Hadoop distribution comes with its own version of MySQL on which Hive depends. Make sure you do not have MySQL installed as part of your operating system roll-out.

Testing Methodologies

Hadoop is quickly growing and as it evolves, different benchmarking options become available. Some of the classic methodologies include TeraSort, TeraGen, RandomTextWriter, WordCount, HiBench and most importantly, your own data and applications.

Teragen and Terasort

Benchmarking clusters is not an exact science, but Hadoop provides a "standard" benchmark called Terasort, which tests both the HDFS and MapReduce layers of a Hadoop cluster. To benchmark with Terasort, we first generate the data required using Teragen. In the example below we also provide the rows parameter and the output file(s).

```
sudo -u hdfs hadoop jar /usr/lib/hadoop-0.20-
mapreduce/hadoop-examples.jar teragen 3500000000
teraout
```

This provides Terasort with the input data. Hadoop then uses MapReduce to sort the data into a total order.

```
sudo -u hdfs hadoop jar /usr/lib/hadoop-0.20-
mapreduce/hadoop-examples.jar terasort teraout
sortout
```

The above jobs will then be available at the ResourceManager user interface. These will indicate the start and finish time, enabling you to calculate the elapsed time. You can use this elapsed time to measure your implementation.

RandomTextWriter and Wordcount

Another set of benchmarking tools is the randomtextwriter and wordcount classes. These are available within the same java archive, typically in `/usr/lib/hadoop-0.20-mapreduce/hadoop-examples.jar`.

The randomtextwriter program uses MapReduce to run a distributed job where there is no interaction between the tasks and each task writes a large unsorted random sequence of words.

The wordcount program counts the occurrence of each word in the input data, which in this example was generated using randomtextwriter.

As for Teragen and Terasort, the details of these jobs will be available at the ResourceManager user interface. Use the elapsed time to compare your different configuration options.

Conclusion

After reading this paper you should have a good idea of how best to optimize and tune your operating system, SUSE Linux Enterprise Server, to get the most from your Hadoop deployment. With this deployment, you are on your way to harnessing the power of big data. In the future, your organization's big data plans will likely go beyond unstructured data, perhaps to new types of databases, like SAP HANA and Google BigTable. By using SUSE Linux Enterprise Server and working with a partner-focused organization like SUSE, you will be prepared to adapt to whatever new phase your big data initiative takes.

About SUSE and Big Data

With one of the largest ISV ecosystems in the enterprise software industry, SUSE is ready to support your big data implementation today through our industry-leading strategic alliances and partnerships with leading big data companies. These companies support their products when deployed on SUSE Linux Enterprise Server, as well as our OpenStack-based SUSE Cloud solution.

SUSE has a long tradition of working with big data related technologies. SUSE has long supported IBM System Z mainframes and is also the recommended and supported OS for the SAP

HANA in-memory database and Teradata's Aster Big Analytics appliance. SUSE is certified for and supports traditional high-end database systems, such as Oracle, and has partnered with SAP for over 15 years.

SUSE works closely with Hadoop vendors to provide SUSE customers and partners with all the choices they need to seamlessly deploy and operate their favorite Hadoop-based big data solutions on SUSE Linux Enterprise Server. In addition to Hadoop, SUSE is also a supported OS for many big data ecosystem vendors including:

- *NoSQL databases from vendors such as MongoDB and MarkLogic.*
- *The SAS Institute, Micro Strategies and others who design products to help organizations retrieve, clean and analyze all their data quickly.*
- *Business analytics programs from Jaspersoft, Talend and others.*

To learn more about our software partnerships, please visit our ISV Catalog at: **www.suse.com/partners/isv/**

B1 Systems

This white paper and the underlying technical workshops have been closely developed with our partner B1 Systems. B1 Systems is an independent company that provides consulting, training, development and support services for their valued partners. As a leader in the market, B1 Systems has comprehensive experience with Linux and other open source software projects that complement the SUSE customer support system. With over a decade of experience helping customers deploy SUSE Linux Enterprise, B1 Systems can effectively support and aid customers looking to integrate SUSE solutions with their existing infrastructures.

IBM

IBM is a global technology company offering solutions in many different areas, including big data. IBM believes that big data and analytics is a catalyst to help clients become more competitive and grow their business. IBM is helping clients harness big data to gather insights and act on the insights to transform their business. SUSE and IBM have partnered together for over 20 years. The testing that informed this white paper was performed on IBM hardware.

With one of the largest ISV ecosystems in the enterprise software industry, SUSE is ready to support your big data implementation today through our industry-leading strategic alliances and partnerships with leading big data companies. See who we work with at: **www.suse.com/partners/big-data/partners/**

www.suse.com

Contact your local SUSE Solutions Provider, or call SUSE at:

1 800 796 3700 U.S./Canada
1 801 861 4500 Worldwide

SUSE
Maxfeldstrasse 5
90409 Nuremberg
Germany

262-002508-002 | 04/14 | © 2014 SUSE LLC. All rights reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.