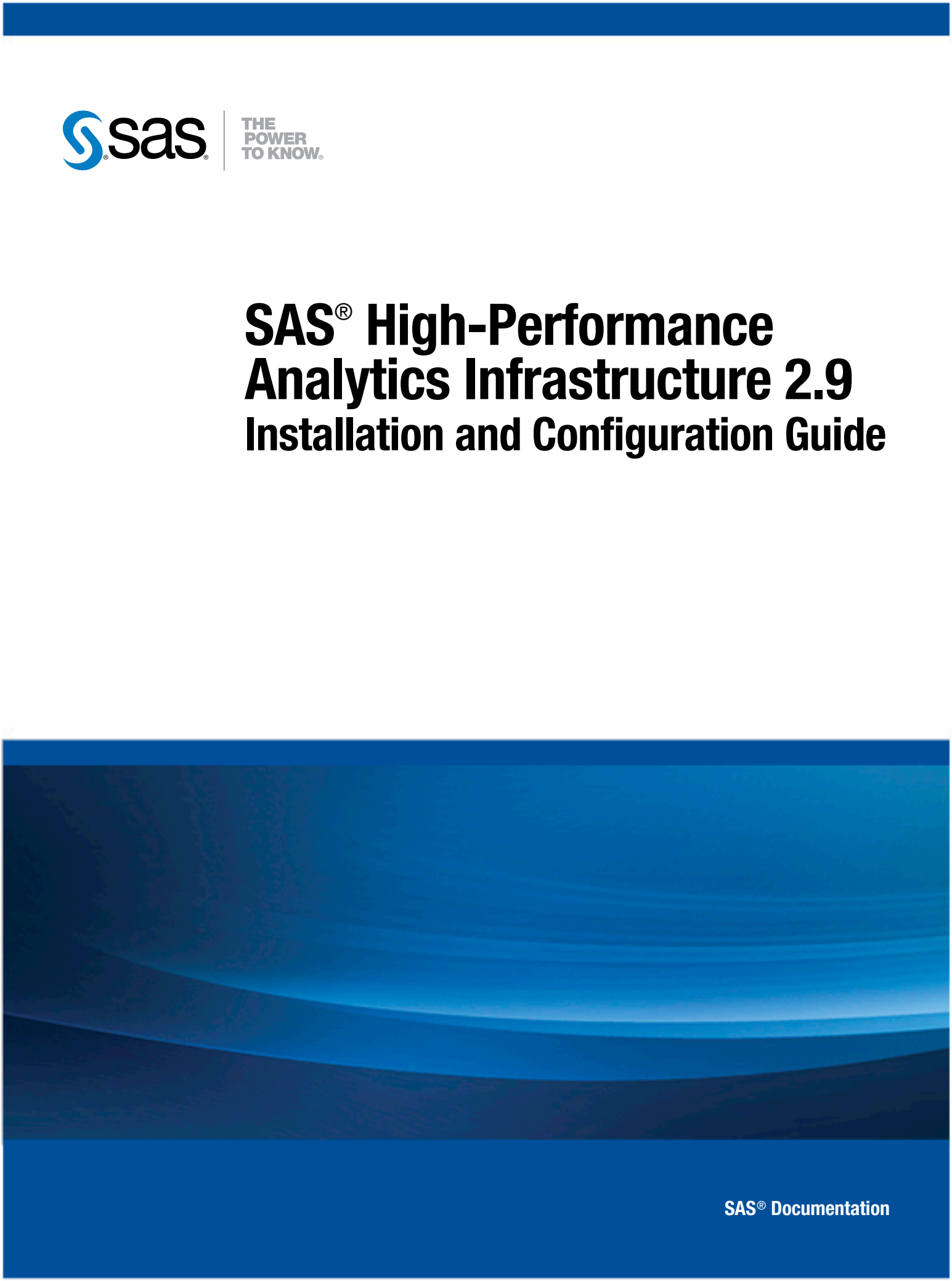


SAS® High-Performance Analytics Infrastructure 2.9

Installation and Configuration Guide

SAS® Documentation

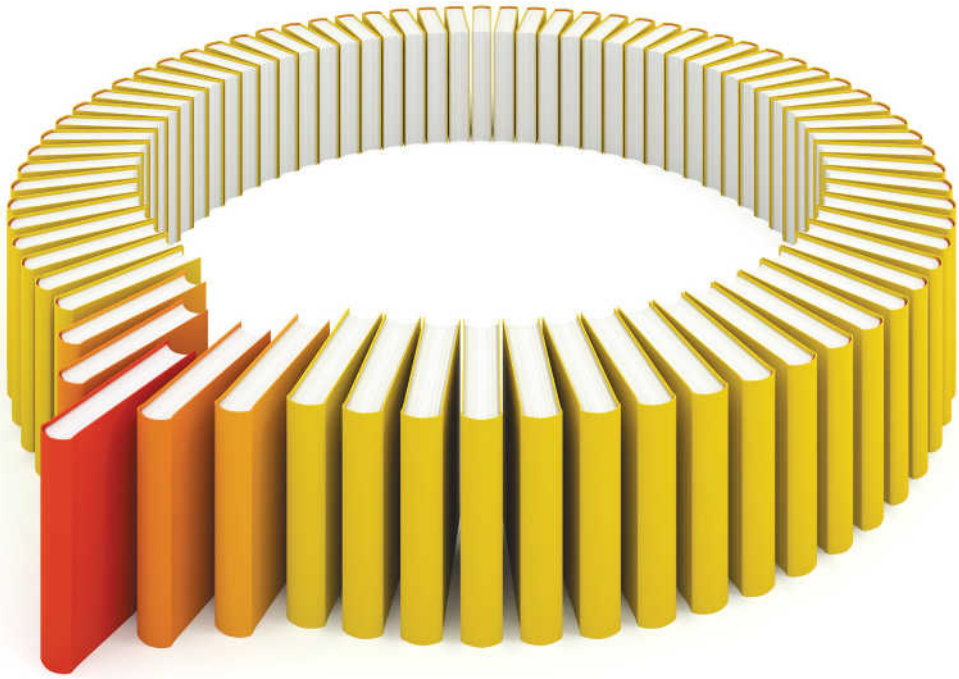
The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2014. *SAS® High-Performance Analytics Infrastructure 2.9: Installation and Configuration Guide*. Cary, NC: SAS Institute Inc.

SAS® High-Performance Analytics Infrastructure 2.9: Installation and Configuration Guide

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

October 2014

SAS provides a complete selection of books and electronic products to help customers use SAS® software to its fullest potential. For more information about our offerings, visit **support.sas.com/bookstore** or call 1-800-727-3228.



Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore
for additional books and resources.

Contents

Accessibility

For information about the accessibility of any of the products mentioned in this document, see the usage documentation for that product.

Recommended Reading

Here is the recommended reading list for this title:

- *Configuration Guide for SAS Foundation for Microsoft Windows for x64*, available at **http://support.sas.com/documentation/installcenter/en/ikfdtnwx6cg/66385/PDF/default/config.pdf**.
- *Configuration Guide for SAS Foundation for UNIX Environments*, available at **http://support.sas.com/documentation/installcenter/en/ikfdtnunxcg/66380/PDF/default/config.pdf**.
- *SAS/ACCESS for Relational Databases: Reference*, **http://support.sas.com/documentation/cdl/en/acreldb/67473/PDF/default/acreldb.pdf**.
- *SAS Deployment Wizard and SAS Deployment Manager: User's Guide*, available at **http://support.sas.com/documentation/installcenter/en/ikdeploywizug/66034/PDF/default/user.pdf**.
- *SAS Guide to Software Updates*, available at **http://support.sas.com/documentation/cdl/en/whatsdiff/66129/PDF/default/whatsdiff.pdf**.
- *SAS High-Performance Computing Management Console: User's Guide*, available at **http://support.sas.com/documentation/solutions/hpainfrastructure/**.
- *SAS In-Database Products: Administrator's Guide*, available at **http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf**.
- *SAS Intelligence Platform: Installation and Configuration Guide*, available at **http://support.sas.com/documentation/cdl/en/biig/63852/PDF/default/biig.pdf**.
- *SAS Intelligence Platform: Security Administration Guide*, available at **http://support.sas.com/documentation/cdl/en/bisecag/67045/PDF/default/bisecag.pdf**.

For a complete list of SAS books, go to support.sas.com/bookstore. If you have questions about which titles you need, please contact a SAS Book Sales Representative:

SAS Books
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-3228
Fax: 1-919-677-8166
E-mail: sasbook@sas.com

Web address: support.sas.com/bookstore

Chapter 1

Introduction to Deploying the SAS High-Performance Analytics Infrastructure

What Is Covered in This Document?

This document covers tasks that are required after you and your SAS representative have decided what software you need and on what machines you will install the software. At this point, you can begin performing some pre-installation tasks, such as creating a SAS Software Depot if your site already does not have one and setting up the operating system user accounts that you will need.

By the end of this document, you will have deployed the SAS High-Performance Analytics environment, and optionally, SAS High-Performance Computing Management Console, and SAS High-Performance Deployment of Hadoop.

You will then be ready to deploy your SAS solution (such as SAS Visual Analytics, SAS High-Performance Risk, and SAS High-Performance Analytics Server) on top of the SAS High-Performance Analytics infrastructure. For more information, see the documentation for your respective SAS solution.

Which Version Do I Use?

This document is published for each major release of the SAS High-Performance Analytics infrastructure, which consists of the following products:

- SAS High-Performance Computing Management Console, version 2.6
- SAS High-Performance Deployment for Hadoop, version 2.6
- SAS High-Performance Analytics environment, version 2.9

  (also referred to as the SAS High-Performance Node Installation)

Refer to your order summary to determine the specific version of the infrastructure that is included in your SAS order. Your order summary resides in your SAS Software Depot for your respective order under the **`install_doc`** directory (for example, **`C:\SAS Software Depot\install_doc\my-order\ordersummary.html`**).

Experimental Software

Experimental software is sometimes included as part of a production-release product. It is provided to (sometimes targeted) customers in order to obtain feedback. All experimental uses are marked Experimental in this document.

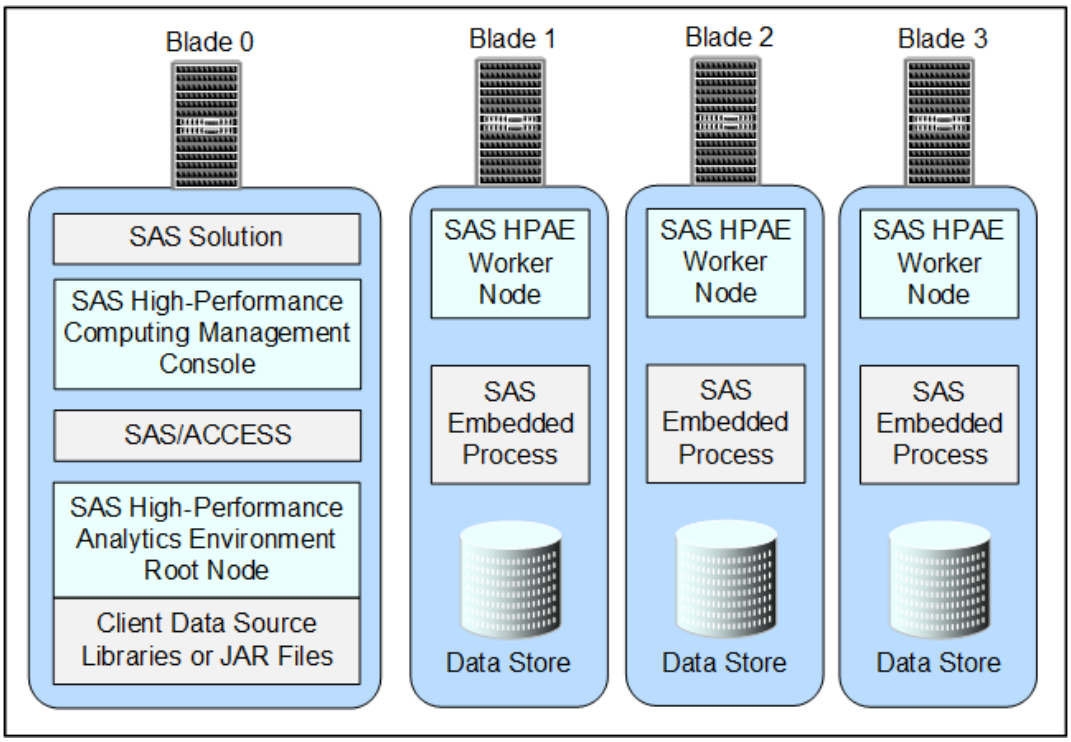
The design and implementation of experimental software might change before any production release. Experimental software has been tested prior to release, but it has not necessarily been tested to production-quality standards, and so should be used with care.

What is the Infrastructure?

The SAS High-Performance Analytics infrastructure consists of software that performs analytic tasks in a high-performance environment, which is characterized by massively parallel processing (MPP). The infrastructure is used by SAS products and solutions that typically analyze big data that resides in a distributed data storage appliance or Hadoop cluster.

The following figure depicts the SAS High-Performance Analytics infrastructure in its most basic topology:

Figure 1.1 *SAS High-Performance Analytics Infrastructure Topology (Simplified)*



The SAS High-Performance Analytics infrastructure consists of the following components:

- SAS High-Performance Analytics environment

  The SAS High-Performance Analytics environment is the core of the infrastructure. The environment performs analytic computations on an analytic cluster. The analytics cluster is a Hadoop cluster or a data appliance.

- (Optional) SAS High-Performance Deployment of Hadoop

  Some solutions, such as SAS Visual Analytics, rely on a SAS data store that is co-located with the SAS High-Performance Analytic environment on the analytic cluster. One option for this co-located data store is the SAS High-Performance Deployment for Hadoop. This is an Apache Hadoop distribution that is easily configured for use with the SAS High-Performance Analytics environment. It adds services to Apache Hadoop to write SASHDAT file blocks evenly across the HDFS filesystem. This even distribution provides a balanced workload across the machines in the cluster and enables SAS analytic processes to read SASHDAT tables at very impressive rates.

  Alternatively, these SAS high-performance analytic solutions can use a pre-existing, supported Hadoop deployment or a Greenplum Data Computing Appliance.

- (Optional) SAS High-Performance Computing Management Console

  The SAS High-Performance Computing Management Console is used to ease the administration of distributed, high-performance computing (HPC) environments. Tasks such as configuring passwordless SSH, propagating user accounts and public

keys, and managing CPU and memory resources on the analytic cluster are all made easier by the management console.

Other software on the analytics cluster include the following:

- SAS/ACCESS Interface and SAS Embedded Process

  Together the SAS/ACCESS Interface and SAS Embedded Process provide a high-speed parallel connection that delivers data from the co-located SAS data source to the SAS-High Performance Analytics environment on the analytic cluster. These components are contained in a deployment package that is specific for your data source.

  For more information, refer to the *SAS In-Database Products: Administrator's Guide*, available at **http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf** and the *SAS/ACCESS for Relational Databases: Reference* available at **http://support.sas.com/documentation/cdl/en/acreldb/67473/PDF/default/acreldb.pdf**.

  Note: For deployments that use Hadoop for the co-located data provider and access SASHDAT tables exclusively, SAS/ACCESS and SAS Embedded Process is *not* needed.

- Database client libraries or JAR files

  Data vendor-supplied client libraries—or in the case of Hadoop, JAR files—are required for the SAS Embedded Process to transfer data to and from the data store and the SAS High-Performance Analytics environment.

- SAS solutions

  The SAS High-Performance Analytics infrastructure is used by various SAS High-Performance solutions such as the following:

  - SAS High-Performance Analytics Server

    For more information, refer to **http://support.sas.com/documentation/onlinedoc/securedoc/index_hpa.html**.

  - SAS High-Performance Marketing Optimization

    For more information, refer to **http://support.sas.com/documentation/onlinedoc/mktopt/index.html**.

  - SAS High-Performance Risk

    For more information, refer to **http://support.sas.com/documentation/onlinedoc/hprisk/index.html**.

  - SAS Visual Analytics

    For more information, refer to **http://support.sas.com/documentation/onlinedoc/va/index.html**.

Where Do I Locate My Analytics Cluster?

Overview of Locating Your Analytics Cluster

You have two options for where to locate your SAS analytics cluster:

- Co-locate SAS with your data store.

- Separate SAS from your data store.

  When your SAS analytics cluster is separated (remote) from your data store, you have two basic options for transferring data:

  - Serial data transfer using SAS/ACCESS.
  - Parallel data transfer using SAS/ACCESS in conjunction with the SAS Embedded Process.

The topics in this section contain simple diagrams that describe each option for analytic cluster placement:

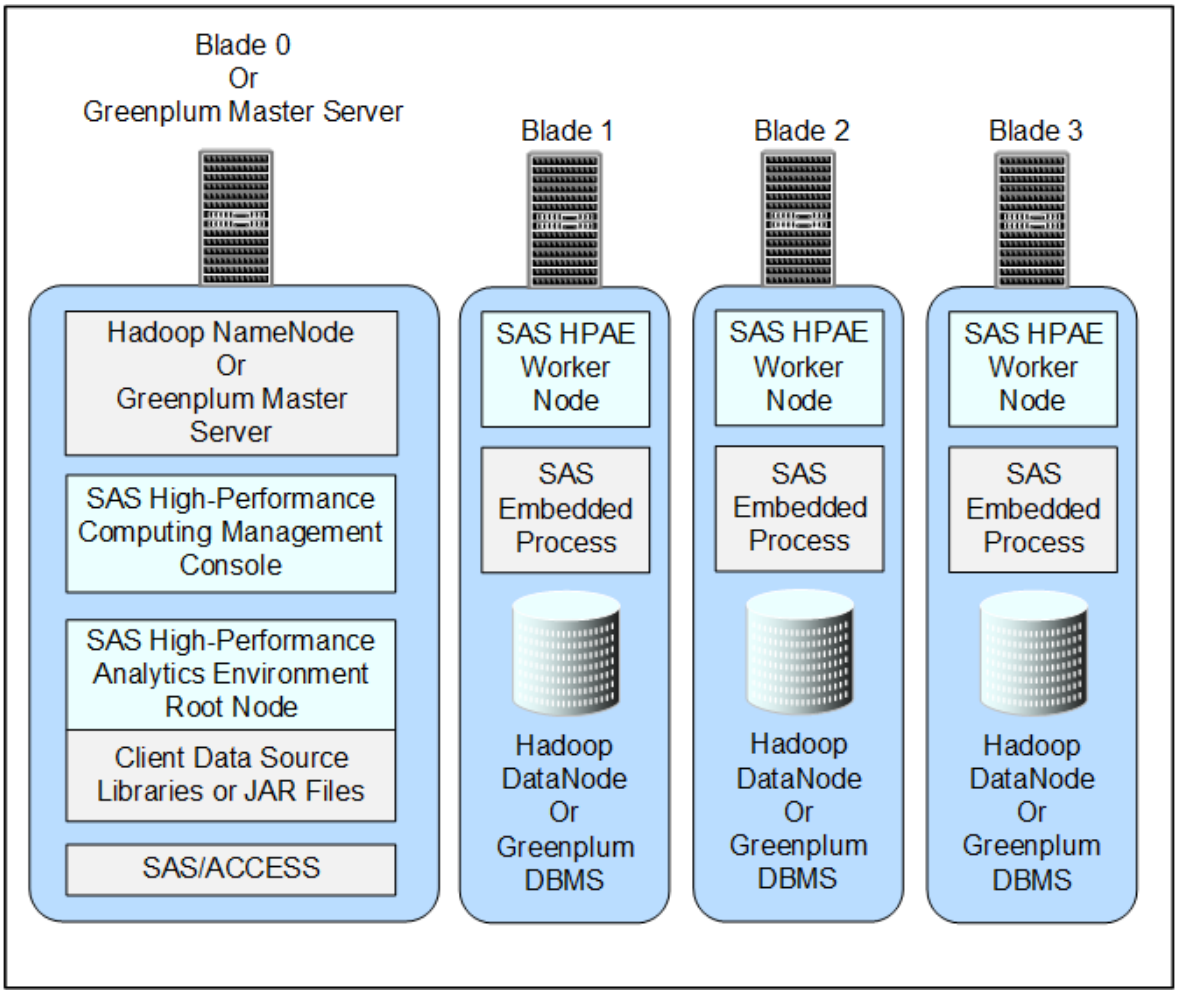
- Co-Located with the data store
- Remote from the data store (serial connection)
- Remote from the data store (parallel connection)

TIP Where you locate your cluster depends on a number of criteria. Your SAS representative will know the latest supported configurations, and can work with you to help you determine which cluster placement option works best for your site. Also, there might be solution-specific criteria that you should consider when determining your analytics cluster location. For more information, see the installation or administration guide for your specific SAS solution.

Analytic Cluster Co-Located with Your Data Store

The following figure shows the analytics cluster co-located on your Hadoop cluster or Greenplum data appliance:

Figure 1.2 *Analytics Cluster Co-Located on the Hadoop Cluster or Greenplum Data Appliance*

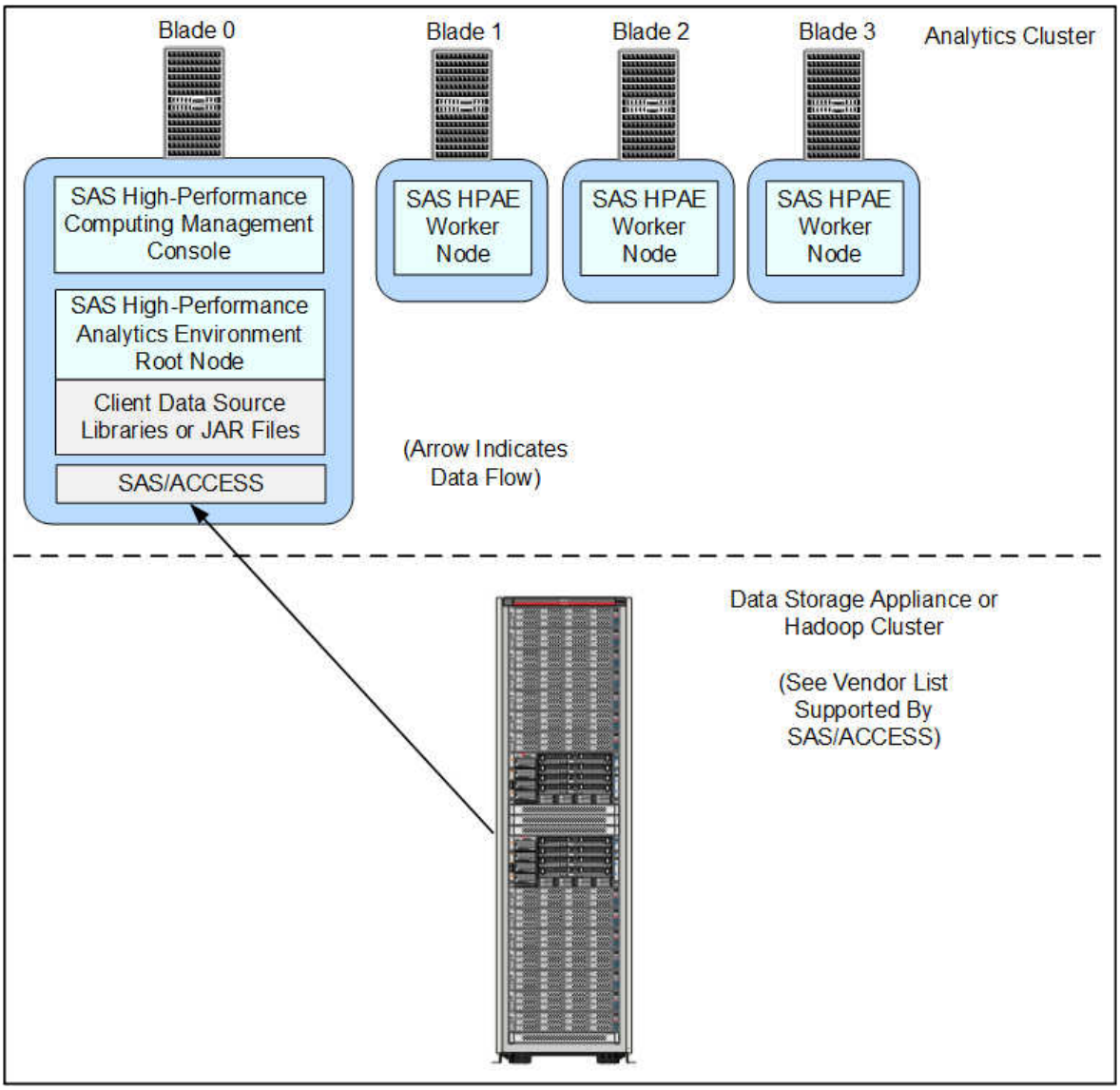


Note: For deployments that use Hadoop for the co-located data provider and access SASHDAT tables exclusively, SAS/ACCESS and the SAS Embedded Process are not needed.

Analytic Cluster Remote from Your Data Store (Serial Connection)

The following figure shows the analytics cluster using a serial connection to your remote data store:

Figure 1.3 *Analytics Cluster Remote from Your Data Store (Serial Connection)*

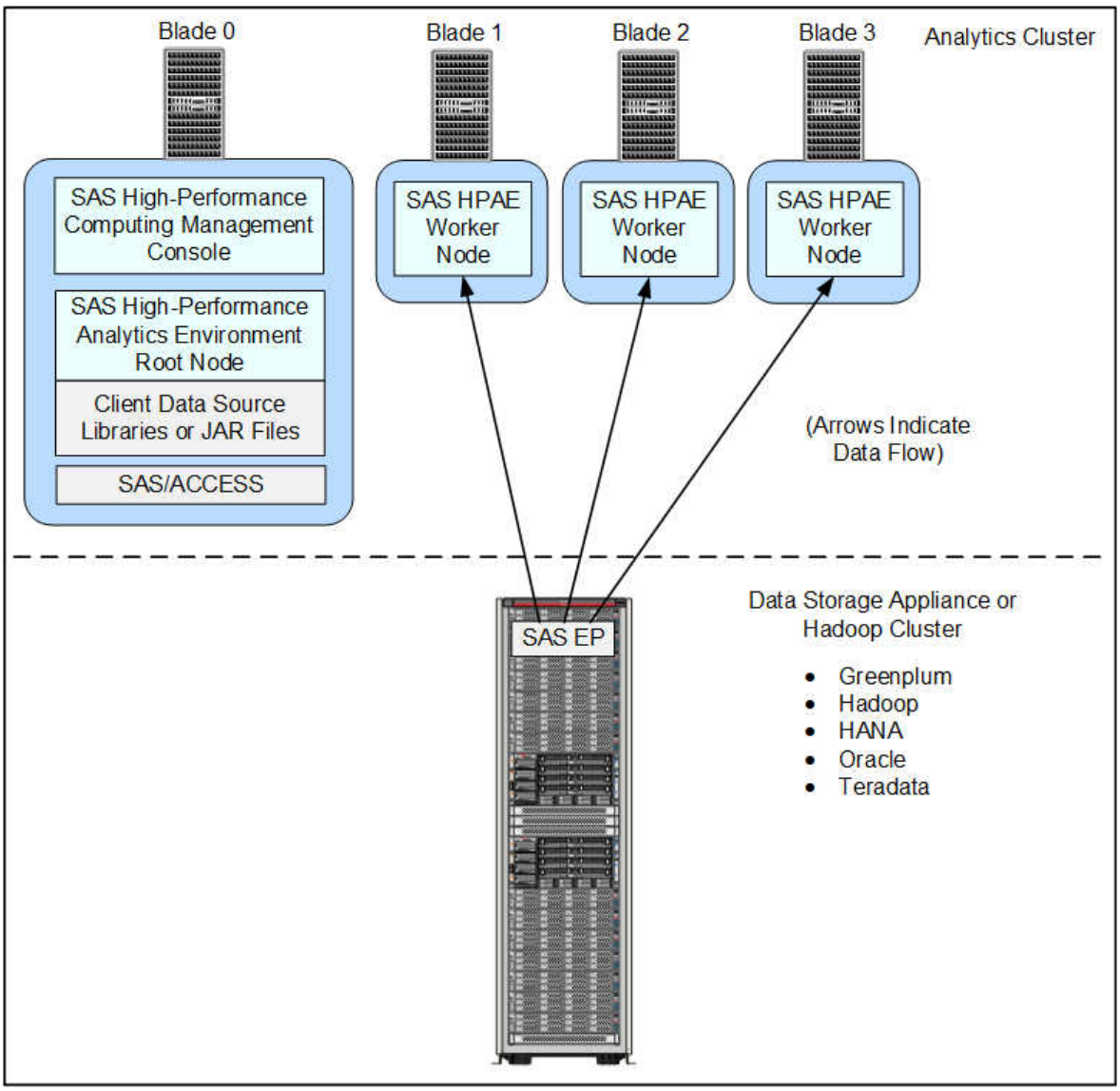


The serial connection between the analytics cluster and your data store is achieved by using the SAS/ACCESS Interface. SAS/ACCESS is orderable in a deployment package that is specific for your data source. For more information, refer to the *SAS/ACCESS for Relational Databases: Reference*, available at **`http://support.sas.com/documentation/onlinedoc/access/index.html`**.

Analytics Cluster Remote from Your Data Store (Parallel Connection)

The following figure shows the analytics cluster using a parallel connection to your remote data store:

Figure 1.4 *Analytics Cluster Remote from Your Data Store (Parallel Connection)*



Together the SAS/ACCESS Interface and SAS Embedded Process provide a high-speed parallel connection that delivers data from your data source to the SAS-High Performance Analytics environment on the analytic cluster. These components are contained in a deployment package that is specific for your data source. For more information, refer to the *SAS In-Database Products: Administrator's Guide*, available at `http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf`.

Deploying the Infrastructure

Overview of Deploying the Infrastructure

The following list summarizes the steps required to install and configure the SAS High-Performance Analytics infrastructure:

1. Create a SAS Software Depot.
2. Check for documentation updates.
3. Prepare your analytics cluster.
4. (Optional) Deploy SAS High-Performance Computing Management Console.
5. (Optional) Deploy Hadoop.
6. Configure your data storage.
7. Deploy the SAS High-Performance Analytics environment.

The following sections provide a brief description of each of these tasks. Subsequent chapters in the guide provide the step-by-step instructions.

Step 1: Create a SAS Software Depot

Create a SAS Software Depot, which is a special file system used to deploy your SAS software. The depot contains the SAS Deployment Wizard—the program used to install and initially configure most SAS software—one or more deployment plans, a SAS installation data file, order data, and product data.

Note: If you have chosen to receive SAS through Electronic Software Delivery, a SAS Software Depot is automatically created for you.

For more information, see "Creating a SAS Software Depot" in the *SAS Intelligence Platform: Installation and Configuration Guide*, available at **http://support.sas.com/documentation/cdl/en/biig/63852/HTML/default/p03intellplatform00installgd.htm**.

Step 2: Check for Documentation Updates

It is very important to check for late-breaking installation information in SAS Notes and also to review the system requirements for your SAS software.

- SAS Notes

  Go to this web page and click **Outstanding Alert Status Installation Problems**:

  http://support.sas.com/notes/index.html.

- system requirements

  Refer to the system requirements for your SAS solution, available at **http://support.sas.com/resources/sysreq/index.html**.

Step 3: Prepare Your Analytics Cluster

Preparing your analytics cluster includes tasks such as creating a list of machine names in your grid hosts file. Setting up passwordless SSH is required, as well as considering system umask settings. You must determine which operating system is required to install, configure, and run the SAS High-Performance Analytics infrastructure. Also, you will need to designate ports for the various SAS components that you are deploying.

For more information, see Chapter 2, "Preparing Your System to Deploy the SAS High-Performance Analytics Infrastructure," on page 11.

Step 4: (Optional) Deploy SAS High-Performance Computing Management Console

SAS High-Performance Computing Management Console is an optional web application tool that eases the administrative burden on multiple machines in a distributed computing environment.

For example, when you are creating operating system accounts and passwordless SSH on all machines in the cluster or on blades across the appliance, the management console enables you to perform these tasks from one location.

For more information, see Chapter 3, "Deploying SAS High-Performance Computing Management Console," on page 29.

Step 5: (Optional) Deploy Hadoop

If your site wants to use Hadoop as the co-located data store, then you can install and configure SAS High-Performance Deployment of Hadoop or use one of the supported Hadoop distributions.

For more information, see Chapter 4, "Deploying Hadoop," on page 41.

Step 6: Configure Your Data Provider

Depending on which data provider you plan to use with SAS, there are certain configuration tasks that you will need to complete on the Hadoop cluster or data appliance.

For more information, see Chapter 5, "Configuring Your Data Provider," on page 65.

Step 7: Deploy the SAS High-Performance Analytics Environment

The SAS High-Performance Analytics environment consists of a root node and worker nodes. The product is installed by a self-extracting shell script.

Software for the root node is deployed on the first host. Software for a worker node is installed on each remaining machine in the cluster or database appliance.

For more information, see Chapter 6, "Deploying the SAS High-Performance Analytics Environment," on page 75.

Chapter 2

Preparing Your System to Deploy the SAS High-Performance Analytics Infrastructure

Infrastructure Deployment Process Overview

Preparing your analytics cluster is the third of seven steps required to install and configure the SAS High-Performance Analytics infrastructure.

1. Create a SAS Software Depot.
2. Check for documentation updates.

⇨ **3. Prepare your analytics cluster.**

4. (Optional) Deploy SAS High-Performance Computing Management Console.

5. (Optional) Deploy Hadoop.
6. Configure your data provider.
7. Deploy the SAS High-Performance Analytics environment.

System Settings for the Infrastructure

Understand the system requirements for a successful SAS High-Performance Analytics infrastructure deployment before you begin. The lists that follow offer recommended settings for the analytics infrastructure on every machine in the cluster or blade in the data appliance:

- Modify **/etc/ssh/sshd_config** with the following setting:

  ```
  MaxStartups 1000
  ```

- Modify **/etc/security/limits.conf** with the following settings:

  ```
  * soft nproc 65536
  * hard nproc 65536
  * soft nofile 350000
  * hard nofile 350000
  ```

- Modify **/etc/security/limits.d/90-nproc.conf** with the following setting:

  ```
  * soft nproc 65536
  ```

- Modify **/etc/sysconfig/cpuspeed** with the following setting:

  ```
  GOVERNOR=performance
  ```

- The SAS High-Performance Analytics components require approximately 1.4 GB of disk space. SAS High-Performance Deployment of Hadoop requires approximately 300 MB of disk space for the software. This estimate does not include the disk space that is needed for storing data that is added to Hadoop Distributed File System (HDFS) for use by the SAS High-Performance Analytics environment.

  For more information, refer to the system requirements for your SAS solution, available at **http://support.sas.com/resources/sysreq/index.html**.

List the Machines in the Cluster or Appliance

Before the SAS High-Performance Analytics infrastructure can be installed on the machines in the cluster, you must create a file that lists all of the host names of the machines in the cluster.

On blade 0 (known as the Master Server on Greenplum), create an **/etc/gridhosts** file for use by SAS High-Performance Computing Management Console, SAS High-Performance Deployment of Hadoop, and the SAS High-Performance Analytic environment. (The grid hosts file is copied to the other machines in the cluster during the installation process.) If the management console is located on a machine that is not a member of the analytic cluster, then this machine must also contain a copy of **/etc/gridhosts** with its host name added to the list of machines. For more information, see "Deploying SAS High-Performance Computing Management Console" on page 29 before you start the installation.

You can use short names or fully qualified domain names so long as the host names in the file resolve to IP addresses. The long and short host names for each node must be resolvable from each node in the environment. The host names listed in the file must be in the same DNS domain and sub-domain. These host names are used for Message Passing Interface (MPI) communication and SAS High-Performance Deployment of Hadoop network communication.

The *root node* is listed first. This is also the machine that is configured as the following, depending on your data provider:

- SAS High-Performance Deployment of Hadoop or a supported Hadoop distribution: NameNode (blade 0)
- Greenplum Data Computing Appliance: Master Server

The following lines are an example of the file contents:

```
grid001
grid002
grid003
grid004
...
```

TIP You can use SAS High-Performance Computing Management Console to create and manage your grid hosts file. For more information, see *SAS High-Performance Computing Management Console: User's Guide* available at **http://support.sas.com/documentation/solutions/hpainfrastructure/**.

Review Passwordless Secure Shell Requirements

Secure Shell (SSH) has the following requirements:

- To support Kerberos, enable GSSAPI authentication methods in your implementation of Secure Shell (SSH).

  Note: If you are using Kerberos, see "Configure Passwordless SSH to use Kerberos" on page 15 .

- Passwordless Secure Shell (SSH) is required on all machines in the cluster or on the data appliance for the following user accounts:
  - root user account

    The root account must run SAS High-Performance Computing Management Console and the simultaneous commands (for example, **simsh**, and **simcp**). For more information about management console user accounts, see "Preparing to Install SAS High-Performance Computing Management Console" on page 21.
  - Hadoop user account

    For more information about Hadoop user accounts, see "Preparing to Deploy Hadoop" on page 22.
  - SAS High-Performance Analytics environment user account

    For more information about the environment's user accounts, see "Preparing to Deploy the SAS High-Performance Analytics Environment" on page 25.

Preparing for Kerberos

Kerberos Prerequisites

The SAS High-Performance Analytics infrastructure supports the Kerberos computer network authentication protocol. Throughout this document, we indicate the particular settings you need to perform in order to make parts of the infrastructure configurable for Kerberos. However, you must understand and be able to verify your security setup. If you are using Kerberos, you need the ability to get a Kerberos ticket.

Note: The SAS High-Performance Analytics environment using YARN is *not* supported with SAS High-Performance of Hadoop running in Secure Mode Hadoop (that is, configured to use Kerberos).

The list of Kerberos prerequisites are as follows:

- A Kerberos key distribution center (KDC)
- All machines configured as Kerberos clients
- Permissions to copy and secure Kerberos keytab files on all machines
- A user principal for the Hadoop user

  (This is used for setting up the cluster and performing administrative functions.)
- Encryption types supported on the Kerberos domain controller should be aes256-cts:normal and aes128-cts:normal

Generate and Test Host Principals

Every machine in the analytic cluster must have a host principal and a Kerberos keytab in order to operate as Kerberos clients.

To generate and test host principals, follow these steps:

1. Execute kadmin.local on the KDC.
2. Run the following command for each machine in the cluster:

   `addprinc –randkey host/$`*`machine-name`*

   where *machine-name* is the host name of the particular machine.
3. Generate host keytab files in kadmin.local for each machine, by running the following command:

   `ktadd -norandkey -k $`*`machine-name`*`.keytab host/$`*`machine-name`*

   where *machine-name* is the name of the particular machine.

   TIP When generating keytab files, it is a best practice to create files by machine. In the event a keytab file is compromised, the keytab will only contain the host principal associated with machine it resides on, instead of a single file that contains every machine in the environment.
4. Copy each generated keytab file to its respective machine under **`/etc`**, rename the file to krb5.keytab, and secure it with mode 600 and owned by root.

   For example:

```
cp keytab /etc/krb5.keytab

chown root:root /etc/krb5.keytab

chmod 600 /etc/krb5.keytab
```

5. At this point, any user with a principal in Kerberos should be able to use kinit successfully to get a ticket granting ticket.

   For example:

   ```
   kinit
   Password for hdfs@HOST.DOMAIN.NET:
   ```

   As the Hadoop user, you can run the klist command to check the status of your Kerberos ticket. For example:

   ```
   klist
   Ticket cache: FILE:/tmp/krb5cc_493
   Default principal: hdfs@HOST.DOMAIN.NET

   Valid starting    Expires           Service principal
   06/20/14 09:51:26 06/27/14 09:51:26 krbtgt/HOST.DOMAIN.NET@HOST.DOMAIN.NET
        renew until 06/22/14 09:51:26
   ```

   Note: If you intend to deploy the SAS Embedded Process on the cluster for use with SAS/ACCESS Interface to Hadoop, then a user keytab file for the user ID that runs HDFS is required.

Configure Passwordless SSH to use Kerberos

Passwordless access of some form is a requirement of the SAS High-Performance Analytics environment through its use of the Message Passing Interface (MPI). Traditionally, public key authentication in Secure Shell (SSH) is used to meet the passwordless access requirement. For Secure Mode Hadoop, GSSAPI with Kerberos is used as the passwordless SSH mechanism. GSSAPI with Kerberos not only meets the passwordless SSH requirements, but also supplies Hadoop with the credentials required for users to perform operations in HDFS with SAS LASR Analytic Server and SASHDAT files. Certain options must be set in the SSH daemon and SSH client configuration files. Those options are as follows and assume a default configuration of sshd.

To configure passwordless SSH to use Kerberos, follow these steps:

1. In the sshd_config file, set:

   ```
   GSSAPIAuthentication yes
   ```

2. In the ssh_config file, set:

   ```
   Host *.domain.net

   GSSAPIAuthentication yes

   GSSAPIDelegateCredentials yes
   ```

   where *domain.net* is the domain name used by the machine in the cluster.

   TIP Although you can specify **host ***, this is not recommended because it would allow GSSAPI Authentication with any host name.

Preparing the Analytics Environment for Kerberos

During startup, the Message Passing Interface (MPI) sends a user's Kerberos credentials cache (KRB5CCNAME) which can cause an issue when Hadoop attempts to use Kerberos credentials to perform operations in HDFS.

Under Secure Shell (SSH), a random set of characters are appended to the credentials cache file, so the value of the KRB5CCNAME environment variable is different for each machine. To set the correct value for KRB5CCNAME on each machine, you must use the option below when asked for additional options to MPIRUN during the analytics environment installation:

```
-genvlist `env | sed -e s/=.*/,/ | sed /KRB5CCNAME/d | tr -d
'\n'`TKPATH,LD_LIBRARY_PATH
```

For more information, see Table 6.1 on page 79.

You must use a launcher that supports GSSAPI authentication because the implementation of SSH that is included with SAS does not support it. Add the following to your SAS programs on the client:

```
option set=GRIDRSHCOMMAND="/path-to-file/ssh";
```

Preparing Hadoop for Kerberos

Overview of Preparing Hadoop for Kerberos

Preparing SAS High-Performance Deployment of Hadoop for Kerberos, consists of the following steps:

1. "Adding the Principals Required by Hadoop" on page 16
2. "Creating the Necessary Keytab Files" on page 17
3. "Download and Compile JSVC" on page 18
4. "Download and Use Unlimited Strength JCE Policy Files" on page 18
5. "Configure Self-Signed Certificates for Hadoop" on page 18

Adding the Principals Required by Hadoop

Secure Mode Hadoop requires a number of principals to work properly with Kerberos. Principals can be created using addprinc within kadmin.local on the KDC.

Your add principal command should resemble:

```
addprinc -randkey nn/$FQDN@$REALM
```

where $*FQDN* is a fully qualified domain name and $*REALM* is the name of the Kerberos Realm.

If you are using HDFS only, then only the HDFS-specific principals and keytab files are required. For HDFS, you need the following principals:

nn/$FQDN@$REALM
: NameNode principal. Create this for the NameNode machine only.

sn/$FQDN@$REALM
: Secondary NameNode principal. Create this for the NameNode machine only.

dn/$FQDN@$REALM
: DataNode principal. Create this for every machine in the cluster except for the NameNode machine.

HTTP/$FQDN@$REALM
: HTTP server principal, used by WebDFS. Create this for every machine in the cluster.

Creating the Necessary Keytab Files

After creating the principals, keytab files must be created for each service and machine. Keytab files are created using ktadd within kadmin.local on the KDC. Your ktadd command for the NameNode service should resemble the following:

ktadd -k */path-to-file/service_name*.keytab nn/$FQDN@$REALM

where *service_name* is a value like hdfs_nnservice, hdfs_dnservice, or http as shown in the following examples.

For example, your keytab files should be similar to the following:

- an hdfs_nnservice.keytab file containing three principals, with two encryption types per principal.

  The NameNode principal starts with nn, the Secondary NameNode principal starts with sn, and the host principal is included in the keytab file as described in the Apache Hadoop documentation. Your KVNO value might differ. hdfs_nnservice.keytab is copied to the NameNode only and owned by the Hadoop user with a mode of 600.

  ```
  Keytab name: FILE:hdfs_nnservice.keytab
  KVNO Principal
  ---- --------------------------------------------------------------------
  2 nn/node1.domain.net@DOMAIN.NET
  2 nn/node1.domain.net@DOMAIN.NET
  2 sn/node1.domain.net@DOMAIN.NET
  2 sn/node1.domain.net@DOMAIN.NET
  3 host/node1.domain.net@DOMAIN.NET
  3 host/node1.domain.net@DOMAIN.NET
  ```

- an hdfs_dnservice.keytab file that contains a principal for each DataNode host, with two encryption types per principal.

  The DataNode principle starts with dn and the host principals are included in the keytab file as described in the Apache Hadoop documentation. Your KVNO value might differ. hdfs_dnservice.keytab is copied only to the DataNodes and owned by the Hadoop user with a mode of 600.

  ```
  Keytab name: FILE:hdfs_dnservice.keytab
  KVNO Principal
  ---- --------------------------------------------------------------------
  2 dn/node2.domain.net@DOMAIN.NET
  2 dn/node2.domain.net@DOMAIN.NET
  2 dn/node3.domain.net@DOMAIN.NET
  2 dn/node3.domain.net@DOMAIN.NET
  ...
  2 dn/noden.domain.net@DOMAIN.NET
  2 dn/noden.domain.net@DOMAIN.NET
  3 host/node2.domain.net@DOMAIN.NET
  3 host/node2.domain.net@DOMAIN.NET
  3 host/node3.domain.net@DOMAIN.NET
  ```

```
3 host/node3.domain.net@DOMAIN.NET
...
3 host/noden.domain.net@DOMAIN.NET
3 host/noden.domain.net@DOMAIN.NET
```

- an http.keytab file that contains a principal for each machine in the cluster, with two encryption types per principal.

  Your KVNO value might differ. http.keytab is copied to all machines and is owned by the Hadoop user with a mode of 600.

```
Keytab name: FILE:http.keytab
KVNO Principal
---- --------------------------------------------------------------------
2 HTTP/node1.domain.net@DOMAIN.NET
2 HTTP/node1.domain.net@DOMAIN.NET
2 HTTP/node2.domain.net@DOMAIN.NET
2 HTTP/node2.domain.net@DOMAIN.NET
...
2 HTTP/noden.domain.net@DOMAIN.NET
2 HTTP/noden.domain.net@DOMAIN.NET
```

Download and Compile JSVC

The JSVC binary is required to start secure DataNodes on a privileged port. A server is started on the privileged port by root and the process is then switched to the secure DataNode user. The JSVC binary is not currently not included with Apache Hadoop. This section details where to get the source, how to compile it on a machine, and where to copy it.

To download and compile JSVC, follow these steps:

1. Download the JSVC source from apache at **http://archive.apache.org/dist/commons/daemon/source/commons-daemon-1.0.15-src.tar.gz**.
2. Extract the file into a directory you have write access to.
3. Change directory to **commons-daemon-1.0.15-src/src/native/unix**.

   Note: This directory contains the INSTALL.txt file that describes the installation process.
4. Execute **./configure** and correct any issues found during the pre-make.
5. After a successful configure, compile the binary by running **make**. This generates a file called jsvc in the directory.
6. Copy the jsvc file to **$HADOOP_HOME/sbin** on every DataNode in the cluster.

   Note the path to jsvc because this path is used later in hadoop-env.sh.

Download and Use Unlimited Strength JCE Policy Files

For encryption strengths above 128 bit, you must download the latest JCE (Java Cryptography Extension) for the JRE you are using. For this document, the JCE was used to provide 256-bit, AES encryption. Keep export and import laws in mind when dealing with encryption. Check with your site's legal department if you have any questions.

Configure Self-Signed Certificates for Hadoop

In order to secure Hadoop communications between cluster machines, you must setup the cluster to use HTTPS. This section goes through the process of generating the

necessary files and the configuration options required to enable HTTPS using self-signed certificates.

To configure self-signed certificates for Hadoop, follow these steps:

1. Create the client and server key directories by running the following commands on each machine:

   ```
   mkdir -p /etc/security/serverKeys
   ```

   ```
   mkdir -p /etc/security/clientKeys
   ```

2. Create the key store on each machine:

   ```
   cd /etc/security/serverKeys
   ```

   ```
   keytool -genkey -alias $shortname -keyalg RSA -keysize 1024
   -dname "CN=
   $shortname.domain.net,OU=unit,O=company,L=location,ST=state,
   C=country" -keypass $somepass -keystore keystore -storepass
   $somepass
   ```

3. Create the certificate on each machine:

   ```
   cd /etc/security/serverKeys
   ```

   ```
   keytool -export -alias $shortname -keystore keystore -rfc -
   file $shortname.cert -storepass $somepass
   ```

4. Import the certificate into the key store on each machine:

   ```
   cd /etc/security/serverKeys
   ```

   ```
   keytool -import -noprompt -alias $shortname -file
   $shortname.cert -keystore truststore -storepass $somepass
   ```

5. Import the certificate for each machine into the all store file. After you complete this on the first machine, copy the generated allstore file to the next machine until you have copied the file and run the following command on each machine. The end result is an allstore file containing each machine's certificate.

   ```
   cd /etc/security/serverKeys
   ```

   ```
   keytool -import -noprompt -alias $shortname -file
   $shortname.cert -keystore allstore –storepass $somepass
   ```

6. Use the command below to verify the allstore file has a certificate from every node.

   ```
   keytool -list -v -keystore allstore -storepass $somepass
   ```

7. Move the allstore file to the **`/etc/security/clientKeys`** directory on each machine.

8. Refer to Table 2.1 and make sure the generated files on each machine are in their respective locations, with appropriate ownership and mode.

   The allstore file to be used is the one containing all the certificates, which was verified in the previous step. The directories **`/etc/security/serverKeys`** and **`/etc/security/clientKeys`** should have a mode of 755 and owned by hdfs:hadoop.

   Table 2.1 *Summary of Certificates*

Filename	Location	Ownership	Mode
keystore	**`/etc/security/serverKeys`**	hdfs:hadoop	r--r-----

Filename	Location	Ownership	Mode
truststore	**/etc/security/serverKeys**	hdfs:hadoop	r--r-----
allstore	**/etc/security/clientKeys**	hdfs:hadoop	r--r--r--
$shortname.cert	**/etc/security/serverKeys**	hdfs:hadoop	r--r-----

9. Make the following SSL related additions or changes to each respective file:
   - core-site.xml
   - ssl-server.xml
   - ssl-client.xml

   core-site.xml:

```
<property>
  <name>hadoop.ssl.require.client.cert</name>
  <value>false</value>
</property>
<property>
  <name>hadoop.ssl.hostname.verifier</name>
  <value>DEFAULT</value>
</property>
<property>
  <name>hadoop.ssl.keystores.factory.class</name>
  <value>org.apache.hadoop.security.ssl.FileBasedKeyStoresFactory</value>
</property>
<property>
  <name>hadoop.ssl.server.conf</name>
  <value>ssl-server.xml</value>
</property>
<property>
  <name>hadoop.ssl.client.conf</name>
  <value>ssl-client.xml</value>
</property>
```

   ssl-server.xml:

   Note: Thessl-server.xml file should be owned by hdfs:hadoop with a mode of 440. Replace *$somepass* with the keystore password. Because this file has the keystore password in clear-text, make sure that only Hadoop service accounts are added to the hadoop group.

```
<property>
  <name>ssl.server.truststore.location</name>
  <value>/etc/security/serverKeys/truststore</value>
</property>
<property>
  <name>ssl.server.truststore.password</name>
  <value>$somepass</value>
</property>
<property>
  <name>ssl.server.truststore.type</name>
  <value>jks</value>
</property>
```

```
<property>
  <name>ssl.server.keystore.location</name>
  <value>/etc/security/serverKeys/keystore</value>
</property>
<property>
  <name>ssl.server.keystore.password</name>
  <value>$somepass</value>
</property>
<property>
  <name>ssl.server.keystore.type</name>
  <value>jks</value>
</property>
<property>
  <name>ssl.server.keystore.keypassword</name>
  <value>$somepass</value>
</property>
```

ssl-client.xml:

Note: The ssl-client.xml file should be owned by hdfs:hadoop with a mode of 440. The same information from the preceding note applies to this file.

```
<property>
  <name>ssl.client.truststore.location</name>
  <value>/etc/security/clientKeys/allstore</value>
<property>
  <name>ssl.client.truststore.password</name>
  <value>$somepass</value>
</property>
<property>
  <name>ssl.client.truststore.type</name>
  <value>jks</value>
</property>
```

Preparing to Install SAS High-Performance Computing Management Console

User Account Considerations for the Management Console

SAS High-Performance Computing Management Console is installed from either an RPM or a tarball package and must be installed and configured with the root user ID. The root user account must have passwordless secure shell (SSH) access between all the machines in the cluster. The console includes a web server. The web server is started with the root user ID, and it runs as the root user ID.

The reason that the web server for the console must run as the root user ID is that the console can be used to add, modify, and delete operating system user accounts from the local passwords database (**`/etc/passwd`** and **`/etc/shadow`**). Only the root user ID has Read and Write access to these files.

Be aware that you do not need to log on to the console with the root user ID. In fact, the console is typically configured to use console user accounts. Administrators can log on to the console with a console user account that is managed by the console itself and does

not have any representation in the local passwords database or whatever security provider the operating system is configured to use.

Management Console Requirements

Before you install SAS High-Performance Computing Management Console, make sure that you have performed the following tasks:

- Make sure that the Perl extension perl-Net-SSLeay is installed.
- For PAM authentication, make sure that the Authen::PAM PERL module is installed.

  Note: The management console can manage operating system user accounts if the machines are configured to use the **/etc/passwd** local database only.
- Create the list of all the cluster machines in the **/etc/gridhosts** file. You can use short names or fully qualified domain names so long as the host names in the file resolve to IP addresses. These host names are used for Message Passing Interface (MPI) communication and Hadoop network communication. For more information, see "List the Machines in the Cluster or Appliance" on page 12.
- Locate the software.

  Make sure that your SAS Software Depot has been created. (For more information, see "Creating a SAS Software Depot" in the *SAS Intelligence Platform: Installation and Configuration Guide*, available at **http://support.sas.com/documentation/cdl/en/biig/63852/HTML/default/p03intellplatform00installgd.htm**.)

Preparing to Deploy Hadoop

If you are using Kerberos, see also "Preparing for Kerberos" on page 14.

Install Hadoop Using root

As is the case with most enterprise Hadoop distributions such as Cloudera or Hortonworks, root privileges are needed when installing SAS High-Performance Deployment of Hadoop.

The installer must be root in order to **chown** and **chmod** files appropriately. Unlike earlier releases, there is a new user (yarn), and the Hadoop user (hdfs) cannot change file ownership to another user. Also, installing Hadoop using root facilitates implementation of Kerberos and Secure Mode Hadoop. For more information, refer to the Apache document available at **http://hadoop.apache.org/docs/r2.4.0/hadoop-project-dist/hadoop-common/SecureMode.html**.

User Accounts for Hadoop

Apache recommends that the HDFS and YARN daemons and the MapReduce JobHistory server run as different Linux users. It is also recommended that these users share the same primary Linux group. The following table summarizes Hadoop user and group information:

Table 2.2 Hadoop Users and Their Primary Group

User:Group	Daemons
hdfs:hadoop	NameNode, Secondary NameNode, JournalNode, DataNode
yarn:hadoop	ResourceManager, NodeManager
mapred:hadoop	MapReduce JobHistory Server

The accounts with which you deploy Hadoop, MapReduce, and YARN must have passwordless secure shell (SSH) access between all the machines in the cluster.

TIP Although the Hadoop installation program can run as any user, you might find it easier to run **hadoopInstall** as root so that it can set permissions and ownership of the Hadoop data directories for the user account that runs Hadoop.

As a convention, this document uses an account and group named **hadoop** when describing how to deploy and run SAS High-Performance Deployment of Hadoop. **mapred** and **yarn** are used for the MapReduce JobHistory Server user and the YARN user, respectively. If you do not already have an account that meets the requirements, you can use SAS High-Performance Computing Management Console to add the appropriate user ID.

If your site has a requirement for a reserved UID and GID for the Hadoop user account, then create the user and group on each machine before continuing with the installation.

Note: We recommend that you install SAS High-Performance Computing Management Console before setting up the user accounts that you will need for the rest of the SAS High-Performance Analytics infrastructure. The console enables you to easily manage user accounts across the machines of a cluster. For more information, see "Create the First User Account and Propagate the SSH Key" on page 36.

SAS High-Performance Deployment of Hadoop is installed from a TAR.GZ file. An installation and configuration program, **hadoopInstall**, is available after the archive is extracted.

Preparing for YARN (Experimental)

When deploying the SAS High-Peformance Deployment for Hadoop, you must decide whether to use YARN. YARN stands for "Yet Another Resource Negotiator." It consists of a framework that manages execution and schedules resource requests for distributed applications. For information about how to configure the analytics environment with YARN, see "Resource Management for the Analytics Environment" on page 86.

Note: The SAS High-Performance Analytics environment using YARN is *not* supported with SAS High-Performance of Hadoop running in Secure Mode Hadoop (that is, configured to use Kerberos).

If you decide to use YARN with the SAS High-Peformance Deployment for Hadoop, you must do the following:

- Create Linux user accounts for YARN and MapReduce to run YARN and MapReduce jobs on the machines in the cluster.

These user accounts must exist on all the machines in the cluster and must be configured for passwordless SSH. For more information, see "User Accounts for Hadoop" on page 22.

- Create a Linux group and make it the primary group for the Hadoop, YARN, and MapReduce users.
- Provide YARN-related input when prompted during the SAS High-Peformance Deployment for Hadoop installation.

  For more information, see "Install SAS High-Performance Deployment of Hadoop" on page 43.

Install a Java Runtime Environment

Hadoop requires a Java Runtime Environment (JRE) or Java Development Kit (JDK) on every machine in the cluster. The path to the Java executable must be the same on all of the machines in the cluster. If this requirement is already met, make a note of the path and proceed to installing SAS High-Performance Deployment of Hadoop.

If the requirement is not met, then install a JRE or JDK on the machine that is used as the grid host. You can use the **simsh** and **simcp** commands to copy the files to the other machines in the cluster.

Example Code 2.1 *Sample simsh and simcp Commands*

```
/opt/TKGrid/bin/simsh mkdir /opt/java
/opt/TKGrid/bin/simcp /opt/java/jdk1.6.0_31 /opt/java
```

For information about the supported Java version, see **http://wiki.apache.org/hadoop/HadoopJavaVersions**. SAS High-Performance Deployment of Hadoop uses the Apache Hadoop 2.4 version.

Plan for Hadoop Directories

The following table lists the default directories where the SAS High-Performance Deployment of Hadoop stores content:

Table 2.3 *Default SAS High-Performance Deployment of HadoopDirectory Locations*

Default Directory Location	Description
hadoop-name	The **hadoop-name** directory is the location on the file system where the NameNode stores the namespace and transactions logs persistently. This location is formatted by Hadoop during the configuration stage.
hadoop-data	The **hadoop-data** directory is the location on the file system where the DataNodes store data in blocks.
hadoop-local	The **hadoop-local** directory is the location on the file system where temporary MapReduce data is written.
hadoop-system	The **hadoop-system** directory is the location on the file system where the MapReduce framework writes system files.

Note: These Hadoop directories must reside on local storage. The exception is the **`hadoop-data`** directory, which can be on a storage area network (SAN). Network attached storage (NAS) devices are not supported.

You create the Hadoop installation directory on the NameNode machine. The installation script prompts you for this Hadoop installation directory and the names for each of the subdirectories (listed in Table 2.3) which it creates for you on every machine in the cluster.

Especially in the case of the data directory, it is important to designate a location that is large enough to contain all of your data. If you want to use more than one data device, see "(Optional) Deploy with Multiple Data Devices" on page 47.

Preparing to Deploy the SAS High-Performance Analytics Environment

If you are using Kerberos, see also "Preparing for Kerberos" on page 14.

User Accounts for the SAS High-Performance Analytics Environment

This topic describes the user account requirements for deploying and running the SAS High-Performance Analytics environment:

- Installation and configuration must be run with the same user account.
- The installer account must have passwordless secure shell (SSH) access between all the machines in the cluster.

TIP We recommend that you install SAS High-Performance Computing Management Console before setting up the user accounts that you will need for the rest of the SAS High-Performance Analytics infrastructure. The console enables you to easily manage user accounts across the machines of a cluster. For more information, see "User Account Considerations for the Management Console" on page 21.

The SAS High-Performance Analytics environment uses a shell script installer. You can use a SAS installer account to install this software if the user account meets the following requirements:

- The SAS installer account has Write access to the directory that you want to use and Write permission to the same directory path on every machine in the cluster.
- The SAS installer account is configured for passwordless SSH on all the machines in the cluster.

The root user ID can be used to install the SAS High-Performance Analytics environment, but it is not a requirement. When users start a process on the machines in the cluster with SAS software, the process runs under the user ID that starts the process. Any user accounts running analytic environment processes must also be configured with passwordless SSH.

Consider Umask Settings

The SAS High-Performance Analytics environment installation script (described in a later section) prompts you for a umask setting. Its default is no setting.

If you do not enter any umask setting, then jobs, servers, and so on, that use the analytics environment create files with the user's pre-existing umask set on the operating system. If you set a value for umask, then that umask is used and overrides each user's system umask setting.

Entering a value of 027 ensures that only users in the same operating system group can read these files.

Note: Remember that the account used to run the LASRMonitor process (by default, sas) must be able to read the table and server files in **/opt/VADP/var** and any other related subdirectories.

Note: Remember that the LASRMonitor process that is part of SAS Visual Analytics must be run with an account (by default, sas) that can read the server signature file. (This signature file is created when you start a SAS LASR Analytic Server and the file is specified in SAS metadata. For more information, see "Establishing Connectivity to a SAS LASR Analytic Server" in Chapter 4 of *SAS Intelligence Platform: Data Administration Guide*, available at **http://support.sas.com/documentation/cdl/en/bidsag/67493/HTML/default/viewer.htm#n1y0g0l4bgiduzn1o6jdy4l8c61d.htm**.

You can also add umask settings to the resource settings file for the SAS Analytics environment. For more information, see "Resource Management for the Analytics Environment" on page 86.

For more information about using umask, refer to your Linux documentation.

Additional Prerequisite for Greenplum Deployments

For deployments that rely on Greenplum data appliances, the SAS High-Performance Analytics environment requires that you also deply the appropriate SAS/ACCESS interface and SAS Embedded Process that SAS supplies with SAS In-Database products. For more information, see *SAS In-Database Products: Administrator's Guide*, available at **http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf**.

Pre-installation Ports Checklist for SAS

While you are creating operating system user accounts and groups, you need to review the set of ports that SAS will use by default. If any of these ports is unavailable, select an alternate port, and record the new port on the ports pre-installation checklist that follows.

The following checklist indicates what ports are used for SAS by default and gives you a place to enter the port numbers that you will actually use.

We recommend that you document each SAS port that you reserve in the following standard location on each machine: **/etc/services**. This practice will help avoid port conflicts on the affected machines.

Note: These checklists are superseded by more complete and up-to-date checklists that can be found at **http://support.sas.com/installcenter/plans**. This website also contains a corresponding deployment plan and an architectural diagram. If you are a SAS solutions customer, consult the pre-installation checklist provided by your SAS representative for a complete list of ports that you must designate.

Table 2.4 *Pre-installation Checklist for SAS Ports*

SAS Component	Default Port	Data Direction	Actual Port
YARN ResourceManager Scheduler	8030	Inbound	
YARN ResourceManager Resource Tracker	8031	Inbound	
YARN ResourceManager	8032	Inbound	
YARN ResourceManager Admin	8033	Inbound	
YARN Node Manager Localizer	8040	Inbound	
YARN Node Manager Web Application	8042	Inbound	
YARN ResourceManager Web Application	8088	Inbound	
SAS High-Performance Computing Management Console server	10020	Inbound	
MapReduce Job History	10021	Inbound	
YARN Web Proxy	10022	Inbound	
MapReduce Job History Admin	10033	Inbound	
MapReduce Job History Web Application	19888	Inbound	
Hadoop Service on the NameNode	15452	Inbound	
Hadoop Service on the DataNode	15453	Inbound	
Hadoop DataNode Address	50010	Inbound	
Hadoop DataNode IPC Address	50020	Inbound	
Hadoop JobTracker	50030	Inbound	
Hadoop TaskTracker	50060	Inbound	
Hadoop Name Node web interface	50070	Inbound	
Hadoop DataNode HTTP Address	50075	Inbound	
Hadoop Secondary NameNode	50090	Inbound	
Hadoop Name Node Backup Address	50100	Inbound	

SAS Component	Default Port	Data Direction	Actual Port
Hadoop Name Node Backup HTTP Address	50105	Inbound	
Hadoop Name Node HTTPS Address	50470	Inbound	
Hadoop DataNode HTTPS Address	50475	Inbound	
SAS High-Performance Deployment of Hadoop	54310	Inbound	
SAS High-Performance Deployment of Hadoop	54311	Inbound	

Chapter 3

Deploying SAS High-Performance Computing Management Console

Infrastructure Deployment Process Overview

Installing and configuring SAS High-Performance Computing Management Console is an optional fourth of seven steps required to install and configure the SAS High-Performance Analytics infrastructure.

1. Create a SAS Software Depot.
2. Check for documentation updates.
3. Prepare your analytics cluster.
⇨ **4. (Optional) Deploy SAS High-Performance Computing Management Console.**
5. (Optional) Deploy Hadoop.
6. Configure your data provider.
7. Deploy the SAS High-Performance Analytics environment.

Benefits of the Management Console

Passwordless SSH is required to start and stop SAS LASR Analytic Servers and to load tables. For some SAS solutions, such as SAS High-Performance Risk and SAS High-Performance Analytic Server, passwordless SSH is required to run jobs on the machines in the cluster.

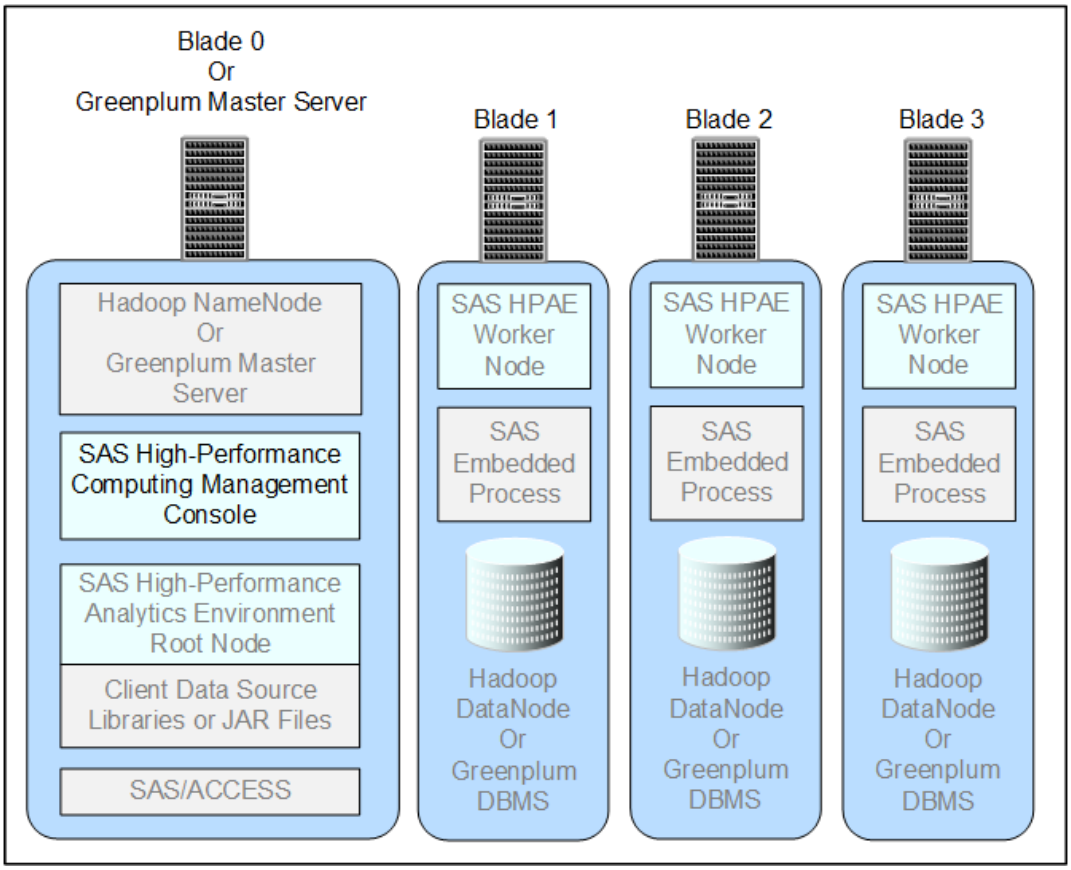
Also, users of some SAS solutions must have an operating system (external) account on all the machines in the cluster and must have the key distributed across the cluster. For more information, see "Create the First User Account and Propagate the SSH Key" on page 36.

SAS High-Performance Computing Management Console enables you to perform these tasks from one location. When you create *new* user accounts using SAS High-Performance Computing Management Console, the console propagates the public key across all the machines in the cluster in a single operation. For more information, see *SAS High-Performance Computing Management Console: User's Guide*, available at `http://support.sas.com/documentation/solutions/hpainfrastructure/`.

Overview of Deploying the Management Console

Deploying SAS High-Performance Computing Management Console requires installing and configuring components on a machine other than the Greenplum data appliance. In this document, the management console is deployed on the machine where the SAS Solution is deployed.

Figure 3.1 *Management Console Deployed with a Data Appliance*



Installing the Management Console

There are two ways to install SAS High-Performance Computing Management Console.

Install SAS High-Performance Computing Management Console Using RPM

To install SAS High-Performance Computing Management Console using RPM, follow these steps:

Note: For information about updating the console, see "Updating the SAS High-Performance Analytics Infrastructure" on page 105.

1. Make sure that you have reviewed all of the information contained in the section "Preparing to Install SAS High-Performance Computing Management Console" on page 21.
2. Log on to the target machine as root.
3. In your SAS Software Depot, locate the **`standalone_installs/SAS_High-Performance_Computing_Management_Console/2_6/Linux_for_x64`** directory.
4. Enter one of the following commands:
   - To install in the default location of **`/opt`**:

     ```
     rpm -ivh sashpcmc*
     ```

   - To install in a location of your choice:

     ```
     rpm -ivh --prefix=directory sashpcmc*
     ```

     where *directory* is an absolute path where you want to install the console.
5. Proceed to the topic "Configure the Management Console" on page 32.

Install the Management Console Using tar

Some versions of Linux use different RPM libraries and require and alternative means to Install SAS High-Performance Computing Management Console. Follow these steps to install the management console using tar:

1. Make sure that you have reviewed all of the information contained in the section "Preparing to Install SAS High-Performance Computing Management Console" on page 21.
2. Log on to the target machine as root.
3. In your SAS Software Depot, locate the **`standalone_installs/SAS_High-Performance_Computing_Management_Console/2_6/Linux_for_x64`** directory.
4. Copy sashpcmc-2.6.tar.gz to the location where you want to install the management console.
5. Change to the directory where you copied the tar file, and run the following command:

```
tar -xzvf sashpcmc-2.6.tar.gz
```

tar extracts the contents into a directory called **sashpcmc**.

6. Proceed to the topic "Configure the Management Console" on page 32.

Configure the Management Console

After installing SAS High-Performance Computing Management Console, you must configure it. This is done with the setup script.

1. Log on to the SAS Visual Analytics server and middle tier machine (blade 0) as root.
2. Run the setup script by entering the following command:

   ```
   management-console-installation-directory/opt/webmin/utilbin/setup
   ```

   Answer the prompts that follow.

   ```
   Enter the username for initial login to SAS HPC MC below.
   This user will have rights to everything in the SAS HPC MC and
   can either be an OS account or new console user. If an OS account
   exists for the user, then system authentication will be used. If
   an OS account does not exist, you will be prompted for a password.
   ```

3. Enter the user name for the initial login.

   ```
   Creating using system authentication
   Use SSL\HTTPS (yes|no)
   ```

4. If you want to use Secure Sockets Layer (SSL) when running the console, enter **yes**. Otherwise, enter **no**.
5. If you chose not to use SSL, then skip to Step 7 on page 32. Otherwise, the script prompts you to use a pre-existing certificate and key file or to create a new one.

   ```
   Use existing combined certificate and key file or create a new one (file|create)?
   ```

6. Make one of two choices:
   - Enter **create** for the script to generate the combined private key and SSL certificate file for you.

     The script displays output of the **openssl** command that it uses to create the private key pair for you.
   - Enter **file** to supply the path to a valid private key pair.

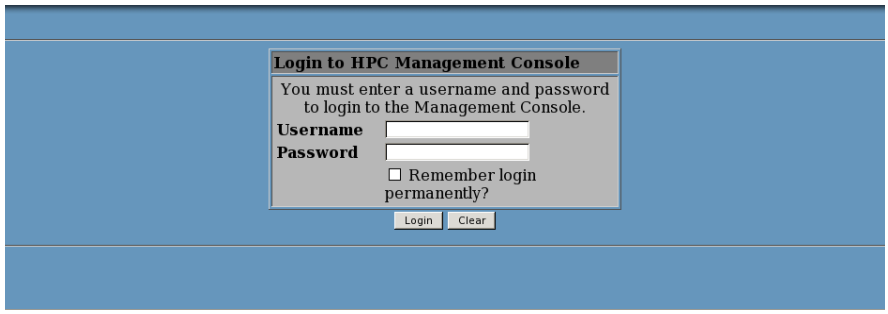
     When prompted, enter the absolute path for the combined certificate and key file.
7. To start the SAS High-Performance Computing Management Console server, enter the following command from any directory:

   ```
   service sashpcmc start
   ```

8. Open a web browser and, in the address field, enter the fully qualified domain name for the blade 0 host followed by port 10020.

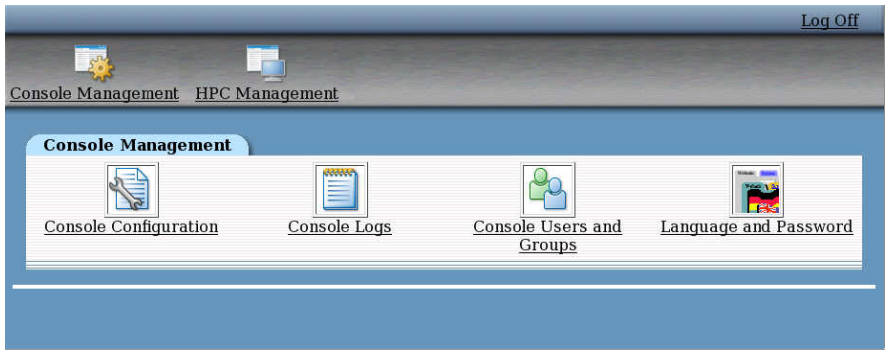
   For example: **https://myserver.example.com:10020**

   The Login page appears.



9. Log on to SAS High-Performance Computing Management Console using the credentials that you specified in Step 2.

   The Console Management page appears.



Create the Installer Account and Propagate the SSH Key

The user account needed to start and stop server instances and to load and unload tables to those servers must be configured with passwordless secure shell (SSH).

To reduce the number of operating system (external) accounts, it can be convenient to use the SAS Installer account for both of these purposes.

Implementing passwordless SSH requires that the public key be added to the authorized_keys file across all machines in the cluster. When you create user accounts using SAS High-Performance Computing Management Console, the console propagates the public key across all the machines in the cluster in a single operation.

To create an operating system account and propagate the public key, follow these steps:

1. Make sure that the SAS High-Performance Computing Management Console server is running. While logged on as the root user, enter the following command from any directory:

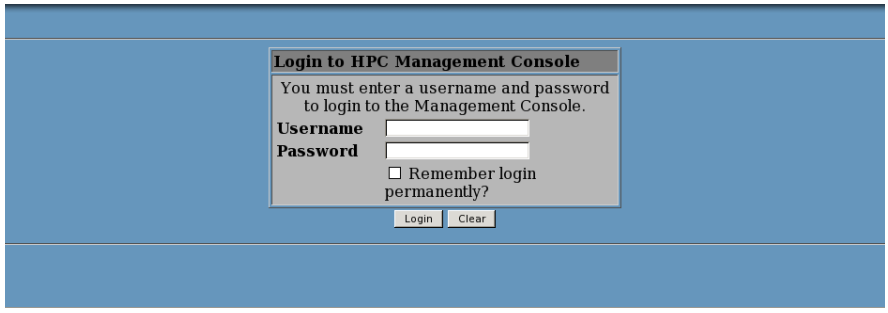
   ```
   service sashpcmc status
   ```

   (If you are logged on as a user other than the root user, the script returns the message `sashpcmc is stopped`.) For more information, see To start the SAS High-Performance Computing Management Console server on page 32.

2. Open a web browser and, in the address field, enter the fully qualified domain name for the blade 0 host followed by port 10020.

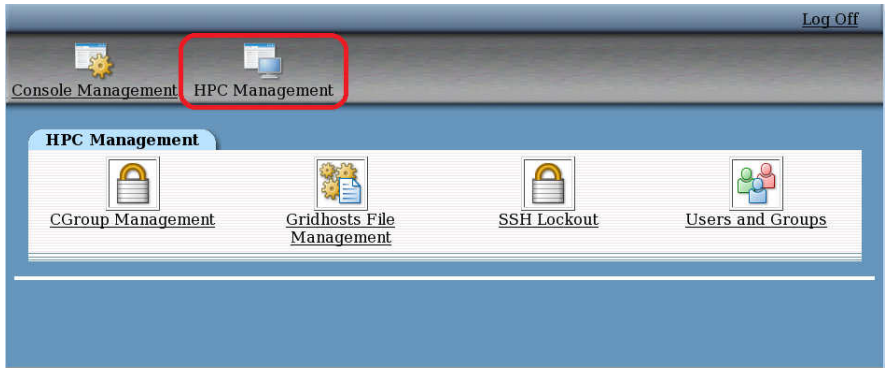
   For example: **`http://myserver.example.com:10020`**

The Login page appears.



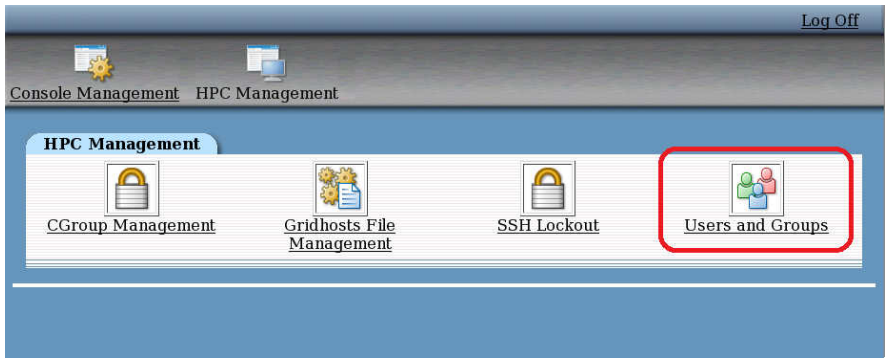
3. Log on to SAS High-Performance Computing Management Console.

   The Console Management page appears.



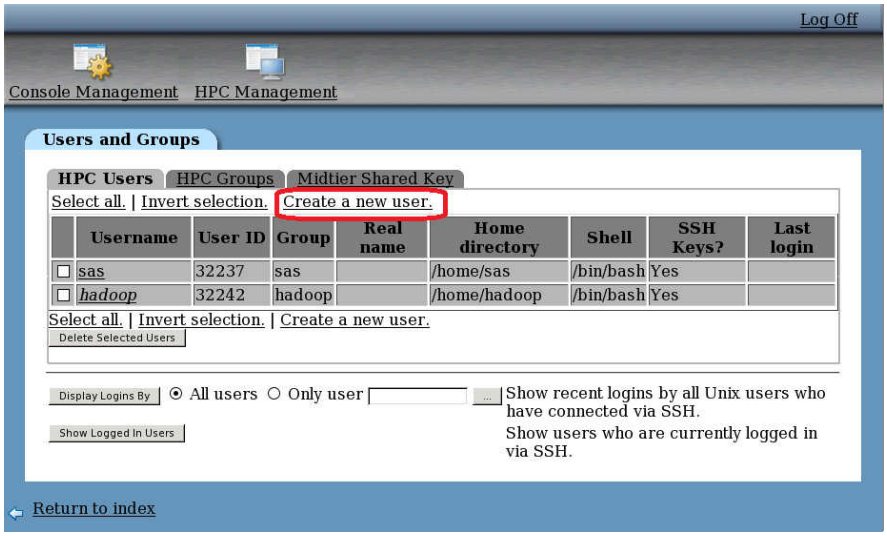
4. Click **HPC Management**.

   The HPC Management page appears.



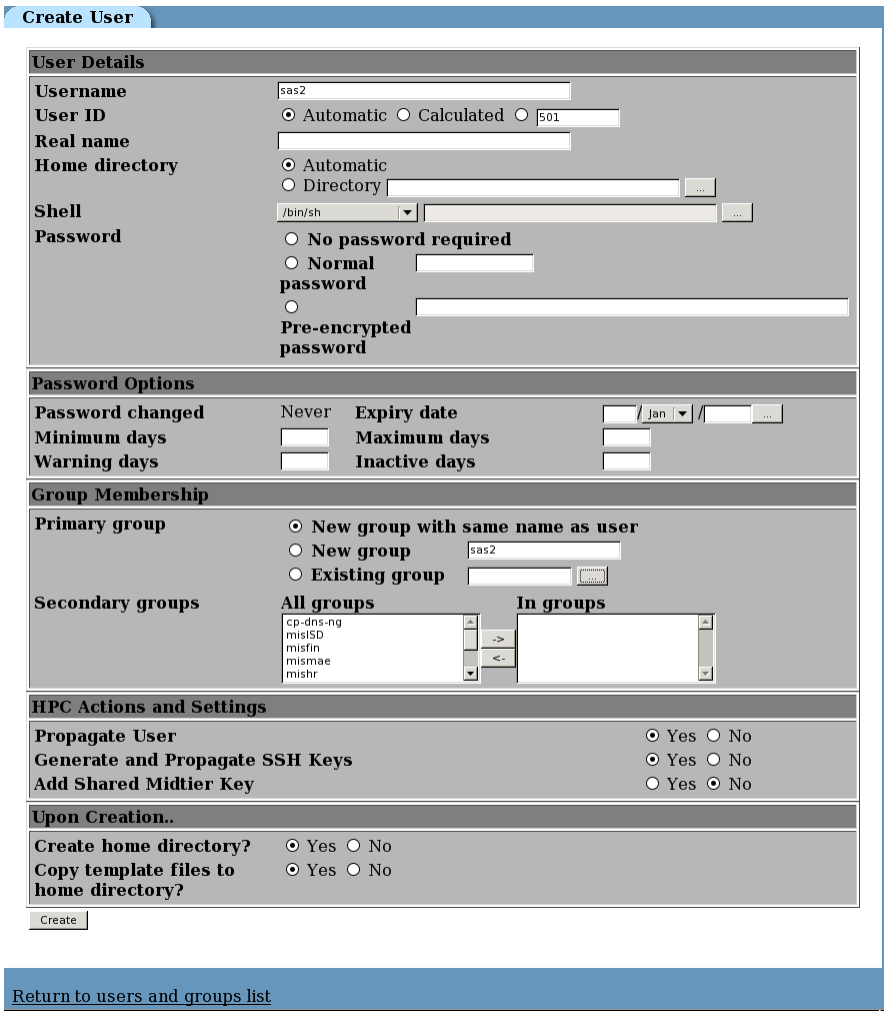
5. Click **Users and Groups**.

   The Users and Groups page appears.



6. Click **Create a new user**.

   The Create User page appears.



7. Enter information for the new user, using the security policies in place at your site.

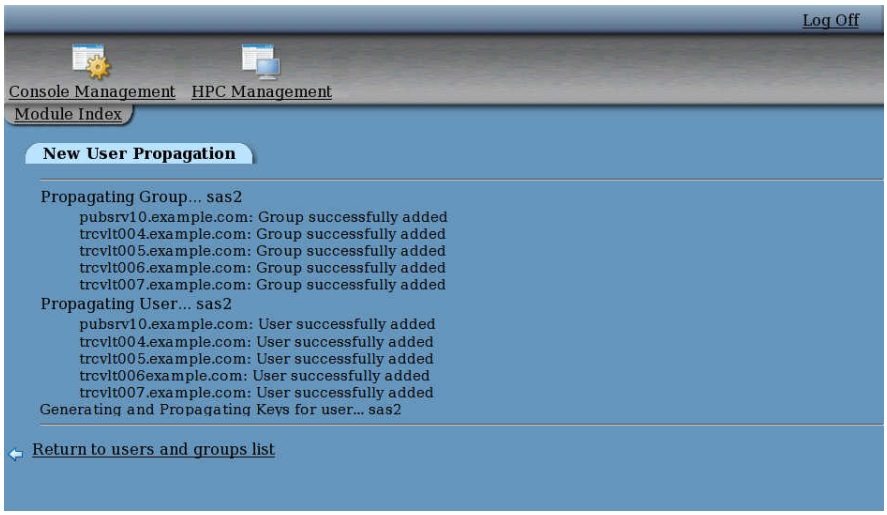
   Be sure to choose **Yes** for the following:

   - **Propagate User**

- **Generate and Propagate SSH Keys**

When you are finished making your selections, click **Create**.

The New User Propagation page appears and lists the status of the create user command. Your task is successful if you see output similar to the following figure.



Create the First User Account and Propagate the SSH Key

Depending on their configuration, some SAS solution users must have an operating system (external) account on all the machines in the cluster. Furthermore, the public key might be distributed on each cluster machine in order for their secure shell (SSH) access to operate properly. SAS High-Performance Computing Management Console enables you to perform these two tasks from one location.

To create an operating system account and propagate the public key for SSH, follow these steps:

1. Make sure that the SAS High-Performance Computing Management Console server is running. Enter the following command from any directory:

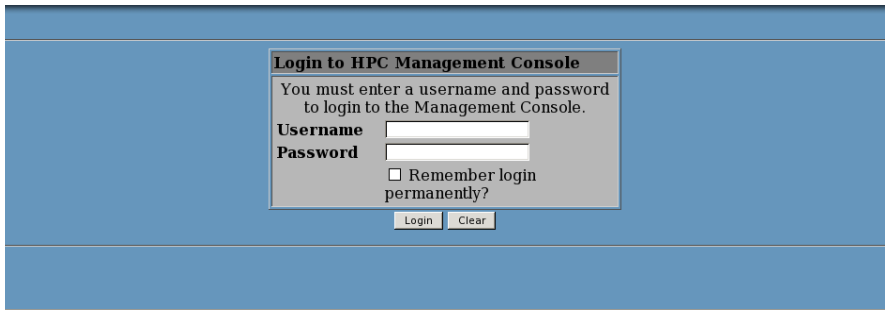
   ```
   service sashpcmc status
   ```

   For more information, see To start the SAS High-Performance Computing Management Console server on page 32.

2. Open a web browser and, in the address field, enter the fully qualified domain name for the blade 0 host followed by port 10020.

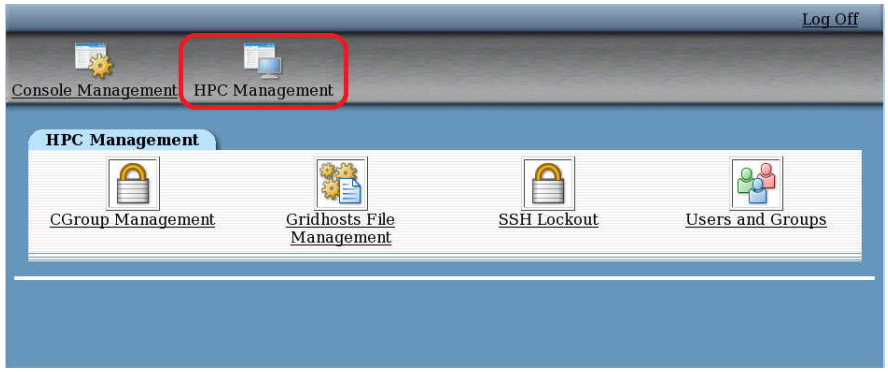
   For example: **`http://myserver.example.com:10020`**

   The Login page appears.



3. Log on to SAS High-Performance Computing Management Console.

   The Console Management page appears.

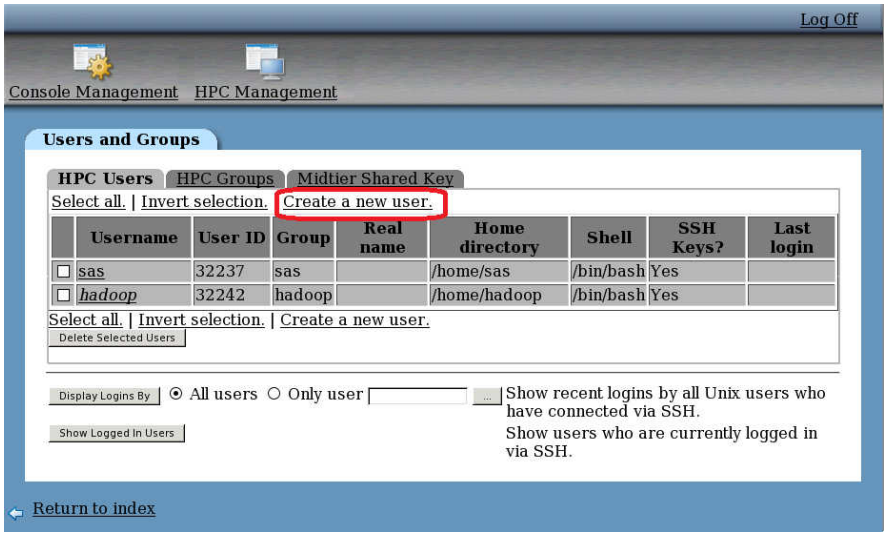


4. Click **HPC Management**.

   The Console Management page appears.

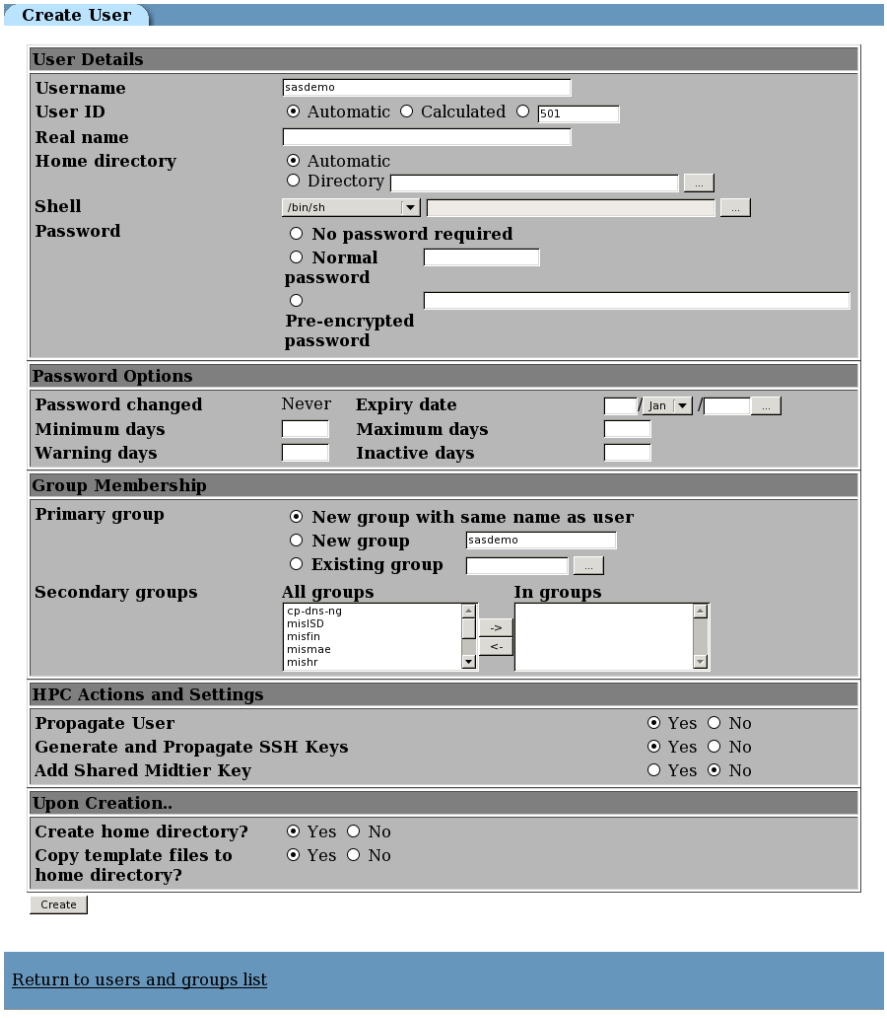
5. Click **Users and Groups**.

   The Users and Groups page appears.



6. Click **Create a new user**.

   The Create User page appears.



7. Enter information for the new user, using the security policies in place at your site.

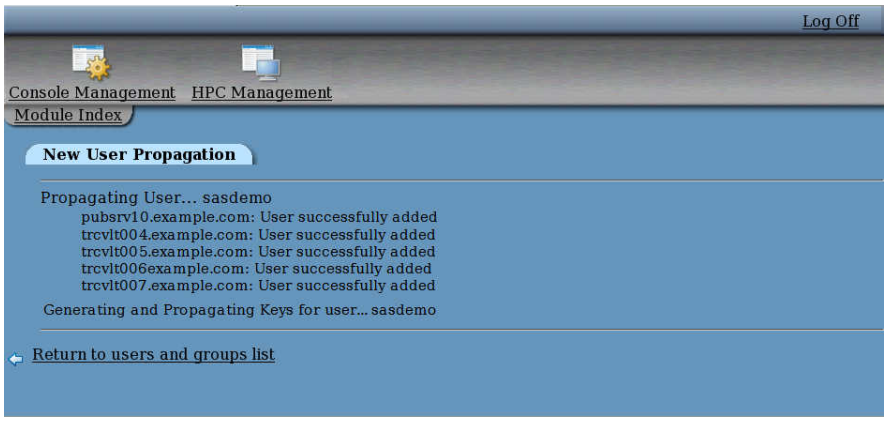
   Be sure to choose **Yes** for the following:

   - **Propagate User**

- **Generate and Propagate SSH Keys**

When you are finished making your selections, click **Create**.

The New User Propagation page appears and lists the status of the create user command. Your task is successful if you see output similar to the following figure.



Chapter 4

Deploying Hadoop

Infrastructure Deployment Process Overview

Installing and configuring SAS High-Performance Deployment of Hadoop is an optional fifth of seven steps required to install and configure the SAS High-Performance Analytics infrastructure.

1. Create a SAS Software Depot.
2. Check for documentation updates.
3. Prepare your analytics cluster.
4. (Optional) Deploy SAS High-Performance Computing Management Console.
5. **(Optional) Deploy Hadoop.**
6. Configure your data provider.
7. Deploy the SAS High-Performance Analytics environment.

Overview of Deploying Hadoop

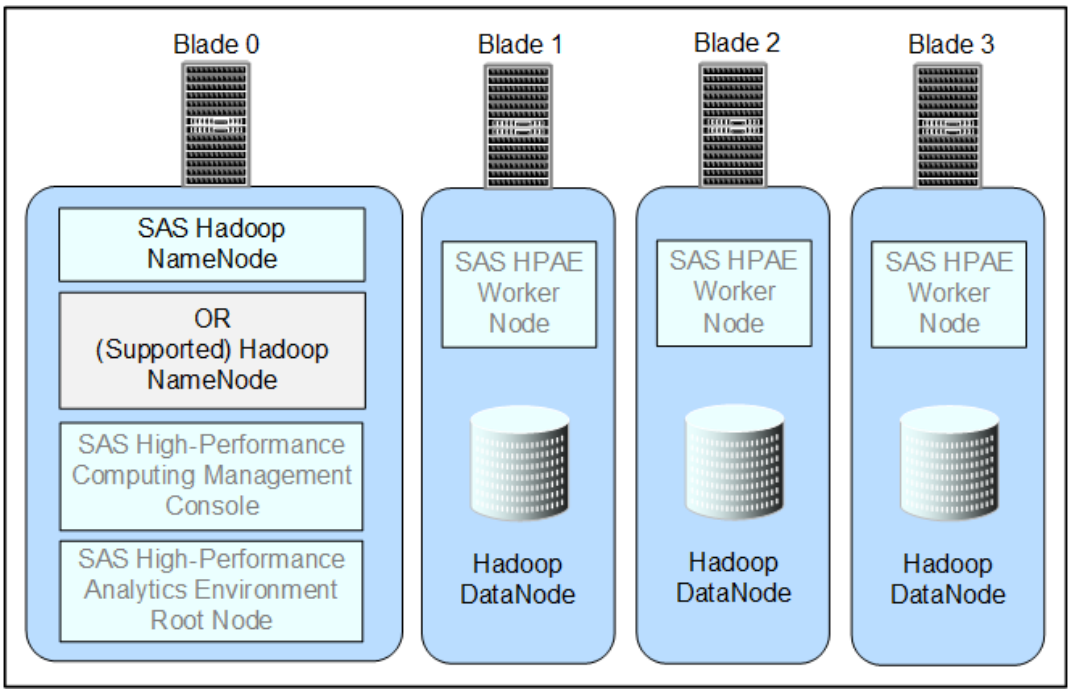
The SAS High-Performance Analytics environment relies on a massively parallel distributed database management system (Greenplum) or a Hadoop Distributed File System.

If you choose to use Hadoop, you have the option of using a Hadoop supplied by SAS, or using another supported Hadoop:

- "Deploying SAS High-Performance Deployment of Hadoop" on page 42.
- "Configuring Existing Hadoop Clusters" on page 55.

Deploying Hadoop requires installing and configuring components on the NameNode machine and DataNodes on the remaining machines in the cluster. In this document, the NameNode is deployed on blade 0.

Figure 4.1 *Analytics Cluster Co-located on the Hadoop Cluster*



Deploying SAS High-Performance Deployment of Hadoop

What Is SAS High-Performance Deployment of Hadoop?

Some solutions, such as SAS Visual Analytics, rely on a SAS data store that is co-located with the SAS High-Performance Analytic environment on the analytic cluster. One option for this co-located data store is the SAS High-Performance Deployment for

Hadoop. This is an Apache Hadoop distribution that is easily configured for use with the SAS High-Performance Analytics environment. It adds services to Apache Hadoop to write SASHDAT file blocks evenly across the HDFS filesystem. This even distribution provides a balanced workload across the machines in the cluster and enables SAS analytic processes to read SASHDAT tables at very impressive rates.

Alternatively, these SAS high-performance analytic solutions can use a pre-existing, supported Hadoop deployment or a Greenplum Data Computing Appliance.

Overview of Deploying SAS High-Performance Deployment of Hadoop

The following steps are required to deploy the SAS High-Performance Deployment of Hadoop:

- "Install SAS High-Performance Deployment of Hadoop" on page 43
- "(Optional) Deploy with Multiple Data Devices" on page 47
- "Format the Hadoop NameNode" on page 47
- "Start HDFS (with Kerberos)" on page 49 or "Start HDFS (without Kerberos)" on page 49
- "Check the HDFS Filesystem and Create HDFS Directories" on page 49
- "Validate Your Hadoop Deployment" on page 50

Install SAS High-Performance Deployment of Hadoop

The software that is needed for SAS High-Performance Deployment of Hadoop is available from within the SAS Software Depot that was created by the site depot administrator:

depot-installation-location`/standalone_installs/SAS_High_Performance_Deployment_for_Hadoop/2_6/Linux_for_x64/sashadoop.tar.gz`

1. Make sure that you have reviewed all of the information contained in the section "Preparing to Deploy Hadoop" on page 22.
2. Log on to the Hadoop NameNode machine (blade 0) as root.

   For more information, see "Install Hadoop Using root" on page 22.
3. Decide where to install Hadoop, and create that directory if it does not exist.

   ```
   mkdir hadoop
   ```
4. Record the name of this directory, as you will need it later in the install process.
5. Copy the sashadoop.tar.gz file to a temporary location and extract it:

   ```
   cp sashadoop.tar.gz /tmp
   cd /tmp
   tar xzf sashadoop.tar.gz
   ```

   A directory that is named **sashadoop** is created.
6. Change directory to the **sashadoop** directory and run the **hadoopInstall** command:

   ```
   cd sashadoop
   ./hadoopInstall
   ```

7. Respond to the prompts from the configuration program:

Table 4.1 *SAS High-Performance Deployment of Hadoop Configuration Parameters*

Parameter	Description
Do you wish to use an existing Hadoop installation? (y/N)	Specify **n** and press Enter to perform a new installation. If you are updating an earlier version of Hadoop, also answer **n** and press Enter here. Before proceeding, see "Updating SAS High-Performance Deployment of Hadoop".
Enter path to install Hadoop. The directory 'hadoop-2.4.0' will be created in the path specified.	Specify the directory that you created in Step 3 on page 43 and press Enter.
Do you wish to use Yarn and MR Jobhistory Server? (y/N)	Enter either **y** or **n** and press Enter. If you are using YARN, be sure to review "Preparing for YARN (Experimental)" on page 23 before proceeding.
Enter replication factor. Default 2	To accept the default, press Enter. Or specify a preferred number of replications for blocks (0 - 10) and press Enter. This prompt corresponds to the **dfs.replication** property for HDFS.
Enter port number for fs.defaultFS. Default 54310 Enter port number for dfs.namenode.https-address. Default 50470 Enter port number for dfs.datanode.https.address. Default 50475 Enter port number for dfs.datanode.address. Default 50010 Enter port number for dfs.datanode.ipc.address. Default 50020 Enter port number for dfs.namenode.http-address. Default 50070 Enter port number for dfs.datanode.http.address. Default 50075 Enter port number for dfs.secondary.http.address. Default 50090 Enter port number for dfs.namenode.backup.address. Default 50100 Enter port number for dfs.namenode.backup.http-address. Default 50105 Enter port number for com.sas.lasr.hadoop.service.namenode.port. Default 15452 Enter port number for com.sas.lasr.hadoop.service.datanode.port. Default 15453	To accept the default port numbers, press Enter for each prompt. Or specify a different port and press Enter. These ports are listed in "Pre-installation Ports Checklist for SAS" on page 26.

Parameter	Description
[The following port prompts are displayed when you choose to deploy YARN:] Enter port number for mapreduce.jobhistory.admin.address. Default 10033 Enter port number for mapreduce.jobhistory.webapp.address. Default 19888 Enter port number for mapreduce.jobhistory.address. Default 10021 Enter port number for yarn.resourcemanager.scheduler.address. Default 8030 Enter port number for yarn.resourcemanager.resource-tracker.address. Default 8031 Enter port number for yarn.resourcemanager.address. Default 8032 Enter port number for yarn.resourcemanager.admin.address. Default 8033 Enter port number for yarn.resourcemanager.webapp.address. Default 8088 Enter port number for yarn.nodemanager.localizer.address. Default 8040 Enter port number for yarn.nodemanager.webapp.address. Default 8042 Enter port number for yarn.web-proxy.address. Default 10022	To accept the default port numbers, press Enter for each prompt. Or specify a different port and press Enter. These ports are listed in "Pre-installation Ports Checklist for SAS" on page 26.
Enter maximum memory allocation per Yarn container. Default 5905	This is the maximum amount of memory (in MB) that YARN can allocate on a particular machine in the cluster. To accept the default, press Enter. Or specify a different value and press Enter.
Enter user that will be running the HDFS server process.	Specify the user name (for example, hdfs) and press Enter. For more information, see "User Accounts for Hadoop" on page 22.
Enter user that will be running Yarn services	Specify the user name (for example, yarn) and press Enter. For more information, see "Preparing for YARN (Experimental)" on page 23.
Enter user that will be running the Map Reduce Job History Server.	Specify the user name (for example, mapred) and press Enter. For more information, see "Preparing for YARN (Experimental)" on page 23.
Enter common primary group for users running Hadoop services.	Apache recommends that the hdfs, mapred, and yarn user accounts share the same primary Linux group (for example, hadoop). Enter a group name and press Enter. For more information, see "Preparing for YARN (Experimental)" on page 23.

Parameter	Description
Enter path for JAVA_HOME directory. (Default: /usr/lib/jvm/jre)	To accept the default, press Enter. Or specify a different path to the JRE or JDK and press Enter. *Note:* The configuration program does not verify that a JRE is installed at **`/usr/lib/jvm/jre`**, which is the default path for some Linux vendors.
Enter path for Hadoop data directory. This should be on a large drive. Default is '/hadoop/hadoop-data'. Enter path for Hadoop name directory. Default is '/hadoop/hadoop-name'.	To accept the default, press Enter. Or specify different paths and press Enter. *Note:* The data directory cannot be the root directory of a partition or mount. *Note:* If you have more than one data device, enter one of the data directories now, and after the installation, refer to "(Optional) Deploy with Multiple Data Devices" on page 47.
Enter full path to machine list. The NameNode '**`host`**' should be listed first.	Enter **`/etc/gridhosts`** and press Enter.

8. The installation program installs SAS High-Performance Deployment of Hadoop on the local host, configures several files, and then provides a prompt:

   ```
   The installer can now copy '/hadoop/hadoop-2.4.0' to all the slave machines
   using scp, skipping the first entry. Perform copy?  (YES/no)
   ```

   Enter **Yes** and press Enter to install SAS High-Performance Deployment of Hadoop on the other machines in the cluster.

   The installation program installs Hadoop. When you see output similar to the following, the installation is finished:

   ```
   Installation complete.

   HADOOP_HOME=/hadoop/hadoop-2.4.0

   Start HDFS as userid 'hdfs' with the following commands:
       Run this command to define HADOOP_HOME in hdfs's environment:
       export HADOOP_HOME=/hadoop/hadoop-2.4.0
       Run this command only if you want to format a new hadoop directory.
       It will erase existing data:
       $HADOOP_HOME/bin/hadoop namenode -format
       Run this command to start hdfs:
       $HADOOP_HOME/sbin/start-dfs.sh
       Run this command ONCE at initial install:
       $HADOOP_HOME/sbin/initial-sas-hdfs-setup.sh
   ```

9. Before you perform the steps described in the preceding output, there are configuration tasks that you might need to perform at this point:

   - If you want to store HDFS data on more than one device, see "(Optional) Deploy with Multiple Data Devices" on page 47.
   - If you want to use Kerberos for Secure Mode Hadoop, see "Post-Installation Configuration Changes to Hadoop for Kerberos" on page 50.

10. How you format the NameNode, start HDFS, and create the initial HDFS directories (the steps described in the preceding output) depends on whether you are using Kerberos. Details for the steps are provided in the following sections:

    a. "Format the Hadoop NameNode" (See page 47.)

    b. "Start HDFS (with Kerberos)" on page 49 or "Start HDFS (without Kerberos)" on page 49

    c. "Check the HDFS Filesystem and Create HDFS Directories" (See page 49.)

11. If your deployment includes SAS/ACCESS Interface to Hadoop, install the SAS Embedded Process on your Hadoop machine cluster. For more information, see Appendix 1, "Installing SAS Embedded Process for Hadoop," on page 89.

(Optional) Deploy with Multiple Data Devices

If you plan to use more than one data device with the SAS High-Performance Deployment of Hadoop, then you must manually declare each device's Hadoop data directory in hdfs-site.xml and push it out to all of your DataNodes.

To deploy SAS High-Performance Deployment for Hadoop with more than one data device, follow these steps:

1. Log on to the Hadoop NameNode using the account with which you plan to run Hadoop.
2. In a text editor, open ***hadoop-installation-directory*/etc/hadoop/hdfs-site.xml**.
3. Locate the **dfs.data.dir** property, specify the location of your additional data devices' data directories, and save the file.

   Separate multiple data directories with a comma.

   For example:

   ```
   <property>
     <name>dfs.data.dir</name>
     <value>/hadoop/hadoop-data,/data/dn</value>
   </property>
   ```

4. Copy hdfs-site.xml to all of your Hadoop DataNodes using the **simcp** command.

   For information about **simcp**, see Appendix 3, "SAS High-Performance Analytics Infrastructure Command Reference," on page 117.

5. If you are performing an initial installation, then return to the step that directed you to this procedure.

   If you have no other configuration tasks to perform and this is not an initial installation, then restart HDFS.

Format the Hadoop NameNode

To format the SAS High-Performance Deployment of Hadoop NameNode, follow these steps:

1. Change to the **hdfs** user account:

   ```
   su - hdfs
   ```

2. Export the HADOOP_HOME environment variable.

   For example:

   ```
   export "HADOOP_HOME=/hadoop/hadoop-2.4.0"
   ```

3. Format the NameNode:

   ```
   $HADOOP_HOME/bin/hadoop namenode -format
   ```

4. At the **`Re-format filesystem in /hadoop-install-dir/hadoop-name ? (Y or N)`** prompt, enter **`Y`**. A line similar to the following highlighted output indicates that the format is successful:

```
Formatting using clusterid: CID-5b96061a-79f4-4264-87e0-99f351b749af
14/06/12 17:17:02 INFO util.HostsFileReader:
Refreshing hosts (include/exclude) list
14/06/12 17:17:03 INFO blockmanagement.DatanodeManager:
dfs.block.invalidate.limit=1000
14/06/12 17:17:03 INFO util.GSet: VM type       = 64-bit
14/06/12 17:17:03 INFO util.GSet: 2% max memory = 19.33375 MB
14/06/12 17:17:03 INFO util.GSet: capacity      = 2^21 = 2097152 entries
14/06/12 17:17:03 INFO util.GSet: recommended=2097152, actual=2097152
14/06/12 17:17:03 INFO blockmanagement.BlockManager:
dfs.block.access.token.enable=false
14/06/12 17:17:03 INFO blockmanagement.BlockManager: defaultReplication = 2
14/06/12 17:17:03 INFO blockmanagement.BlockManager: maxReplication     = 512
14/06/12 17:17:03 INFO blockmanagement.BlockManager: minReplication     = 1
14/06/12 17:17:03 INFO blockmanagement.BlockManager:
maxReplicationStreams      = 2
14/06/12 17:17:03 INFO blockmanagement.BlockManager:
shouldCheckForEnoughRacks  = false
14/06/12 17:17:03 INFO blockmanagement.BlockManager:
replicationRecheckInterval = 3000
14/06/12 17:17:03 INFO namenode.FSNamesystem: fsOwner=nn/node1.domain.net@DOMAIN.NET (auth:KERBEROS)
14/06/12 17:17:03 INFO namenode.FSNamesystem: supergroup=supergroup
14/06/12 17:17:03 INFO namenode.FSNamesystem: isPermissionEnabled=true
14/06/12 17:17:03 INFO namenode.NameNode:
Caching file names occuring more than 10 times
14/06/12 17:17:04 INFO namenode.NNStorage: Storage directory
/hadoop/hadoop-name has been successfully formatted.
14/06/12 17:17:04 INFO namenode.FSImage: Saving image file
/hadoop/hadoop-name/current/fsimage.ckpt_0000000000000000000 using no compression
14/06/12 17:17:04 INFO namenode.FSImage: Image file of size 119 saved in 0 seconds.
14/06/12 17:17:04 INFO namenode.NNStorageRetentionManager:
Going to retain 1 images with txid >= 0
14/06/12 17:17:04 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at node1.domain.net/192.0.0.0
************************************************************/
```

Note: Without Kerberos, the log record for fsOwner is similar to the following:

```
14/06/12 17:17:03 INFO namenode.FSNamesystem: fsOwner=hdfs (auth:SIMPLE)
```

Start HDFS (with Kerberos)

To start HDFS using Kerberos, follow these steps:

1. Log on to the NameNode machine as the **hdfs** user and start the NameNode. For example:

   ```
   export HADOOP_HOME=/hadoop/hadoop-2.4.0
   $HADOOP_HOME/sbin/hadoop-daemon.sh start namenode
   ```

   This command starts the NameNode and the Secondary NameNode.

2. To start the DataNodes, log on to the first DataNode as the **root** user.

3. Run the following commands on each DataNode in the cluster:

   ```
   export HADOOP_HOME=/hadoop/hadoop-2.4.0
   $HADOOP_HOME/sbin/hadoop-daemon.sh start datanode
   ```

   You start the process with the **root** user, but it switches to the user ID specified in the HADOOP_SECURE_DN_USER variable from the hadoop-env.sh file.

   All secure DataNodes should be running.

Start HDFS (without Kerberos)

Log on to the NameNode machine as the **hdfs** user and start HDFS. For example:

```
export HADOOP_HOME=/hadoop/hadoop-2.4.0
$HADOOP_HOME/sbin/start-dfs.sh
```

This command starts the NameNode, Secondary NameNode, and the DataNodes in the cluster.

Check the HDFS Filesystem and Create HDFS Directories

To perform a filesystem check and create the initial HDFS directories, follow these steps:

1. Log on to the NameNode as the **hdfs** user.

   Note: If you are using Kerberos, then use kinit to get a ticket. For example: **kinit hdfs@DOMAIN.NET**.

2. Run the following commands to check the filesystem and display the number of DataNodes:

   ```
   export HADOOP_HOME=/hadoop/hadoop-2.4.0
   $HADOOP_HOME/bin/hadoop fsck /
   ```

3. Run the following command to create the directories in HDFS:

   ```
   $HADOOP_HOME/sbin/initial-sas-hdfs-setup.sh
   ```

4. Run the following command to verify the directories have been created:

   ```
   $HADOOP_HOME/bin/hadoop fs -ls /
   ```

   You should output similar to the following:

   ```
   drwxrwxrwx - hdfs supergroup 0 2014-07-24 13:29 /hps
   drwxrwxrwx - hdfs supergroup 0 2014-07-23 13:59 /test
   drwxrwxrwt - hdfs supergroup 0 2014-07-24 13:29 /tmp
   ```

```
drwxr-xr-x - hdfs supergroup 0 2014-07-24 13:29 /user
drwxrwxrwt - hdfs supergroup 0 2014-07-24 13:29 /vapublic
```

Validate Your Hadoop Deployment

You can confirm that Hadoop is running successfully by opening a browser to **http://*NameNode*:50070/dfshealth.jsp**. Review the information in the cluster summary section of the page. Confirm that the number of live nodes equals the number of DataNodes and that the number of dead nodes is zero.

Note: It can take a few seconds for each node to start. If you do not see every node, then refresh the connection in the web interface.

Post-Installation Configuration Changes to Hadoop for Kerberos

There are additional HDFS options not covered by the SAS Hadoop installer that need to be specified in order for Secure Mode Hadoop to work properly. Those additional options are defined in the various Hadoop configuration files. Your configuration files should match the ones below. You could copy and paste the files below and make environment-specific changes for the following items:

- hostnames
- JAVA_HOME
- HADOOP_HOME
- DOMAIN.NET is used as the example Kerberos realm

Note: Do not replace **_HOST**, as shown in the example files, with Kerberos principal names.

Be aware that you need to check and correct line breaks. Additions and changes relative to Secure Mode Hadoop are highlighted.

hadoop-env.sh:

```
export JAVA_HOME=/usr/java/latest
export HADOOP_HOME=/hadoop/hadoop-2.4.0
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export HADOOP_LOG_DIR=$HADOOP_HOME/logs/hdfs
for f in $HADOOP_HOME/contrib/capacity-scheduler/*.jar; do
  if [ "$HADOOP_CLASSPATH" ]; then
    export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$f
  else
    export HADOOP_CLASSPATH=$f
  fi
done

for f in $HADOOP_HOME/share/hadoop/sas/*.jar; do
  if [ "$HADOOP_CLASSPATH" ]; then
    export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:$f
  else
    export HADOOP_CLASSPATH=$f
  fi
done
```

```
export HADOOP_COMMON_LIB_NATIVE_DIR=${HADOOP_PREFIX}/lib/native
export HADOOP_OPTS="$HADOOP_OPTS -Djava.net.preferIPv4Stack=true
-Djava.library. path=$HADOOP_PREFIX/lib"
export HADOOP_NAMENODE_OPTS="-Dhadoop.security.logger=
${HADOOP_SECURITY_LOGGER:
- INFO,RFAS} -Dhdfs.audit.logger=${HDFS_AUDIT_LOGGER:
-INFO,NullAppender} $HADOOP_N AMENODE_OPTS"
export HADOOP_DATANODE_OPTS="-Dhadoop.security.logger=
ERROR,RFAS $HADOOP_DATANOD E_OPTS"
export HADOOP_SECONDARYNAMENODE_OPTS="-Dhadoop.security.logger=
${HADOOP_SECURITY _LOGGER:-INFO,RFAS} -Dhdfs.audit.logger=
${HDFS_AUDIT_LOGGER:-INFO,NullAppender} $HADOOP_SECONDARYNAMENODE_OPTS"
export HADOOP_CLIENT_OPTS="-Xmx512m $HADOOP_CLIENT_OPTS"
export HADOOP_SECURE_DN_USER=hdfs
export JSVC_HOME=/hadoop/hadoop-2.4.0/sbin
export HADOOP_SECURE_DN_LOG_DIR=${HADOOP_LOG_DIR}/${HADOOP_HDFS_USER}
export HADOOP_PID_DIR=/hadoop/hadoop-2.4.0/tmp
export HADOOP_SECURE_DN_PID_DIR=${HADOOP_PID_DIR}
export HADOOP_IDENT_STRING=$USER
```

core-site.xml:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!-- Put site-specific property overrides in this file. -->
<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://node1.domain.net:54310</value>
  </property>
  <property>
    <name>io.file.buffer.size</name>
    <value>102400</value>
  </property>
  <property>
    <name>hadoop.tmp.dir</name>
    <value>/hadoop/hadoop-2.4.0/tmp</value>
  </property>
  <property>
    <name>hadoop.security.authentication</name>
    <value>kerberos</value>
  </property>
  <property>
    <name>hadoop.security.authorization</name>
    <value>true</value>
  </property>
  <property>
    <name>hadoop.rpc.protection</name>
    <value>privacy</value>
  </property>
  <property>
    <name>hadoop.ssl.require.client.cert</name>
    <value>false</value>
  </property>
  <property>
    <name>hadoop.ssl.hostname.verifier</name>
    <value>DEFAULT</value>
```

```
  </property>
  <property>
    <name>hadoop.ssl.keystores.factory.class</name>
    <value>org.apache.hadoop.security.ssl.FileBasedKeyStoresFactory</value>
  </property>
  <property>
    <name>hadoop.ssl.server.conf</name>
    <value>ssl-server.xml</value>
  </property>
  <property>
    <name>hadoop.ssl.client.conf</name>
    <value>ssl-client.xml</value>
  </property>
  <property>
    <name>hadoop.security.auth_to_local</name>
   <value>
   RULE:[2:$1;$2](^dn;.*$)s/^.*$/hdfs/
   RULE:[2:$1;$2](^sn;.*$)s/^.*$/hdfs/
   RULE:[2:$1;$2](^nn;.*$)s/^.*$/hdfs/
   RULE:[1:$1@$0](.@DOMAIN.NET)s/@.//
   DEFAULT
   </value>
  </property>
</configuration>
```

hdfs-site.xml:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!-- Put site-specific property overrides in this file. -->
<configuration>
  <property>
    <name>dfs.replication</name>
    <value>2</value>
    <description>Default block replication.
    The actual number of replications can be specified when the file is created.
    The default is used if replication is not specified in create time.
    </description>
  </property>
  <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:///hadoop/hadoop-name</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>/hadoop/hadoop-data</value>
  </property>
  <property>
    <name>dfs.permissions.enabled</name>
    <value>true</value>
  </property>
  <property>
    <name>dfs.namenode.plugins</name>
    <value>com.sas.lasr.hadoop.NameNodeService</value>
  </property>
  <property>
    <name>dfs.datanode.plugins</name>
```

```
  <value>com.sas.lasr.hadoop.DataNodeService</value>
</property>
<property>
  <name>com.sas.lasr.hadoop.fileinfo</name>
  <value>ls -l {0}</value>
  <description>The command used to get the user, group, and permission
  information for a file.
  </description>
</property>
<property>
  <name>com.sas.lasr.service.allow.put</name>
  <value>true</value>
  <description>Flag indicating whether the PUT command is enabled when
  running as a service. The default is false.
  </description>
</property>
<property>
  <name>dfs.namenode.https-address</name>
  <value>0.0.0.0:50470</value>
</property>
<property>
  <name>dfs.datanode.https.address</name>
  <value>0.0.0.0:50475</value>
</property>
<property>
  <name>dfs.datanode.ipc.address</name>
  <value>0.0.0.0:50020</value>
</property>
<property>
  <name>dfs.namenode.http-address</name>
  <value>0.0.0.0:50070</value>
</property>
<property>
  <name>dfs.secondary.http.address</name>
  <value>0.0.0.0:50090</value>
</property>
<property>
  <name>dfs.namenode.backup.address</name>
  <value>0.0.0.0:50100</value>
</property>
<property>
  <name>dfs.namenode.backup.http-address</name>
  <value>0.0.0.0:50105</value>
</property>
<property>
  <name>com.sas.lasr.hadoop.service.namenode.port</name>
  <value>15452</value>
</property>
<property>
  <name>com.sas.lasr.hadoop.service.datanode.port</name>
  <value>15453</value>
</property>
<property>
  <name>dfs.namenode.fs-limits.min-block-size</name>
  <value>0</value>
</property>
```

```
  <property>
    <name>dfs.block.access.token.enable</name>
    <value>true</value>
  </property>
  <property>
    <name>dfs.http.policy</name>
    <value>HTTPS_ONLY</value>
  </property>
  <property>
    <name>dfs.namenode.keytab.file</name>
    <value>/hadoop/hadoop-2.4.0/etc/hadoop/hdfs_nnservice.keytab</value>
  </property>
  <property>
    <name>dfs.namenode.kerberos.principal</name>
    <value>nn/_HOST@DOMAIN.NET</value>
  </property>
  <property>
    <name>dfs.namenode.kerberos.https.principal</name>
    <value>host/_HOST@DOMAIN.NET</value>
  </property>
  <property>
    <name>dfs.secondary.namenode.https-port</name>
    <value>50471</value>
  </property>
  <property>
    <name>dfs.secondary.namenode.keytab.file</name>
    <value>/hadoop/hadoop-2.4.0/etc/hadoop/hdfs_nnservice.keytab</value>
  </property>
  <property>
    <name>dfs.secondary.namenode.kerberos.principal</name>
    <value>sn/_HOST@DOMAIN.NET</value>
  </property>
  <property>
    <name>dfs.secondary.namenode.kerberos.https.principal</name>
    <value>host/_HOST@DOMAIN.NET</value>
  </property>
  <property>
    <name>dfs.datanode.data.dir.perm</name>
    <value>700</value>
  </property>
  <property>
    <name>dfs.datanode.address</name>
    <value>0.0.0.0:1004</value>
  </property>
  <property>
    <name>dfs.datanode.http.address</name>
    <value>0.0.0.0:1006</value>
  </property>
  <property>
    <name>dfs.datanode.keytab.file</name>
    <value>/hadoop/hadoop-2.4.0/etc/hadoop/hdfs_dnservice.keytab</value>
  </property>
  <property>
    <name>dfs.datanode.kerberos.principal</name>
    <value>dn/_HOST@DOMAIN.NET</value>
  </property>
```

```
  <property>
    <name>dfs.datanode.kerberos.https.principal</name>
    <value>host/_HOST@DOMAIN.NET</value>
  </property>
  <property>
    <name>dfs.encrypt.data.transfer</name>
    <value>true</value>
  </property>
  <property>
    <name>dfs.webhdfs.enabled</name>
    <value>true</value>
  </property>
  <property>
    <name>dfs.web.authentication.kerberos.principal</name>
    <value>HTTP/_HOST@DOMAIN.NET</value>
  </property>
  <property>
    <name>dfs.web.authentication.kerberos.keytab</name>
    <value>/hadoop/hadoop-2.4.0/etc/hadoop/http.keytab</value>
  </property>
</configuration>
```

Configuring Existing Hadoop Clusters

Overview of Configuring Existing Hadoop Clusters

If your site uses a Hadoop implementation that is supported, then you can configure your Hadoop cluster for use with the SAS High-Performance Analytics environment.

The following steps are needed to configure your existing Hadoop cluster:

1. Make sure that your Hadoop deployment meets the analytic environment prerequisites. For more information, see "Prerequisites for Existing Hadoop Clusters" on page 55
2. Follow steps specific to your implementation of Hadoop:
   - "Configuring the Existing Cloudera Hadoop Cluster" on page 56
   - "Configuring the Existing Hortonworks Data Platform Hadoop Cluster" on page 60
   - "Configuring the Existing IBM BigInsights Hadoop Cluster" on page 61
   - "Configuring the Existing Pivotal HD Hadoop Cluster" on page 63

Prerequisites for Existing Hadoop Clusters

The following is required for existing Hadoop clusters that will be configured for use with the SAS High-Performance Analytics environment:

- Each machine machine in the cluster must be able to resolve the host name of all the other machines.
- The NameNode and secondary NameNode are not defined as the same host.

- The NameNode host does not also have a DataNode configured on it.
- For Kerberos, in the SAS High-Performance Analytics environment, **/etc/hosts** must contain the machine names in the cluster in this order: short name, fully qualified domain name.
- Time must be synchronized across all machines in the cluster.
- (Cloudera 5 only) Make sure that all machines configured for the SAS High-Performance Analytics environment are in the same role group.

Configuring the Existing Cloudera Hadoop Cluster

Managing Cloudera Configuration Priorities

Cloudera uses the Linux **alternatives** command for client configuration files. Therefore, make sure that the client configuration path has the highest priority for all machines in the cluster. (Often, the mapreduce client configuration has a higher priority over the hdfs configuration.)

If the output of the command **alternatives -display hadoop-conf** returns the Cloudera server configuration, or if mapreduce client configuration has priority over the client configuration, you will experience problems because SAS makes additions to the client configuration. For more information about **alternatives**, refer to its man page.

Configure the Existing Cloudera Hadoop Cluster, Version 5

Use the Cloudera Manager to configure your existing Cloudera 5 Hadoop deployment to interoperate with the SAS High-Performance Analytics environment.

1. Untar the SAS High-Performance Deployment for Hadoop tarball, and propagate three files (identified in the following steps) on every machine in your Cloudera Hadoop cluster:

   a. Navigate to the SAS High-Performance Deployment for Hadoop tarball in your SAS Software depot:

   ```
   cd depot-installation-location/standalone_installs/
   SAS_High_Performance_Deployment_for_Hadoop/2_6/Linux_for_x64/
   ```

   b. Copy sashadoop.tar.gz to a temporary location where you have Write access.

   c. Untar sashadoop.tar.gz:

   ```
   tar xzf sashadoop.tar.gz
   ```

   d. Locate sas.lasr.jar and sas.lasr.hadoop.jar and propagate these two JAR files to every machine in the Cloudera Hadoop cluster into the CDH library path.

   TIP If you have already installed the SAS High-Performance Computing Management Console or the SAS High-Performance Analytics environment, you can issue a single **simcp** command to propagate JAR files across all machines in the cluster. For example:

   ```
   /opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.jar
   /opt/cloudera/parcels/CDH-5.0.0-0.cdh5b1.p0.57/lib/hadoop/lib/
   /opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.hadoop.jar
   /opt/cloudera/parcels/CDH-5.0.0-0.cdh5b1.p0.57/lib/hadoop/lib/
   ```

   For more information, see Appendix 3, “SAS High-Performance Analytics Infrastructure Command Reference,” on page 117.

e. Locate saslasrfd and propagate this file to every machine in the Cloudera Hadoop cluster into the CDH **bin** directory. For example:

```
/opt/sashpcmc/opt/webmin/utilbin/simcp saslasrfd
/opt/cloudera/parcels/CDH-5.0.0-0.cdh5b1.p0.57/lib/hadoop/bin/
```

2. Log on to the Cloudera Manager as an administrator.
3. Add the following to the plug-in configuration for the NameNode:

```
com.sas.lasr.hadoop.NameNodeService
```

4. Add the following to the plug-in configuration for DataNodes:

```
com.sas.lasr.hadoop.DataNodeService
```

5. Add the following lines to the advanced configuration for service-wide. These lines are placed in the HDFS Service Advanced Configuration Snippet (Safety Valve) for hdfs-site.xml:

```
<property>
<name>com.sas.lasr.service.allow.put</name>
<value>true</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.datanode.port</name>
<value>15453</value>
</property>
<property>
<name> dfs.namenode.fs-limits.min-block-size</name>
<value>0</value>
</property>
```

6. Add the following property to the **HDFS Client Configuration Safety Valve** under **Advanced** within the **Gateway Default Group**. Make sure that you change *path-to-data-dir* to the data directory location for your site (for example, **<value>file://dfs/dn</value>**):

```
<property>
<name>com.sas.lasr.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.datanode.port</name>
<value>15453</value>
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file://path-to-data-dir</value>
</property>
```

7. Add the location of JAVA_HOME to the **HDFS Client Environment Advanced Configuration Snippet for hadoop-env.sh (Safety Valve)**, located under **Advanced** in the **Gateway Default Group**. For example:

```
JAVA_HOME=/usr/lib/java/jdk1.7.0_07
```

8. Save your changes and deploy the client configuration to each host in the cluster.

9. Restart the HDFS service and any dependencies in Cloudera Manager.
10. If needed, set the following environment variables before running the Hadoop commands.

```
export JAVA_HOME=/path-to-java
export HADOOP_HOME=/opt/cloudera/parcels/CDH-5.0.0-0.cdh5b1.p0.57/lib/hadoop
```

11. Create and mode the **/test** directory in HDFS for testing the cluster with SAS test jobs. You might need to set HADOOP_HOME first, and you must run the following commands as the user running HDFS (typically, hdfs).

```
$HADOOP_HOME/bin/hadoop fs -mkdir /test
```

```
$HADOOP_HOME/bin/hadoop fs -chmod 777 /test
```

12. Make sure that the client configuration path has the highest priority for all machines in the cluster. For more information, see "Managing Cloudera Configuration Priorities" on page 56.

Configure the Existing Cloudera Hadoop Cluster, Version 4

Use the Cloudera Manager to configure your existing Cloudera 4 Hadoop deployment to interoperate with the SAS High-Performance Analytics environment.

TIP In Cloudera 4.2 and earlier, you must install the enterprise license, even if you are below the stated limit of 50 nodes in the Hadoop cluster for requiring a license.

1. Untar the SAS High-Performance Deployment for Hadoop tarball, and propagate three files (identified in the following steps) on every machine in your Cloudera Hadoop cluster:
   a. Navigate to the SAS High-Performance Deployment for Hadoop tarball in your SAS Software depot:

      ```
      cd depot-installation-location/standalone_installs/
      SAS_High_Performance_Deployment_for_Hadoop/2_6/Linux_for_x64/
      ```

   b. Copy sashadoop.tar.gz to a temporary location where you have Write access.
   c. Untar sashadoop.tar.gz:

      ```
      tar xzf sashadoop.tar.gz
      ```

   d. Locate sas.lasr.jar and sas.lasr.hadoop.jar and propagate these two JAR files to every machine in the Cloudera Hadoop cluster into the CDH library path.

      TIP If you have already installed the SAS High-Performance Computing Management Console or the SAS High-Performance Analytics environment, you can issue a single **simcp** command to propagate JAR files across all machines in the cluster. For example:

      ```
      /opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.jar
      /opt/cloudera/parcels/CDH-4.4.0-1.cdh4.4.0.p0.39/lib/hadoop/lib/
      /opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.hadoop.jar
      /opt/cloudera/parcels/CDH-4.4.0-1.cdh4.4.0.p0.39/lib/hadoop/lib/
      ```

      For more information, see Appendix 3, "SAS High-Performance Analytics Infrastructure Command Reference," on page 117.

   e. Locate saslasrfd and propagate this file to every machine in the Cloudera Hadoop cluster into the CDH **bin** directory. For example:

```
/opt/sashpcmc/opt/webmin/utilbin/simcp saslasrfd
/opt/cloudera/parcels/CDH-4.4.0-1.cdh4.4.0.p0.39/lib/hadoop/bin/
```

2. Log on to the Cloudera Manager as an administrator.
3. Add the following to the plug-in configuration for the NameNode:

```
com.sas.lasr.hadoop.NameNodeService
```

4. Add the following to the plug-in configuration for DataNodes:

```
com.sas.lasr.hadoop.DataNodeService
```

5. Add the following lines to the advanced configuration for service-wide. These lines are placed in the **HDFS Service Configuration Safety Valve** property for hdfs-site-xml:

```
<property>
<name>com.sas.lasr.service.allow.put</name>
<value>true</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.datanode.port</name>
<value>15453</value>
</property>
<property>
<name> dfs.namenode.fs-limits.min-block-size</name>
<value>0</value>
</property>
```

6. Restart all Cloudera Manager services.
7. Create and set the mode for the **`/test`** directory in HDFS for testing. You might need to set HADOOP_HOME first, and you must run the following commands as the user running HDFS (normally, the hdfs user).
8. If needed, set the following environment variables before running the Hadoop commands.

```
export HADOOP_HOME=/opt/cloudera/parcels/CDH-4.4.0-1.cdh4.4.0.p0.39/lib/hadoop
```

9. Run the following commands to create the **`/test`** directory in HDFS. This directory is to be used for testing the cluster with SAS test jobs.

```
$HADOOP_HOME/bin/hadoop fs -mkdir /test
```

```
$HADOOP_HOME/bin/hadoop fs -chmod 777 /test
```

10. Add the following to the **HDFS Client Configuration Safety Valve**:

```
<property>
<name>com.sas.lasr.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.datanode.port</name>
<value>15453</value>
```

```
</property>
<property>
<name>dfs.datanode.data.dir</name>
<value>file:///hadoop/hadoop-data</value>
</property>
```

11. Add the location of JAVA_HOME to the **Client Environment Safety Valve** for hadoop-env.sh. For example:

```
JAVA_HOME=/usr/lib/java/jdk1.7.0_07
```

12. Save your changes and deploy the client configuration to each host in the cluster.
13. Make sure that the client configuration path has the highest priority for all machines in the cluster. For more information, see "Managing Cloudera Configuration Priorities" on page 56.

TIP Remember the value of HADOOP_HOME as the SAS High-Performance Analytics environment prompts for this during its install. By default, these are the values for Cloudera:

- Cloudera 4.5:

  /opt/cloudera/parcels/CDH-4.5.0-1.cdh4.5.0.p0.30

- Cloudera 4.2 and earlier:

  /opt/cloudra/parcels/CDH-4.2.0-1.cdh4.2.0.po.10/lib/Hadoop

Configuring the Existing Hortonworks Data Platform Hadoop Cluster

Use the Ambari interface to configure your existing Hortonworks Data Platform deployment to interoperate with the SAS High-Performance Analytics environment.

1. Log on to Ambari as an administrator, and stop all HDP services.
2. Untar the SAS High-Performance Deployment for Hadoop tarball, and propagate three files (identified in the following steps) on every machine in your Hortonworks Hadoop cluster:

   a. Navigate to the SAS High-Performance Deployment for Hadoop tarball in your SAS Software depot:

   ```
   cd depot-installation-location/standalone_installs/
   SAS_High_Performance_Deployment_for_Hadoop/2_6/Linux_for_x64/
   ```

   b. Copy sashadoop.tar.gz to a temporary location where you have Write access.

   c. Untar sashadoop.tar.gz:

   ```
   tar xzf sashadoop.tar.gz
   ```

   d. Locate sas.lasr.jar and sas.lasr.hadoop.jar and propagate these two JAR files to every machine in the HDP cluster into the HDP library path.

   TIP If you have already installed the SAS High-Performance Computing Management Console or the SAS High-Performance Analytics environment, you can issue a single **simcp** command to propagate JAR files across all machines in the cluster. For example:

   ```
   /opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.jar /usr/lib/hadoop/lib/
   /opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.hadoop.jar /usr/lib/hadoop/lib/
   ```

For more information, see Appendix 3, "SAS High-Performance Analytics Infrastructure Command Reference," on page 117.

e. Locate saslasrfd and propagate this file to every machine in the HDP cluster into the HDP **`bin`** directory. For example:

```
/opt/sashpcmc/opt/webmin/utilbin/simcp saslasrfd /usr/lib/hadoop/bin/
```

3. In the Ambari interface, create a custom hdfs-site.xml and add the following properties:

dfs.namenode.plugins
com.sas.lasr.hadoop.NameNodeService

dfs.datanode.plugins
com.sas.lasr.hadoop.DataNodeService

com.sas.lasr.hadoop.fileinfo
ls -l {0}

com.sas.lasr.service.allow.put
true

com.sas.lasr.hadoop.service.namenode.port
15452

com.sas.lasr.hadoop.service.datanode.port
15453

dfs.namenode.fs-limits.minblock.size
0

4. Save the properties and start the HDFS service.
5. Run the following commands as the hdfs user to create the **`/test`** directory in HDFS. This directory is used for testing your cluster with SAS test jobs.

```
hadoop fs -mkdir /test
```

```
hadoop fs -chmod 777 /test
```

Configuring the Existing IBM BigInsights Hadoop Cluster

To configure your existing IBM BigInsights Hadoop deployment to interoperate with the SAS High-Performance Analytics environment, follow these steps:

1. Untar the SAS High-Performance Deployment for Hadoop tarball, and propagate three files (identified in the following steps) on every machine in your BigInsights Hadoop cluster:

   a. Navigate to the SAS High-Performance Deployment for Hadoop tarball in your SAS Software depot:

   ```
   cd depot-installation-location/standalone_installs/
   SAS_High_Performance_Deployment_for_Hadoop/2_6/Linux_for_x64/
   ```

   b. Copy sashadoop.tar.gz to a temporary location where you have Write access.

   c. Untar sashadoop.tar.gz:

   ```
   tar xzf sashadoop.tar.gz
   ```

   d. Locate sas.lasr.jar and sas.lasr.hadoop.jar and propagate these two JAR files to every machine in the BigInsights cluster into the library path.

Note: HADOOP_HOME: default location: **/opt/ibm/biginsights/IHC**.
BIGINSIGHT_HOME: default location: **/opt/ibm/biginsights**.

TIP If you have already installed the SAS High-Performance Computing Management Console or the SAS High-Performance Analytics environment, you can issue a single **simcp** command to propagate JAR files across all machines in the cluster. For example:

```
/opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.jar
$HADOOP_HOME/share/hadoop/hdfs/libs
/opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.hadoop.jar
$HADOOP_HOME/share/hadoop/hdfs/libs
```

For more information, see Appendix 3, "SAS High-Performance Analytics Infrastructure Command Reference," on page 117.

e. Locate saslasrfd and propagate this file to every machine in the BigInsights cluster into the **$HADOOP_HOME/bin** directory. For example:

```
/opt/sashpcmc/opt/webmin/utilbin/simcp saslasrfd $HADOOP_HOME/bin
```

2. On the machine where you initially installed BigInsights, add the following properties for SAS for the HDFS configuration to the file **$BIGINSIGHT_HOME/hdm/hadoop-conf-staging/hdfs-site.xml**. Adjust values appropriately for your deployment:

```
<property>
<name>dfs.datanode.plugins</name>
<value>com.sas.lasr.hadoop.DataNodeService</value>
</property>
<property>
<name>com.sas.lasr.service.allow.put</name>
<value>true</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.datanode.port</name>
<value>15453</value>
</property>
<property>
<name> dfs.namenode.fs-limits.min-block-size</name>
<value>0</value>
</property>
```

3. Synchronize this new configuration by running the following command on the machine where you initially deployed BigInsights: **$BIGINSIGHT_HOME/bin/synconf.sh**.

4. On the machine where you initially deployed BigInsights, log on as the biadmin user and run the following commands to restart the cluster with the new configuration:

   stop-all.sh

   start-all.sh

5. Note the location of HADOOP_HOME. You will need to refer to this value when installing the SAS High-Performance Analytics environment.

6. Run the following commands as the hdfs user to create the **/test** directory in HDFS. This directory is used for testing your cluster with SAS test jobs.

```
hadoop fs -mkdir /test

hadoop fs -chmod 777 /test
```

Configuring the Existing Pivotal HD Hadoop Cluster

Use the Pivotal Command Center (PCC) to configure your existing Pivotal HD deployment to interoperate with the SAS High-Performance Analytics environment.

1. Log on to PCC as gpadmin. (The default password is gpadmin.)
2. Untar the SAS High-Performance Deployment for Hadoop tarball, and propagate three files (identified in the following steps) on every machine in your Cloudera Hadoop cluster:
   a. Navigate to the SAS High-Performance Deployment for Hadoop tarball in your SAS Software depot:

      ```
      cd depot-installation-location/standalone_installs/
      SAS_High_Performance_Deployment_for_Hadoop/2_6/Linux_for_x64/
      ```

   b. Copy sashadoop.tar.gz to a temporary location where you have Write access.
   c. Untar sashadoop.tar.gz:

      ```
      tar xzf sashadoop.tar.gz
      ```

   d. Locate sas.lasr.jar and sas.lasr.hadoop.jar and propagate these two JAR files to every machine in the Pivotal HD cluster into the library path.

      TIP If you have already installed the SAS High-Performance Computing Management Console or the SAS High-Performance Analytics environment, you can issue a single **simcp** command to propagate JAR files across all machines in the cluster. For example:

```
/opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.jar /usr/lib/gphd/hadoop/lib/
/opt/sashpcmc/opt/webmin/utilbin/simcp sas.lasr.hadoop.jar
/usr/lib/gphd/hadoop/lib/
```

      For more information, see Appendix 3, “SAS High-Performance Analytics Infrastructure Command Reference,” on page 117.

   e. Locate saslasrfd and propagate this file to every machine in the Pivotal HD cluster into the Pivotal HD **bin** directory. For example:

      ```
      /opt/sashpcmc/opt/webmin/utilbin/simcp saslasrfd /usr/lib/gphd/hadoop/bin/
      ```

3. In the PCC, for YARN, make sure that Resource Manager, History Server, and Node Managers have unique host names.
4. In the PCC, make sure that the Zookeeper Server contains a unique host name.
5. Add the following properties for SAS for the HDFS configuration to the file hdfs-site.xml:

```
<property>
<name>dfs.datanode.plugins</name>
<value>com.sas.lasr.hadoop.DataNodeService</value>
</property>
<property>
<name>com.sas.lasr.service.allow.put</name>
```

```
<value>true</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.lasr.hadoop.service.datanode.port</name>
<value>15453</value>
</property>
<property>
<name> dfs.namenode.fs-limits.min-block-size</name>
<value>0</value>
</property>
```

6. Save your changes and deploy.
7. Restart your cluster using PCC and verify that HDFS is running in the dashboard.
8. Run the following commands as the gpadmin user to create the **`/test`** directory in HDFS. This directory is used for testing your cluster with SAS test jobs.

```
hadoop fs -mkdir /test

hadoop fs -chmod 777 /test
```

Chapter 5

Configuring Your Data Provider

Infrastructure Deployment Process Overview

Configuring your data storage is the sixth of seven steps for deploying the SAS High-Performance Analytics infrastructure.

1. Create a SAS Software Depot.
2. Check for documentation updates.
3. Prepare your analytics cluster.
4. (Optional) Deploy SAS High-Performance Computing Management Console.
5. (Optional) Deploy Hadoop.
6. **⇨ 6. Configure your data provider.**
7. Deploy the SAS High-Performance Analytics environment.

Overview of Configuring Your Data Provider

The SAS High-Performance Analytics environment relies on a massively parallel distributed database management system or a Hadoop Distributed File System.

The topics that follow describe how you configure the data sources that you are using with the analytics environment:

- "Recommended Database Names" on page 69
- "Preparing the Greenplum Database for SAS" on page 70
- "Overview of Preparing Your Data Provider for a Parallel Connection with SAS" on page 71

The figures that follow illustrate the various ways in which you can configure data access for the analytics environment:

- Analytics cluster co-located on the Hadoop cluster or Greenplum data appliance on page 67
- Analytics cluster remote from your data store (serial connection) on page 68
- Analytics cluster remote from your data store (parallel Connection) on page 69

Figure 5.1 *Analytics Cluster Co-Located on the Hadoop Cluster or Greenplum Data Appliance*

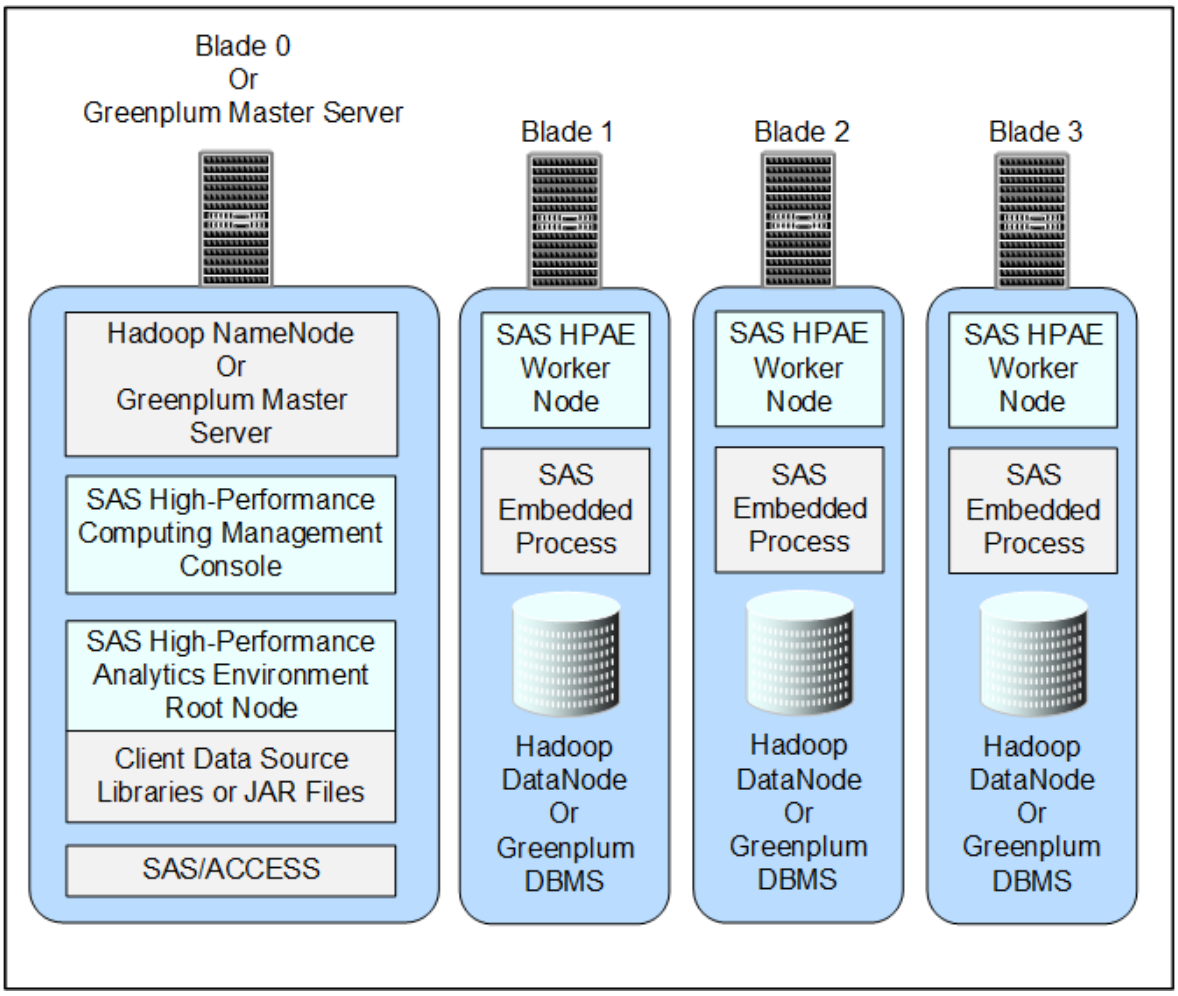


Figure 5.2 *Analytics Cluster Remote from Your Data Store (Serial Connection)*

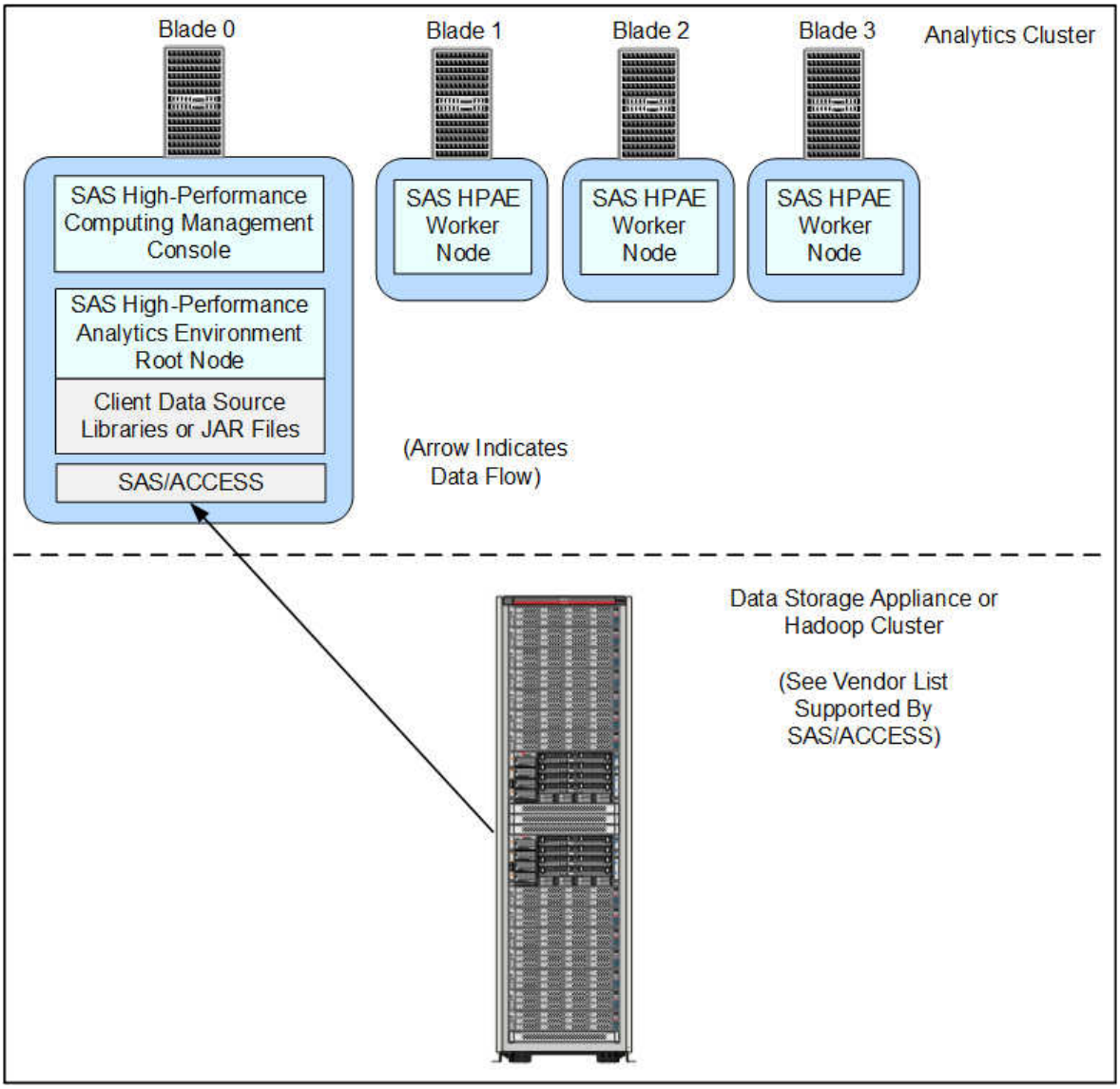
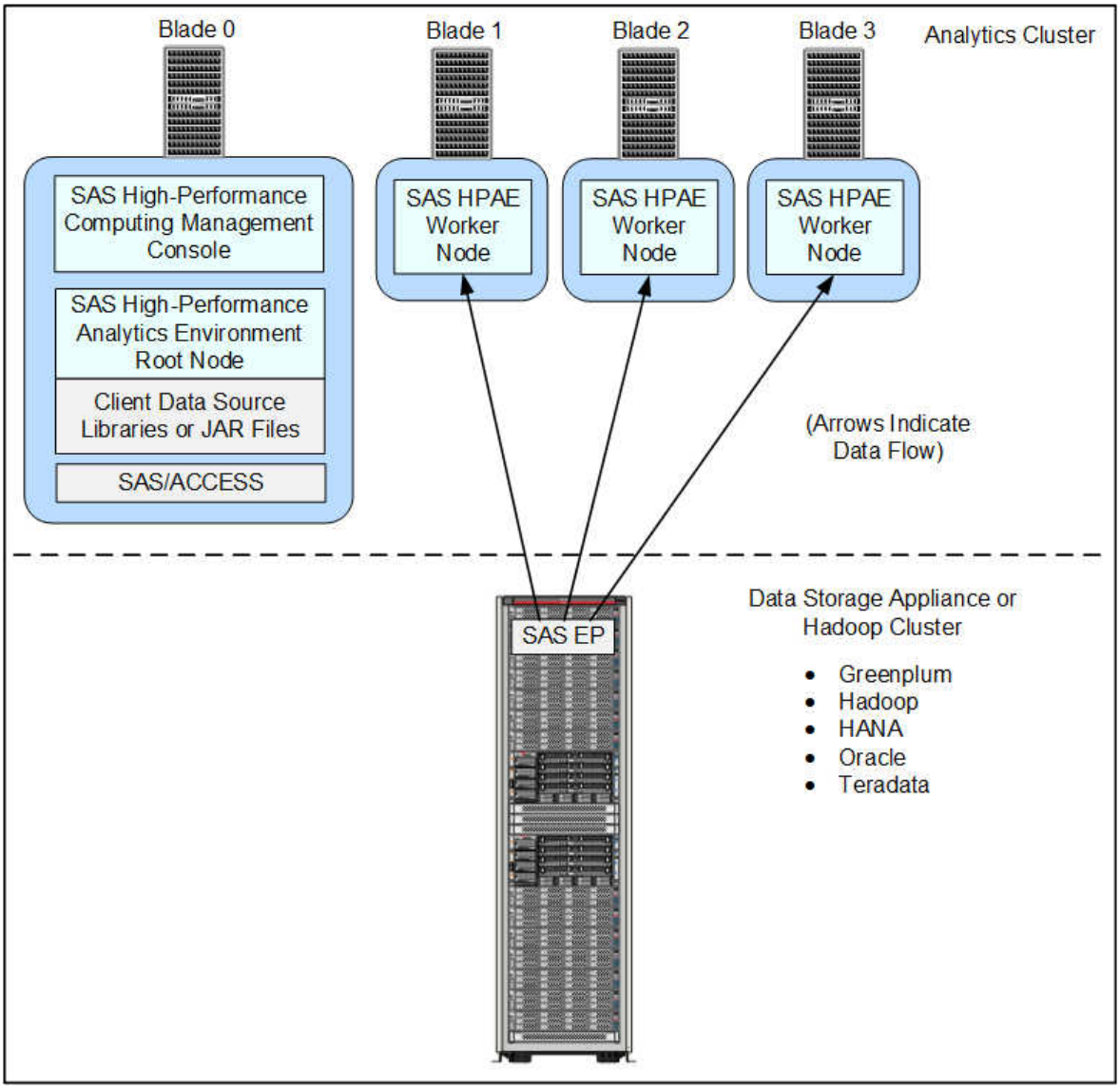


Figure 5.3 *Analytics Cluster Remote from Your Data Store (Parallel Connection)*



Recommended Database Names

SAS solutions, such as SAS Visual Analytics, that rely on a co-located data provider can make use of two database instances.

The first instance often already exists and is expected to have your operational or transactional data that you want to explore and analyze.

A second database instance is used to support the self-service data access features of SAS Visual Analytics. This database is commonly named “vapublic,” but you can specify a different name if you prefer. Keep these names handy, as the SAS Deployment Wizard prompts you for them when deploying your SAS solution.

Preparing the Greenplum Database for SAS

Overview of Preparing the Greenplum Database for SAS

The steps required to configure your Greenplum database for the SAS High-Performance Analytics environment consist of the following:

1. Associate users with a group role.
2. Configure the SAS/ACCESS Interface to Greenplum.
3. Install the SAS Embedded Process for Greenplum.

Recommendations for Greenplum Database Roles

If multiple users access the SAS High-Performance Analytics environment on the Greenplum database, it is recommended that you set up a group role and associate the database roles for individual users with the group. The Greenplum database administrator can then associate access to the environment at the group level.

The following is one example of how you might accomplish this.

1. First, create the group.

   For example:

   ```
   CREATE GROUP sas_cust_group NOLOGIN;
   ALTER ROLE sas_cust_group CREATEEXTTABLE;
   ```

   Note: Remember that in Greenplum, only object privileges are inheritable. When granting the CREATEEXTTABLE, you are granting a system privilege. You can grant CREATEEXTTABLE to a group role, but the role must use a set role as a role group first.

2. For each user, create a database role and associate it with the group.

   For example:

   ```
   CREATE ROLE megan  LOGIN IN ROLE sas_cust_group PASSWORD 'megan';
   CREATE ROLE calvin LOGIN IN ROLE sas_cust_group PASSWORD 'calvin';
   ```

3. If a resource queue exists, associate the roles with the queue.

   For example:

   ```
   CREATE RESOURCE QUEUE sas_cust_queue WITH
                   (MIN_COST=10000.0   ,
                   ACTIVE_STATEMENTS=20,
                   PRIORITY=HIGH       ,
                   MEMORY_LIMIT='4GB'  );

   ALTER ROLE megan  RESOURCE QUEUE sas_cust_queue;
   ALTER ROLE calvin RESOURCE QUEUE sas_cust_queue;
   ```

4. Finally, grant the database roles rights on the schema where the SAS Embedded Process has been published. For more information, see *SAS In-Database Products: Administrator's Guide*, available at **http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf**.

For example:

```
GRANT ALL ON SCHEMA SASLIB TO sas_cust_group;
```

Configure the SAS/ACCESS Interface to Greenplum Software

SAS solutions, such as SAS High-Performance Analytics Server, rely on SAS/ACCESS to communicate with the Greenplum Data Computing Appliance.

When you deploy the SAS/ACCESS Interface to Greenplum, make sure that the following configuration steps are performed:

1. Set the ODBCHOME environment variable to your ODBC home directory.
2. Set the ODBCINI environment variable to the location and name of your odbc.ini file.

   TIP You can set both the ODBCHOME and ODBCINI environment variables in the SAS sasenv_local file and affect all executions of SAS. For more information, see *SAS Intelligence Platform: Data Administration Guide*, available at **http://support.sas.com//dcumentation/cdl/en/bidsag/65041/PDF/default/bidsag.pdf**.
3. Include the Greenplum ODBC drivers in your shared library path (LD_LIBRARY_PATH).
4. Edit odbc.ini and odbcinst.ini following the instructions listed in the *Configuration Guide for SAS Foundation for UNIX Environments*, available at **http://support.sas.com/documentation/installcenter/en/ikfdtnunxcg/66380/PDF/default/config.pdf**

Install the SAS Embedded Process for Greenplum

If you have not done so already, install the appropriate SAS Embedded Process on your Greenplum data appliance. For more information, see *SAS In-Database Products: Administrator's Guide*, available at **http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf**.

Note: While following the instructions in the *SAS In-Database Products: Administrator's Guide*, there is no need to run the %INDGP_PUBLISH_COMPILEUDF macro. All the other steps, including running the %INDGP_PUBLISH_COMPILEUDF_EP macro, are required.

Preparing Your Data Provider for a Parallel Connection with SAS

Overview of Preparing Your Data Provider for a Parallel Connection with SAS

Before you can configure the SAS High-Performance Analytics environment to use the SAS Embedded Process for a parallel connection with your data store, you must locate particular JAR files and gather particular information about your data provider. If you

are using a Hadoop not supplied by SAS, then you must also complete a few configuration steps.

From the following list, choose the topic for your respective data provider:

1. “Prepare for Hadoop” on page 72 .
2. “Prepare for a Greenplum Data Computing Appliance” on page 72.
3. “Prepare for a HANA Cluster” on page 73.
4. “Prepare for an Oracle Exadata Appliance” on page 73.
5. “Prepare for a Teradata Managed Server Cabinet” on page 74.

Prepare for Hadoop

Before you can configure the SAS High-Performance Analytics environment to use the SAS Embedded Process for a parallel connection with your Hadoop data store, there are certain requirements that must be met.

1. Record the path to the Hadoop JAR files required by SAS in the table that follows:

   Table 5.1 *Record the Location of the Hadoop JAR Files Required by SAS*

Example	Actual Path of the Required Hadoop JAR Files on Your System
/opt/hadoop_jars (common and core JAR files)	
/opt/hadoop_jars/MR1 (Map Reduce JAR files)	
/opt/hadoop_jars/MR2 (Map Reduce JAR files)	

2. Record the location (JAVA_HOME) of the 64-bit Java Runtime Engine (JRE) on your Hadoop clusterin the table that follows:

   Table 5.2 *Record the Location of the JRE*

Example	Actual Path of the JRE on Your System
/opt/java/jre1.7.0_07	

Prepare for a Greenplum Data Computing Appliance

Before you can configure the SAS High-Performance Analytics environment to use the SAS Embedded Process for a parallel connection with your Greenplum Data Computing Appliance, there are certain requirements that must be met.

1. Install the Greenplum client on the Greenplum Master Server (blade 0) in your analytics cluster.

   For more information, refer to your Greenplum documentation.

2. Record the path to the Greenplum client in the table that follows:

   Table 5.3 *Record the Location of the Greenplum Client*

Example	Actual Path of the Greenplum Client on Your System
/usr/local/greenplum-db	

Prepare for a HANA Cluster

Before you can configure the SAS High-Performance Analytics environment to use the SAS Embedded Process for a parallel connection with your HANA cluster, there are certain requirements that must be met.

1. Install the HANA client on blade 0 in your analytics cluster.

   For more information, refer to your HANA documentation.

2. Record the path to the HANA client in the table that follows:

   Table 5.4 *Record the Location of the HANA Client*

Example	Actual Path of the HANA Client on Your System
/usr/local/lib/hdbclient	

Prepare for an Oracle Exadata Appliance

Before you can configure the SAS High-Performance Analytics environment to use the SAS Embedded Process for a parallel connection with your Oracle Exadata appliance, there are certain requirements that must be met.

1. Install the Oracle client on blade 0 in your analytics cluster.

   For more information, refer to your Oracle documentation.

2. Record the path to the Oracle client in the table that follows. (This should be the absolute path to libclntsh.so):

   Table 5.5 *Record the Location of the Oracle Client*

Example	Actual Path of the Oracle Client on Your System
/usr/local/ora11gr2/product/11.2.0/cli 1/lib	

3. Record the value of the Oracle TNS_ADMIN environment variable in the table that follows. (Typically, this is the directory that contains the tnsnames.ora file):

Table 5.6 *Record the Value of the Oracle TNS_ADMIN Environment Variable*

Example	Oracle TNS_ADMIN Environment Variable Value on Your System
/my_server/oracle	

Prepare for a Teradata Managed Server Cabinet

Before you can configure the SAS High-Performance Analytics environment to use the SAS Embedded Process for a parallel connection with your Teradata Managed Server Cabinet, there are certain requirements that must be met.

1. Install the Teradata client on blade 0 in your analytics cluster.

   For more information, refer to your Teradata documentation.

2. Record the path to the Teradata client in the table that follows. (This should be the absolute path to the directory that contains the `odbc_64` subdirectory):

Table 5.7 *Record the Location of the Teradata Client*

Example	Actual Location of the Teradata Client on Your System
/opt/teradata/client/13.10	

Chapter 6

Deploying the SAS High-Performance Analytics Environment

Infrastructure Deployment Process Overview

Installing and configuring the SAS High-Performance Analytics environment is the last of seven steps.

1. Create a SAS Software Depot.
2. Check for documentation updates.
3. Prepare your analytics cluster.
4. (Optional) Deploy SAS High-Performance Computing Management Console.
5. (Optional) Deploy Hadoop.
6. Configure your data provider.

⇨ **7. Deploy the SAS High-Performance Analytics environment.**

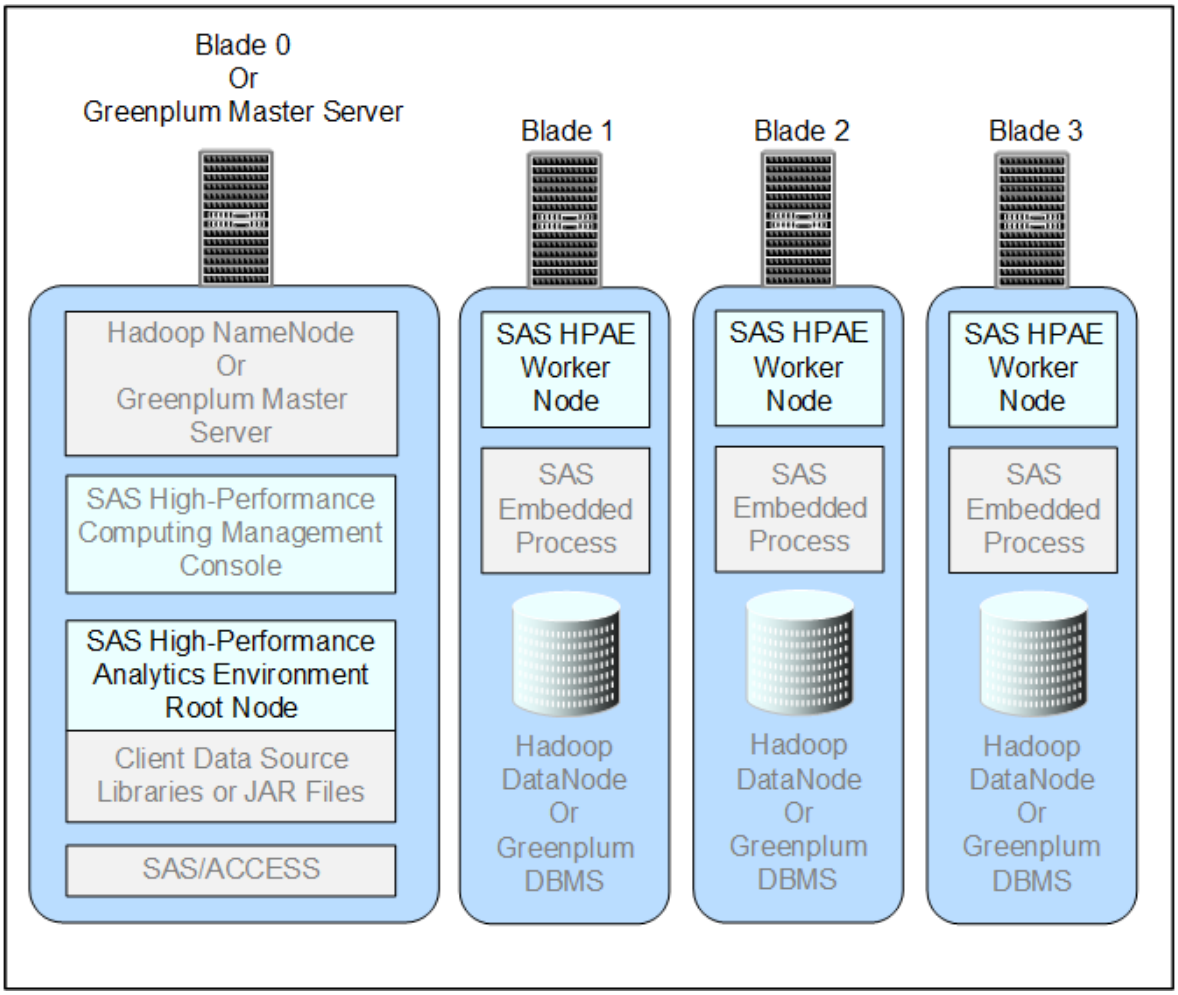
This chapter describes how to install and configure all of the components for the SAS High-Performance Analytics environment on the machines in the cluster.

Overview of Deploying the Analytics Environment

Deploying the SAS High-Performance Analytics environment requires installing and configuring components on the root node machine and on the remaining machines in the cluster. In this document, the root node is deployed on blade 0 (Hadoop) or on the Master Server (Greenplum).

The following figure shows the SAS High-Performance Analytics environment co-located on your Hadoop cluster or Greenplum data appliance:

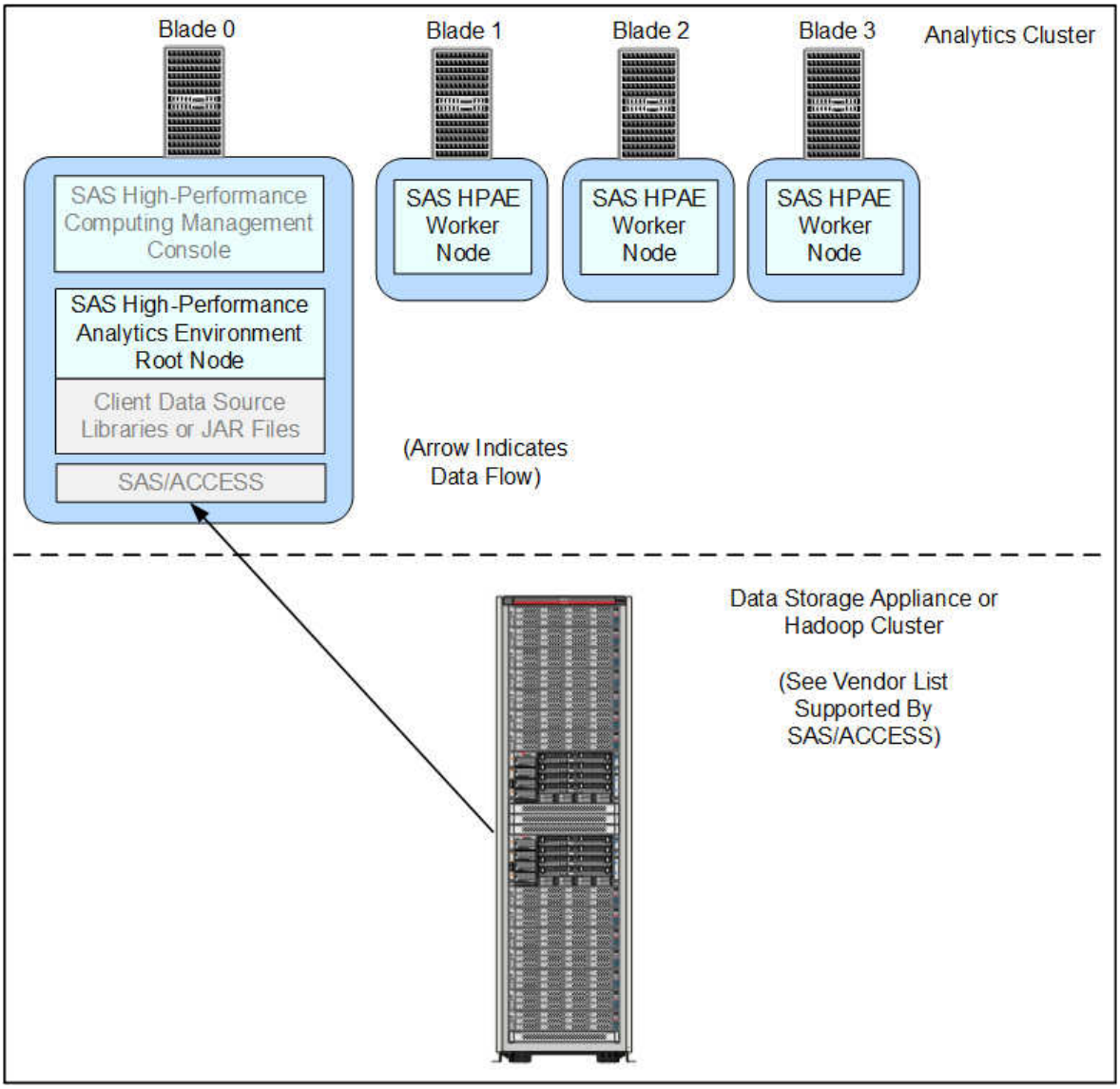
Figure 6.1 *Analytics Environment Co-Located on the Hadoop Cluster or Greenplum Data Appliance*



Note: For deployments that use Hadoop for the co-located data provider and access SASHDAT tables exclusively, SAS/ACCESS and SAS Embedded Process are not needed.

The following figure shows the SAS High-Performance Analytics environment using a serial connection through the SAS/ACCESS Interface to your remote data store:

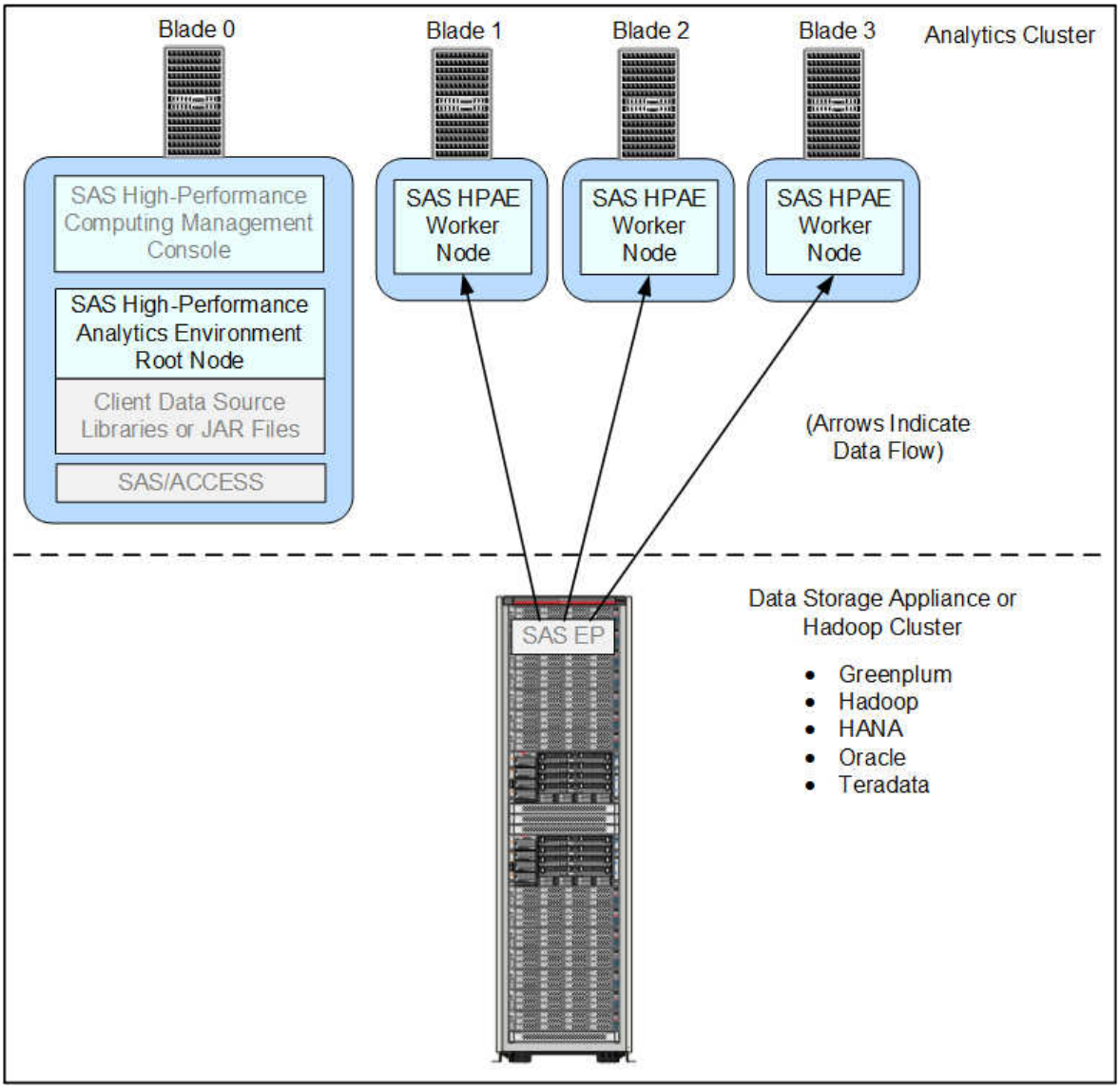
Figure 6.2 *Analytics Environment Remote from Your Data Store (Serial Connection)*



TIP There might be solution-specific criteria that you should consider when determining your analytics cluster location. For more information, see the installation or administration guide for your specific SAS solution.

The following figure shows the SAS High-Performance Analytics environment using a parallel connection through the SAS Embedded Process to your remote data store:

Figure 6.3 *Analytics Environment Remote from Your Data Store (Parallel Connection)*



Install the Analytics Environment

The SAS High-Performance Analytics environment components are installed with two shell scripts. Follow these steps to install:

1. Make sure that you have reviewed all of the information contained in the section "Preparing to Deploy the SAS High-Performance Analytics Environment" on page 25.
2. The software that is needed for the SAS High-Performance Analytics environment is available from within the SAS Software Depot that was created by the site depot

administrator: *depot-installation-location*`/standalone_installs/SAS_High-Performance_Node_Installation/2_9/Linux_for_x64`.

3. Copy the file that is appropriate for your operating system to the `/tmp` directory of the root node of the cluster:
   - Red Hat Linux (pre-version 6) and SUSE Linux 10:

     TKGrid_Linux_x86_64_rhel5.sh
   - Red Hat Linux 6 and other equivalent, kernel-level Linux systems:

     TKGrid_Linux_x86_64.sh
4. Copy TKTGDat.sh to the `/tmp` directory of the root node of the cluster.

   Note: TKTGDat.sh contains the SAS linguistic binary files required to perform text analysis in SAS LASR Analytic Server with SAS Visual Analytics and to run PROC HPTMINE and HPTMSCORE with SAS Text Miner.
5. Log on to the machine that will serve as the root node of the cluster or the data appliance with a user account that has the necessary permissions.

   For more information, see "User Accounts for the SAS High-Performance Analytics Environment" on page 25.
6. Change directories to the desired installation location, such as `/opt`.

   Record the location of where you installed the analytics environment, as other configuration programs will prompt you for this path later in the deployment process.
7. Run the TKGrid shell script in this directory.

   The shell script creates the `TKGrid` subdirectory and places all files under that directory.
8. Respond to the prompts from the shell script:

Table 6.1 *Configuration Parameters for the TKGrid Shell Script*

Parameter	Description
Shared install or replicate to each node? (Y=SHARED/ n=replicated)	If you are installing to a local drive on each node, then specify `n` and press Enter to indicate that this is a replicated installation. If you are installing to a drive that is shared across all the nodes (for example, NFS), then specify `y` and press Enter.
Enter additional paths to include in LD_LIBRARY_PATH, separated by colons (:)	If you have any external library paths that you want to be accessible to the SAS High-Performance Analytics environment, enter the paths here and press Enter. Otherwise, press Enter.
Enter remote process launcher command (Default is ssh).	If you want to use another program other than SSH to start processes on remote nodes, specify the program's absolute installation path and press Enter. Otherwise, press Enter.

Parameter	Description
Enter additional options to mpirun.	If you have any **mpirun** options to add, specify them and press Enter. If you are using Kerberos, specify the following option and press Enter: `-genvlist `env | sed -e s/=.*/,/ | sed /KRB5CCNAME/d | tr -d '\n'`TKPATH,LD_LIBRARY_PATH` If you have no additional options, press Enter.
Enter path to use for Utility files. (default is /tmp).	SAS High-Performance Analytics applications might write scratch files. By default, these files are created in the **/tmp** directory. To accept the default, press Enter. Or, to redirect the files to a different location, specify the path and press Enter. *Note:* If the directory that you specified does not exist, you must create it manually.
Enter path to Hadoop. (default is Hadoop not installed).	If your site uses Hadoop, enter the installation directory (the value of the variable, HADOOP_HOME) and press Enter. If your site does not use Hadoop, press Enter. If you are using SAS High-Performance Deployment of Hadoop, use the directory that you specified earlier in Step 3 on page 43.
Force Root Rank to run on headnode? (y/N)	If the appliance resides behind a firewall and only the root node can connect back to the client machines, specify **y** and press Enter. Otherwise, specify **n** and press Enter.
Enter full path to machine list. The head node '**head-node-machine-name**' should be listed first.	Specify the name of the file that you created in the section "List the Machines in the Cluster or Appliance" (for example, **/etc/gridhosts**) and press Enter.
Enter maximum runtime for grid jobs (in seconds). Default 7200 (2 hours).	If a SAS High-Performance Analytics application executes for more than the maximum allowable run time, it is automatically terminated. You can adjust that run-time limit here. To accept the default, press Enter. Or, specify a different maximum run time (in seconds) and press Enter.
Enter value for UMASK. (default is unset.)	To set *no* umask value, press Enter. Or, specify a umask value and press Enter. For more information, see "Consider Umask Settings" on page 25.

9. If you selected a replicated installation at the first prompt, you are now prompted to choose the technique for distributing the contents to the appliance nodes:

```
The install can now copy this directory to all the machines
listed in 'filename' using scp, skipping the first entry.
Perform copy?
(YES/no)
```

Press Enter if you want the installation program to perform the replication. Enter **no** if you are distributing the contents of the installation directory by some other technique.

10. Next, in the same directory from which you ran the TKGrid shell script, run TKTGDat.sh.

    The shell script creates the **TKTGDat** subdirectory and places all files in that directory.

11. Respond to the prompts from the shell script:

Table 6.2 *Configuration Prompts for the TKTG Dat Shell Script*

Shared install or replicate to each node? (Y=SHARED/ n=replicated)	If you are installing to a local drive on each node, then specify **n** and press Enter to indicate that this is a replicated installation. If you are installing to a drive that is shared across all the nodes (for example, NFS), then specify **y** and press Enter.
Enter full path to machine list.	Specify the name of the file that you created in the section "List the Machines in the Cluster or Appliance" (for example, **/etc/gridhosts**) and press Enter.

12. If you selected a replicated installation at the first prompt, you are now prompted to choose the technique for distributing the contents to the appliance nodes:

    ```
    The install can now copy this directory to all the machines
    listed in 'filename' using scp, skipping the first entry.
    Perform copy? (YES/no)
    ```

    If you want the installation program to perform the replication, specify **yes** and press Enter. If you are distributing the contents of the installation directory by some other technique, specify **no** and press Enter.

13. Proceed to "Validating the Analytics Environment Deployment" on page 85.

Configuring for Access to a Data Store with a SAS Embedded Process

Overview of Configuring for Access to a Data Store with a SAS Embedded Process

The process involved for configuring the SAS High-Performance Analytics environment for a remote data store consists of the following steps:

1. Prepare for the data provider that the analytics environment will query.

   For more information, see "Preparing Your Data Provider for a Parallel Connection with SAS" on page 71.

2. Review the considerations for configuring the analytics environment for use with a remote data store.

   For more information, see "How the Configuration Script Works" on page 82.

3. Configure the analytics environment for a remote data store.

   For more information, see "Configure for Access to a Data Store with a SAS Embedded Process" on page 82.

How the Configuration Script Works

You configure the SAS High-Performance Analytics environment for a remote data store using a shell script. The script enables you to configure the environment for the various third-party data stores supported by the SAS Embedded Process.

The Analytics environment is designed on the principle, install once, configure many. For example, suppose that your site has three remote data stores from three different third-party vendors whose data you want to analyze. You run the analytics environment configuration script one time and provide the information for each data store vendor as you are prompted for it. (When prompted for a data store vendor that you do not have, simply ignore that set of prompts.)

When you have different versions of the same vendor's data store, specifying the vendor's *latest* client data libraries usually works. However, this choice can be problematic for different versions of Hadoop, where a later set of JAR files is not typically backwardly compatible with earlier versions, or for sites that use Hadoop implementations from more than one vendor. (The configuration script does not delineate between different Hadoop vendors.) In these situations, you must run the analytics environment configuration script once for each different Hadoop version or vendor. As the configuration script creates a **TKGrid_REP** directory underneath the current directory, it is important to run the script a second time from a different directory.

To illustrate how you might manage configuring the analytics environment for two different Hadoop vendors, consider this example: suppose your site uses Cloudera Hadoop 4 and Hortonworks Data Platform 2. When running the analytics environment script to configure for Cloudera 4, you would create a directory similar to:

```
cdh4
```

When configuring the analytics environment for Cloudera, you would run the script from the **cdh4** directory. When complete, the script creates a **TKGrid_REP** child directory:

```
cdh4/TKGrid_REP
```

For Hortonworks, you would create a directory similar to:

```
hdp2
```

When configuring the analytics environment for Hortonworks, you would run the script from the **hdp2** directory. When complete, the script creates a **TKGrid_REP** child directory:

```
hdp2/TKGrid_REP
```

Configure for Access to a Data Store with a SAS Embedded Process

To configure the High-Performance Analytics environment for a remote data store, follow these steps:

1. Make sure that you have reviewed all of the information contained in the section "Preparing Your Data Provider for a Parallel Connection with SAS" on page 71.

2. Make sure that you understand how the analytics environment configuration script works, as described in "How the Configuration Script Works" on page 82.
3. The software that is needed for the analytics environment is available from within the SAS Software Depot that was created by the site depot administrator: *depot-installation-location*`/standalone_installs/SAS_High-Performance_Node_Installation/2_9/Linux_for_x64`.
4. Copy the TKGrid_REP file that is appropriate for your operating system to the `/tmp` directory of the root node of the analytic cluster.
5. Log on to the machine that will serve as the root node of the cluster with a user account that has the necessary permissions.

   For more information, see "User Accounts for the SAS High-Performance Analytics Environment" on page 25.
6. Change directories to the desired installation location, such as `/opt`.
7. Run the shell script in this directory.

   The shell script creates the `TKGrid_REP` subdirectory and places all files under that directory.
8. Respond to the prompts from the configuration program:

Table 6.3 *Configuration Parameters for the TKGrid_REP Shell Script*

Parameter	Description
Do you want to configure remote access to Teradata? (yes/NO)	If you are using a Teradata Managed Cabinet for your data provider, specify `yes` and press Enter. Otherwise, specify `no` and press Enter.
Do you want to use Teradata client installed in /opt/teradata/client/13.10 ? (YES/no)	If you have installed the Teradata client in the default path, then specify `yes` and press Enter. Otherwise, specify `no` and press Enter.
Enter path of Teradata client install. i.e.: /opt/teradata/client/13.10	If you specified `no` in the previous step, specify the path where the Teradata client was installed and press Enter. (This path was recorded earlier in Table 5.7 on page 74.)
Do you want to configure remote access to Greenplum? (yes/NO)	If you are using a Greenplum Data Computing Appliance for your data provider, specify `yes` and press Enter. Otherwise, specify `no` and press Enter.
Do you want to use Greenplum client installed in /usr/local/greenplum-db ? (YES/no)	If you have installed the Greenplum client in the default path, then specify `yes` and press Enter. Otherwise, specify `no` and press Enter.
Enter path of Greenplum client install. i.e.: /usr/local/greenplum-db	If you specified `no` in the previous step, specify the path where the Greenplum client was installed and press Enter. (This path was recorded earlier in Table 5.3 on page 73.)
Do you want to configure remote access to Hadoop? (yes/NO)	If you are using a Hadoop machine cluster for your data provider, specify `yes` and press Enter. Otherwise, specify `no` and press Enter.

Parameter	Description
Do you want to use the JRE installed in /opt/java/jre1.7.0_07 ?	If you want to use the JRE at the path that the install program lists, then press Enter. Otherwise, specify **no** and press Enter.
Enter path of the JRE i.e.: /opt/java/jre1.7.0_07	If you chose **no** in the previous step, specify the path where the JRE is installed and press Enter. (This path was recorded earlier in Table 5.2 on page 72.)
Enter path of the directory containing the Hadoop and client jars.	Specify the path where the Cloudera Hadoop JAR files required by SAS reside and press Enter. (This path was recorded earlier in Table 5.1 on page 72.)
Do you want to configure remote access to Oracle? (yes/NO)	If you are using an ORACLE Exadata appliance for your data provider, specify **yes** and press Enter. Otherwise, specify **no** and press Enter.
Enter path of Oracle client libraries. i.e.:/usr/local/ora11gr2/product/11.2.0/client_1/lib	Enter the path where the Oracle client libraries reside and press Enter. (This path was recorded earlier in Table 5.5 on page 73.)
Enter path of TNS_ADMIN, or just enter if not needed.	Enter the value of the Oracle TNS_ADMIN environment variable and press Enter. (This value was recorded earlier in Table 5.6 on page 74.)
Do you want to configure remote access to SAP HANA? (yes/NO)	If you are using a HANA cluster for your data provider, specify **yes** and press Enter. Otherwise, specify **no** and press Enter.
Enter path of HANA client install. i.e.: /usr/local/lib/hdbclient	Enter the path where the HANA client libraries reside and press Enter. (This path was recorded earlier in Table 5.4 on page 73.)
Shared install or replicate to each node? (Y=SHARED/ n=replicated)	If you are installing to a local drive on each node, then select **no** and press Enter to indicate that this is a replicated installation. If you are installing to a drive that is shared across all the nodes (for example, NFS), then specify **yes** and press Enter.
Enter path to TKGrid install	Specify the absolute path to where the SAS High-Performance Analytics environment is installed and press Enter. This should be the directory in which the analytics environment install program was run with **TKGrid** appended to it (for example, **/opt/TKGrid**). For more information, see Step 6 on page 79.
Enter additional paths to include in LD_LIBRARY_PATH, separated by colons (:)	If you have any external library paths that you want to be accessible to the SAS High-Performance Analytics environment, specify the paths here and press Enter. Separate paths with a colon (:). If you have no paths to specify, press Enter.

9. If you selected a replicated installation at the first prompt, you are now prompted to choose the technique for distributing the contents to the appliance nodes:

```
The install can now copy this directory to all the machines
listed in 'pathname' using scp, skipping the first entry. Perform copy?  (YES/no)
```

Press Enter if you want the installation program to perform the replication. Enter **no** if you are distributing the contents of the installation directory by some other technique.

10. You have finished deploying the analytics environment for a remote data source. If you have not done so already, install the appropriate SAS Embedded Process on the remote data appliance or machine cluster for your respective data provider.

    For more information, see *SAS In-Database Products: Administrator's Guide*, available at **http://support.sas.com/documentation/cdl/en/indbag/67365/PDF/default/indbag.pdf**.

Validating the Analytics Environment Deployment

Overview of Validating

You have at least two methods to validate your SAS High-Performance Analytics environment deployment:

- "Use simsh to Validate" on page 85.
- "Use MPI to Validate" on page 85.

Use simsh to Validate

To validate your SAS High-Performance Analytics environment deployment by issuing a **simsh** command, follow these steps:

1. Log on to the machine where SAS High-Performance Computing Management Console is installed.
2. Enter the following command:

   /HPA-environment-installation-directory/bin/simsh hostname

   This command invokes the **hostname** command on each machine in the cluster. The host name for each machine is printed to the screen.

   You should see a list of known hosts similar to the following:

```
myblade006.example.com: myblade006.example.com
myblade007.example.com: myblade007.example.com
myblade004.example.com: myblade004.example.com
myblade005.example.com: myblade005.example.com
```

3. Proceed to "Configuring Your Data Provider" on page 65.

Use MPI to Validate

To validate your SAS High-Performance Analytics environment deployment by issuing a Message Passing Interface (MPI) command, follow these steps:

1. Log on to the root node using the SAS High-Performance Analytics environment installation account.
2. Enter the following command:

   ```
   /HPA-environment-installation-directory/TKGrid/mpich2-install/bin/mpirun
   -f /etc/gridhosts hostname
   ```

   You should see a list of known hosts similar to the following:

```
myblade006.example.com
myblade007.example.com
myblade004.example.com
myblade005.example.com
```

3. Proceed to "Configuring Your Data Provider" on page 65.

Resource Management for the Analytics Environment

You can set limits on any TKGrid process running across the SAS High-Performance Analytics environment with a resource settings file supplied by SAS. Located in **/opt/TKGrid/**, resource.settings is in the format of a shell script. When the analytics environment starts, the environment variables contained in the file are set and last for the duration of the run.

Initially, all of the settings in resource.settings are commented. Uncomment the variables and add values that make sense for your site. For more information, see Appendix 5, "Using CGroups and Memory Limits," in *SAS LASR Analytic Server: Reference Guide* .

When you are finished editing, copy resource.settings to every machine in the analytics environment:

/opt/TKGrid/bin/simcp /opt/TKGrid/resource.settings /opt/TKGrid

If YARN is used on the cluster, then you can configure the analytic environment to participate in the resource accounting that YARN performs. For more information, see Appendix 5, "Managing Resources," in *SAS LASR Analytic Server: Reference Guide*.

resource.settings consists of the following:

```
# if [ "$USER" = "lasradm" ]; then
# Custom settings for any process running under the lasradm account.
#   export TKMPI_ULIMIT="-v 50000000"
#   export TKMPI_MEMSIZE=50000
#   export TKMPI_CGROUP="cgexec -g cpu:75"
# fi

# if [ "$TKMPI_APPNAME" = "lasr" ]; then
# Custom settings for a lasr process running under any account.
#   export TKMPI_ULIMIT="-v 50000000"
#   export TKMPI_MEMSIZE=50000
#   export TKMPI_CGROUP="cgexec -g cpu:75"

# Allow other users to read server and tables, but not add or term.
```

```
#   export TKMPI_UMASK=0033

# Allow no access by other users to lasr server.
#   export TKMPI_UMASK=0077

# To exclude from YARN resource manager.
#   unset TKMPI_RESOURCEMANAGER

# Use default nice for LASR
# unset TKMPI_NICE
# fi
# if [ "$TKMPI_APPNAME" = "tklogis" ]; then
# Custom settings for a tklogis process running under any account.
#   export TKMPI_ULIMIT="-v 25000000"
#   export TKMPI_MEMSIZE=25000
#   export TKMPI_CGROUP="cgexec -g cpu:25"
#   export TKMPI_MAXRUNTIME=7200
# fi
```

Appendix 1

Installing SAS Embedded Process for Hadoop

In-Database Deployment Package for Hadoop

Prerequisites

The following prerequisites are required before you install and configure the in-database deployment package for Hadoop:

- SAS Foundation and the SAS/ACCESS Interface to Hadoop are installed.
- You have working knowledge of the Hadoop vendor distribution that you are using (for example, Cloudera or Hortonworks).
- You have root or sudo access. Your user has Write permission to the root of HDFS.
- You know the location of the MapReduce home.
- You know the host name of the NameNode.
- You understand and can verify your Hadoop user authentication.
- You understand and can verify your security setup.

If you are using Kerberos, you need the ability to get a Kerberos ticket.

- You have permission to restart the Hadoop MapReduce service.
- In order to avoid SSH key mismatches during installation, add the following two options to the SSH **config** file, under the user's home .ssh folder. An example of a home .ssh folder is **/root/.ssh/**. *nodes* is a list of nodes separated by a space.

```
host nodes
   StrictHostKeyChecking no
   UserKnownHostsFile /dev/null
```

For more details about the SSH **config** file, see the SSH documentation.

- All machines in the cluster are set up to communicate with passwordless SSH. Verify that the nodes can access the node that you chose to be the master node by using SSH.

SSH keys can be generated with the following example.

```
[root@raincloud1 .ssh]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
09:f3:d7:15:57:8a:dd:9c:df:e5:e8:1d:e7:ab:67:86 root@raincloud1

add id_rsa.pub public key from each node to the master node authorized
   key file under /root/.ssh/authorized_keys
```

Overview of the In-Database Deployment Package for Hadoop

This section describes how to install and configure the in-database deployment package for Hadoop (SAS Embedded Process).

The in-database deployment package for Hadoop must be installed and configured before you can perform the following tasks:

- Read and write data to HDFS in parallel for SAS High-Performance Analytics.

  Note: For deployments that use SAS High-Performance Deployment of Hadoop for the co-located data provider, and access SASHDAT tables exclusively, SAS/ACCESS and SAS Embedded Process are not needed.

The in-database deployment package for Hadoop includes the SAS Embedded Process and two SAS Hadoop MapReduce JAR files. The SAS Embedded Process is a SAS server process that runs within Hadoop to read and write data. The SAS Embedded Process contains macros, run-time libraries, and other software that is installed on your Hadoop system.

The SAS Embedded Process must be installed on all nodes capable of executing either MapReduce 1 or MapReduce 2 and YARN tasks. The SAS Hadoop MapReduce JAR files must be installed on all nodes of a Hadoop cluster.

Hadoop Installation and Configuration

Hadoop Installation and Configuration Steps

Before you begin the Hadoop installation and configuration, please review "Prerequisites" on page 89.

1. Move the SAS Embedded Process and SAS Hadoop MapReduce JAR file install scripts to the Hadoop master node.

   CAUTION:
   Create a new directory that is not part of an existing directory structure, such as /sasep. This path will be created on each node in the Hadoop cluster during the SAS Embedded Process installation. Do not use existing system directories such as **`/opt`** or **`/usr`**. This new directory becomes the SAS Embedded Process home and is referred to as *SASEPHome* throughout this chapter.

   For more information, see "Moving the SAS Embedded Process and SAS Hadoop MapReduce JAR File Install Scripts" on page 91.

   Note: Both the SAS Embedded Process install script and the SAS Hadoop MapReduce JAR file install script must be transferred to the *SASEPHome* directory.

2. Install the SAS Embedded Process and the SAS Hadoop MapReduce JAR files.

   For more information, see "Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files" on page 92.

3. Move the Hadoop core and common Hadoop JAR files to the client machine.

   For more information, see "Moving Hadoop JAR Files to the Client Machine" on page 96.

Moving the SAS Embedded Process and SAS Hadoop MapReduce JAR File Install Scripts

Creating the SAS Embedded Process Directory

Before you can install the SAS Embedded Process and the SAS Hadoop MapReduce JAR files, you must move the SAS Embedded Process and SAS Hadoop MapReduce JAR file install scripts to the Hadoop master node.

CAUTION:
Create a new directory that is not part of an existing directory structure, such as /sasep. This path will be created on each node in the Hadoop cluster during the SAS Embedded Process installation. Do not use existing system directories such as **`/opt`** or **`/usr`**. This new directory becomes the SAS Embedded Process home and is referred to as *SASEPHome* throughout this chapter.

Moving the SAS Embedded Process Install Script

The SAS Embedded Process install script is contained in a self-extracting archive file named tkindbsrv-9.41_M2-*n*_lax.sh. *n* is a number that indicates the latest version of the file. If this is the initial installation, *n* has a value of 1. Each time you reinstall or upgrade, *n* is incremented by 1. The self-extracting archive file is located in the ***SAS-installation-directory*/SASTKInDatabaseServer/9.4/HadooponLinuxx64/** directory.

Using a method of your choice, transfer the SAS Embedded Process install script to your Hadoop master node.

This example uses secure copy, and *SASEPHome* is the location where you want to install the SAS Embedded Process.

```
scp tkindbsrv-9.41_M2-n_lax.sh username@hadoop:/SASEPHome
```

Note: Both the SAS Embedded Process install script and the SAS Hadoop MapReduce JAR file install script must be transferred to the *SASEPHome* directory.

Moving the SAS Hadoop MapReduce JAR File Install Script

The SAS Hadoop MapReduce JAR file install script is contained in a self-extracting archive file named hadoopmrjars-9.41_M2-*n*_lax.sh. *n* is a number that indicates the latest version of the file. If this is the initial installation, *n* has a value of 1. Each time you reinstall or upgrade, *n* is incremented by 1. The self-extracting archive file is located in the ***SAS-installation-directory*/SASACCESStoHadoopMapReduceJARFiles/9.41** directory.

Using a method of your choice, transfer the SAS Hadoop MapReduce JAR file install script to your Hadoop master node.

This example uses Secure Copy, and *SASEPHome* is the location where you want to install the SAS Hadoop MapReduce JAR files.

```
scp hadoopmrjars-9.41_M2-n_lax.sh username@hadoop:/SASEPHome
```

Note: Both the SAS Embedded Process install script and the SAS Hadoop MapReduce JAR file install script must be transferred to the *SASEPHome* directory.

Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files

To install the SAS Embedded Process, follow these steps.

Note: Permissions are needed to install the SAS Embedded Process and SAS Hadoop MapReduce JAR files. For more information, see "Hadoop Permissions" on page 102.

1. Log on to the server using SSH as root with sudo access.

   ```
   ssh username@serverhostname
   sudo su - root
   ```

2. Move to your Hadoop master node where you want the SAS Embedded Process installed.

   ```
   cd /SASEPHome
   ```

   SASEPHome is the same location to which you copied the self-extracting archive file. For more information, see "Moving the SAS Embedded Process Install Script" on page 92.

Note: Before continuing with the next step, ensure that each self-extracting archive file has Execute permission.

3. Use the following script to unpack the tkindbsrv-9.41_M2-*n*_lax.sh file.

```
./tkindbsrv-9.41_M2-n_lax.sh
```

n is a number that indicates the latest version of the file. If this is the initial installation, *n* has a value of 1. Each time you reinstall or upgrade, *n* is incremented by 1.

Note: If you unpack in the wrong directory, you can move it after the unpack.

After this script is run and the files are unpacked, the script creates the following directory structure, where *SASEPHome* is the master node from Step 1.

```
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/misc
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/sasexe
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/utilities
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/build
```

The content of the ***SASEPHome*/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin** directory should look similar to this.

```
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sas.ep4hadoop.template
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sasep-servers.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sasep-common.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sasep-server-start.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sasep-server-status.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sasep-server-stop.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/InstallTKIndbsrv.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/MANIFEST.MF
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/dqasetup.sh
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/S00qkb
SASEPHome/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/sas.tools.qkb.hadoop.jar
```

4. Use this command to unpack the SAS Hadoop MapReduce JAR files.

```
./hadoopmrjars-9.41_M2-1_lax.sh
```

After the script is run, the script creates the following directory and unpacks these files to that directory.

```
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/ep-config.xml
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/
   sas.hadoop.ep.apache023.jar
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/
    sas.hadoop.ep.apache023.nls.jar
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/
   sas.hadoop.ep.apache121.jar
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/
   sas.hadoop.ep.apache121.nls.jar
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/
   sas.hadoop.ep.apache205.jar
SASEPHome/SAS/SASACCESStoHadoopMapReduceJARFiles/9.41_M2/lib/
   sas.hadoop.ep.apache205.nls.jar
```

5. Use the **sasep-servers.sh -add** script to deploy the SAS Embedded Process installation across all nodes. The SAS Embedded Process is installed as a Linux service.

Note: If you are running on a cluster with Kerberos, complete both steps a and b. If you are not running with Kerberos, only complete step b.

TIP There are many options available when installing the SAS Embedded Process. We recommend that you review the script syntax before running it. For more information, see "SASEP-SERVERS.SH Script" on page 96.

a. If you are running on a cluster with Kerberos, you must **kinit** the HDFS user.

```
sudo - root
su - hdfs | hdfs-userid
kinit -kt location of keytab file
   user for which you are requesting a ticket
exit
```

Here is an example:

```
sudo - root
su - hdfs
kinit -kt hdfs.keytab hdfs
exit
```

Note: The default HDFS user is **hdfs**. You can specify a different user ID with the -hdfsuser argument when you run the **sasep-servers.sh -add** script.

Note: If you are running on a cluster with Kerberos, a keytab is required for the -hdfsuser running the **sasep-servers.sh -add** script.

Note: You can run klist while you are running as the -hdfsuser user to check the status of your Kerberos ticket on the server. Here is an example:

```
klist
Ticket cache: FILE/tmp/krb5cc_493
Default principal: hdfs@HOST.COMPANY.COM

Valid starting    Expires           Service principal
06/20/14 09:51:26 06/27/14 09:51:26 krbtgt/HOST.COMPANY.COM@HOST.COMPANY.COM
     renew until 06/22/14 09:51:26
```

b. After reviewing the notes that follow, run the **sasep-servers.sh** script.

```
cd SASEPHOME/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin
./sasep-servers.sh -add
```

During the installation process, the script asks whether you want to start the SAS Embedded Process. If you choose **Y** or **y**, the SAS Embedded Process is started on all nodes after the installation is complete. If you choose **N** or **n**, you can start the SAS Embedded Process later by running **./sasep-servers.sh -start**.

Note: When you run the **sasep-servers.sh -add** script, a user and group named **sasep** is created. You can specify a different user and group name with the -epuser and -epgroup arguments when you run the **sasep-servers.sh -add** script.

Note: The **sasep-servers.sh** script can be run from any location. You can also add its location to the PATH environment variable.

Note: Although you can install the SAS Embedded Process in multiple locations, the best practice is to install only one instance. Only one version of the SASEP JAR files will be installed in your ***/HadoopHOME*/lib** directory.

Note: The SAS Embedded Process runs on all the nodes that are capable of running a MapReduce task. In some instances, the node that you chose to be the master node can also serve as a MapReduce task node.

Note: If you install the SAS Embedded Process on a large cluster, the SSHD daemon might reach the maximum number of concurrent connections. The `ssh_exchange_identification: Connection closed by remote host` SSHD error might occur. Follow these steps to work around the problem:

1. Edit the /etc/ssh/sshd_config file and change the *MaxStartups* option to the number that accommodates your cluster.
2. Save the file and reload the SSHD daemon by running the /etc/init.d/sshd reload command.

6. Verify that the SAS Embedded Process is installed and running. Change directories and then run the **`sasep-servers.sh`** script with the **`-status`** option.

   ```
   cd SASEPHOME/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin
   ./sasep-servers.sh -status
   ```

   This command returns the status of the SAS Embedded Process running on each node of the Hadoop cluster. Verify that the SAS Embedded Process home directory is correct on all the nodes.

   Note: The **`sasep-servers.sh -status`** script will not run successfully if the SAS Embedded Process is not installed.

7. Verify that the sas.hadoop.ep.apache*.jar files are now in place.

   Note: You can find the location of *HadoopHome* by using the following command:

   ```
   hadoop version
   ```

   The JAR files are located at ***`/HadoopHome`*`/lib`**.

8. If this is the first installation of the SAS Embedded Process, a restart of the Hadoop YARN or MapReduce service is required.

   This enables the cluster to reload the SAS Hadoop JAR files (sas.hadoop.ep.*.jar).

   Note: It is preferable to restart the service by using Cloudera Manager or Hortonworks Ambari.

9. Verify that an init.d service with a sas.ep4hadoop file was created in the following directory.

   ```
   /etc/init.d/
   ```

   View the sas.ep4hadoop file and verify that the SAS Embedded Process home directory is correct.

   The init.d service is configured to start at level 3 and level 5.

   Note: The SAS Embedded Process needs to run on all nodes in the Hadoop cluster.

10. Verify that the configuration file, ep-config.xml, was written to the HDFS file system.

    ```
    hadoop fs -ls /sas/ep/config
    ```

    Note: If you are running on a cluster with Kerberos, you need a Kerberos ticket. If not, you can use the WebHDFS browser.

    Note: The **`/sas/ep/config`** directory is created automatically when you run the install script.

Moving Hadoop JAR Files to the Client Machine

For SAS components that interface with Hadoop, a specific set of common and core Hadoop JAR files must be in one location on the client machine. Examples of those components are the SAS Scoring Accelerator and SAS High-Performance Analytics.

When you run the **sasep-servers.sh -add** script to install the SAS Embedded Process, the script detects the Hadoop distribution and creates a HADOOP_JARS.zip file in the ***SASEPHome*/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/** directory. This file contains the common and core Hadoop JAR files that are required for the SAS Embedded Process. For more information, see "Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files" on page 92.

To get the Hadoop JAR files on your client machine, follow these steps:

1. Copy the HADOOP_JARS.zip file to a directory on your client machine and unzip the file.

   Note: You can use this command to list the JAR files in the HADOOP_JARS.zip file.

   ```
   unzip -l HADOOP_JARS.zip
   ```

2. Set the SAS_HADOOP_JAR_PATH environment variable to point to the directory that contains the core and common Hadoop JAR files.

Note: You can run the **sasep-servers.sh -getjars** script at any time to create a new ZIP file and refresh the JAR file list.

Note: The MapReduce 1 and MapReduce 2 JAR files cannot be on the same Java classpath.

Note: The JAR files in the SAS_HADOOP_JAR_PATH directory must match the Hadoop server to which SAS is connected. If multiple Hadoop servers are running different Hadoop versions, then create and populate separate directories with version-specific Hadoop JAR files for each Hadoop version. Then dynamically set SAS_HADOOP_JAR_PATH, based on the target Hadoop server to which each SAS job or SAS session is connected. One way to dynamically set SAS_HADOOP_JAR_PATH is to create a wrapper script associated with each Hadoop version. Then invoke SAS via a wrapper script that sets SAS_HADOOP_JAR_PATH appropriately to pick up the JAR files that match the target Hadoop server. Upgrading your Hadoop server version might involve multiple active Hadoop versions. The same multi-version instructions apply.

SASEP-SERVERS.SH Script

Overview of the SASEP-SERVERS.SH Script

The **sasep-servers.sh** script enables you to perform the following actions:

- install or uninstall the SAS Embedded Process and SAS Hadoop MapReduce JAR files on a single node or a group of nodes.
- start or stop the SAS Embedded Process on a single node or on a group of nodes.

- determine the status of the SAS Embedded Process on a single node or on a group of nodes.
- write the installation output to a log file.
- pass options to the SAS Embedded Process.
- create a HADOOP_JARZIP file in the local folder. This ZIP file contains all required client JAR files.

Note: The **`sasep-servers.sh`** script can be run from any folder on any node in the cluster. You can also add its location to the PATH environment variable.

Note: You must have sudo access to run the **`sasep-servers.sh`** script.

SASEP-SERVERS.SH Syntax

sasep-servers.sh

-add | -remove | -start | -stop | -status | -restart
<-mrhome *path-to-mr-home*>
<-hdfsuser *user-id*>
<-epuser>*epuser-id*
<-epgroup>*epgroup-id*
<-hostfile *host-list-filename* | -host <">*host-list*<">>
<-epscript *path-to-ep-install-script*>
<-mrscript *path-to-mr-jar-file-script*>
<-options "*option-list*">
<-log *filename*>
<-version *apache-version-number*>
<-getjars>

Arguments

-add
installs the SAS Embedded Process.

Note The -add argument also starts the SAS Embedded Process (same function as the -start argument). You are prompted and can choose whether to start the SAS Embedded Process.

Tip You can specify the hosts on which you want to install the SAS Embedded Process by using the -hostfile or -host option. The -hostfile or -host options are mutually exclusive.

See -hostfile and -host option on page 99

-remove
removes the SAS Embedded Process.

Tip You can specify the hosts for which you want to remove the SAS Embedded Process by using the -hostfile or -host option. The -hostfile or -host options are mutually exclusive.

See -hostfile and -host option on page 99

-start
starts the SAS Embedded Process.

Tip You can specify the hosts on which you want to start the SAS Embedded Process by using the -hostfile or -host option. The -hostfile or -host options are mutually exclusive.

See -hostfile and -host option on page 99

-stop
stops the SAS Embedded Process.

Tip You can specify the hosts on which you want to stop the SAS Embedded Process by using the -hostfile or -host option. The -hostfile or -host options are mutually exclusive.

See -hostfile and -host option on page 99

-status
provides the status of the SAS Embedded Process on all hosts or the hosts that you specify with either the -hostfile or -host option.

Tips The status also shows the version and path information for the SAS Embedded Process.

You can specify the hosts for which you want the status of the SAS Embedded Process by using the -hostfile or -host option. The -hostfile or -host options are mutually exclusive.

See -hostfile and -host option on page 99

-restart
restarts the SAS Embedded Process.

Tip You can specify the hosts on which you want to restart the SAS Embedded Process by using the -hostfile or -host option. The -hostfile or -host options are mutually exclusive.

See -hostfile and -host option on page 99

-mrhome *path-to-mr-home*
specifies the path to the MapReduce home.

-hdfsuser *user-id*
specifies the user ID that has Write access to HDFS root directory.

Default hdfs

Note The user ID is used to copy the SAS Embedded Process configuration files to HDFS.

-epuser *epuser-name*
specifies the name for the SAS Embedded Process user.

Default sasep

-epgroup *epgroup-name*
specifies the name for the SAS Embedded Process group.

Default sasep

-hostfile ***host-list-filename***
specifies the full path of a file that contains the list of hosts where the SAS Embedded Process is installed, removed, started, stopped, or status is provided.

Default If you do not specify -hostfile, the sasep-servers.sh script will discover the cluster topology and uses the retrieved list of data nodes.

Tip You can also assign a host list filename to a UNIX variable, **sas_ephosts_file**.

```
export sasep_hosts=/etc/hadoop/conf/slaves
```

See "-hdfsuser *user-id*" on page 98

Example

```
-hostfile /etc/hadoop/conf/slaves
```

-host <">*host-list*<">
specifies the target host or host list where the SAS Embedded Process is installed, removed, started, stopped, or checked for status.

Default If you do not specify -host, the sasep-servers.sh script will discover the cluster topology and uses the retrieved list of data nodes.

Requirement If you specify more than one host, the hosts must be enclosed in double quotation marks and separated by spaces.

Tip You can also assign a list of hosts to a UNIX variable, **sas_ephosts**.

```
export sasep_hosts="server1 server2 server3"
```

See "-hdfsuser *user-id*" on page 98

Example

```
-host "server1 server2 server3"
-host bluesvr
```

-epscript ***path-to-ep-install-script***
copies and unpacks the SAS Embedded Process install script file to the host.

Restriction Use this option only with the -add option.

Requirement You must specify either the full or relative path of the SAS Embedded Process install script, tkindbsrv-9.41_M2-*n*_lax.sh file.

Example

```
-epscript /home/hadoop/image/current/tkindbsrv-9.41_M2-2_lax.sh
```

-mrscript ***path-to-mr-jar-file-script***
copies and unpacks the SAS Hadoop MapReduce JAR files install script on the hosts.

Restriction Use this option only with the -add option.

Requirement You must specify either the full or relative path of the SAS Hadoop MapReduce JAR file install script, hadoopmrjars-9.42.*n*_lax.sh file.

Example

```
-mrscript /home/hadoop/image/current/tkindbsrv-9.41_M2-2_lax.sh
```

-options "*option-list*"
specifies options that are passed directly to the SAS Embedded Process. The following options can be used.

-trace *trace-level*
specifies what type of trace information is created.

0	no trace log
1	fatal error
2	error with information or data value
3	warning
4	note
5	information as an SQL statement
6	critical and command trace
7	detail trace, lock
8	enter and exit of procedures
9	tedious trace for data types and values
10	trace all information

Default 02

Note The trace log messages are stored in the MapReduce job log.

-port *port-number*
specifies the TCP port number where the SAS Embedded Process accepts connections.

Default 9261

Requirement The options in the list must be separated by spaces, and the list must be enclosed in double quotation marks.

-log *filename*
writes the installation output to the specified filename.

-version *apache-version-number*
specifies the Hadoop version of the JAR file that you want to install on the cluster. The *apache-version-number* can be one of the following values.

0.23
installs the SAS Hadoop MapReduce JAR files that are built from Apache Hadoop 0.23 (sas.hadoop.ep.apache023.jar and sas.hadoop.ep.apache023.nls.jar).

1.2
installs the SAS Hadoop MapReduce JAR files that are built from Apache Hadoop 1.2.1 (sas.hadoop.ep.apache121.jar and sas.hadoop.ep.apache121.nls.jar).

2.0
installs the SAS Hadoop MapReduce JAR files that are built from Apache Hadoop 0.2.3 (sas.hadoop.ep.apache023.jar and sas.hadoop.ep.apache023.nls.jar).

2.1
installs the SAS Hadoop MapReduce JAR files that are built from Apache Hadoop 2.0.5 (sas.hadoop.ep.apache205.jar and sas.hadoop.ep.apache205.nls.jar).

Default If you do not specify the -version option, the sasep.servers.sh script will detect the version of Hadoop that is in use and install the JAR files associated with that version. For more information, see "Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files" on page 92.

Interaction The -version option overrides the version that is automatically detected by the sasep.servers.sh script.

-getjars
creates a HADOOP_JARZIP file in the local folder. This ZIP file contains all required client JAR files.

You need to move this ZIP file to your client machine and unpack it. If you want to replace the existing JAR files, move it to the same directory where you previously unpacked the existing JAR files.

See For more information, see "Moving Hadoop JAR Files to the Client Machine" on page 96.

Starting the SAS Embedded Process

There are three ways to manually start the SAS Embedded Process.

Note: Root authority is required to run the **`sasep-servers.sh`** script.

- Run the **`sasep-servers.sh`** script with the **`-start`** option on the master node.

  This starts the SAS Embedded Process on all nodes. For more information about running the **`sasep-servers.sh`** script, see "SASEP-SERVERS.SH Syntax" on page 97.

- Run **`sasep-server-start.sh`** on a node.

  This starts the SAS Embedded Process on the local node only. The **`sasep-server-start.sh`** script is located in the ***`SASEPHome`*`/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/`** directory. For more information, see "Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files" on page 92.

- Run the UNIX **`service`** command on a node.

  This starts the SAS Embedded Process on the local node only. The **`service`** command calls the init script that is located in the **`/etc/init.d`** directory. A symbolic link to the init script is created in the **`/etc/rc3.d`** and **`/etc/rc5.d`** directories, where **`3`** and **`5`** are the run level at which you want the script to be executed.

  Because the SAS Embedded Process init script is registered as a service, the SAS Embedded Process is started automatically when the node is rebooted.

Stopping the SAS Embedded Process

The SAS Embedded Process continues to run until it is manually stopped. The ability to control the SAS Embedded Process on individual nodes could be useful when performing maintenance on an individual node.

There are three ways to stop the SAS Embedded Process.

Note: Root authority is required to run the **sasep-servers.sh** script.

- Run the **sasep-servers.sh** script with the **-stop** option from the master node.

  This stops the SAS Embedded Process on all nodes. For more information about running the sasep-servers.sh script, see "SASEP-SERVERS.SH Syntax" on page 97.

- Run **sasep-server-stop.sh** on a node.

  This stops the SAS Embedded Process on the local node only. The **sasep-server-stop.sh** script is located in the ***SASEPHome*/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/** directory. For more information, see "Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files" on page 92.

- Run the UNIX **service** command on a node.

  This stops the SAS Embedded Process on the local node only.

Determining the Status of the SAS Embedded Process

You can display the status of the SAS Embedded Process on one node or all nodes. There are three ways to display the status of the SAS Embedded Process.

Note: Root authority is required to run the **sasep-servers.sh** script.

- Run the **sasep-servers.sh** script with the **-status** option from the master node.

  This displays the status of the SAS Embedded Process on all nodes. For more information about running the **sasep-servers.sh** script, see "SASEP-SERVERS.SH Syntax" on page 97.

- Run **sasep-server-status.sh** from a node.

  This displays the status of the SAS Embedded Process on the local node only. The **sasep-server-status.sh** script is located in the ***SASEPHome*/SAS/SASTKInDatabaseServerForHadoop/9.41_M2/bin/** directory. For more information, see "Installing the SAS Embedded Process and SAS Hadoop MapReduce JAR Files" on page 92.

- Run the UNIX **service** command on a node.

  This displays the status of the SAS Embedded Process on the local node only.

Hadoop Permissions

The person who installs the SAS Embedded Process must have sudo access.

Documentation for Using In-Database Processing in Hadoop

For information about using in-database processing in Hadoop, see the following publications:

- *SAS In-Database Products: User's Guide*
- High-performance procedures in various SAS publications
- *SAS Data Integration Studio: User's Guide*
- SAS/ACCESS Interface to Hadoop and PROC HDMD in *SAS/ACCESS for Relational Databases: Reference*
- *SAS Intelligence Platform: Data Administration Guide*
- PROC HADOOP in *Base SAS Procedures Guide*
- FILENAME Statement, Hadoop Access Method in *SAS Statements: Reference*
- *SAS Data Quality Accelerator 2.5 for Hadoop: User's Guide*

Appendix 2

Updating the SAS High-Performance Analytics Infrastructure

Overview of Updating the Analytics Infrastructure

Here are some considerations for updating the SAS High-Performance Analytics infrastructure:

- Because of dependencies, if you update the analytics environment, you must also update SAS High-Performance Deployment of Hadoop.
- Update Hadoop first, followed by the analytics environment.

Updating the SAS High-Performance Computing Management Console

Overview of Updating the Management Console

Starting in version 2.6 of SAS High-Performance Computing Management Console, there is no longer support for memory management through CGroups.

Before upgrading the management console to version 2.6, make sure that you manually record any memory settings and then clear them on the **CGroup Resource Management** page. You can manually transfer these memory settings to the SAS High-Performance Analytics environment resource settings file. Or, if you are are

implementing YARN, transfer these settings to YARN. For more information, see *SAS LASR Analytic Server: Reference Guide*, available at **http://support.sas.com/documentation/solutions/va/index.html**.

Update the Management Console Using RPM

To update your deployment of SAS High-Performance Computing Management Console, follow these steps:

1. Make sure that you have manually recorded and then cleared any memory settings in the management console. For more information, see "Overview of Updating the Management Console" on page 105.
2. Stop the server by entering the following command as the **root** user:

   ```
   service sashpcmc stop
   ```

3. Update the management console using the following RPM command:

   ```
   rpm -U /SAS-Software-Depot-Root-Directory/standalone_installs/
   SAS_High-Performance_Computing_Management_Console/2_6/Linux_for_x64/
   sashpcmc-2.6.x86_64.rpm
   ```

4. Log on to the console to validate your update.

Updating SAS High-Performance Deployment of Hadoop

Overview of Updating SAS High-Performance Deployment of Hadoop

The SAS High-Performance Deployment of Hadoop package consists of the following major components:

- Apache Hadoop
- LASR Analytic Server Hadoop adapter components (JAR files and shared libraries)

SAS gives you two options for updating SAS High-Performance Deployment of Hadoop:

- Update LASR Analytic Server Hadoop adapter components only:

  You update the LASR Analytic Server Hadoop adapter components (JAR files and shared libraries) only. Apache Hadoop and the HDFS file system are not modified.

  This approach is simpler than a full Hadoop upgrade, and has a lesser impact from a change management perspective.

  For more information, see "Update SAS High-Performance Deployment of Hadoop (SAS LASR Adapter Components Only)" on page 107.

- Update SAS High-Performance Deployment of Hadoop:

  You update the LASR Analytic Server Hadoop adapter components (JAR files and shared libraries), Apache Hadoop, and the HDFS file system. Your data that resides in your current version of Hadoop will be upgraded in place. The new version of Hadoop will access that data.

This approach is more complicated than updating the LASR Hadoop adapter components only, and has a greater impact from a change management perspective.

For more information, see "Update SAS High-Performance Deployment of Hadoop" on page 109.

Preparing to Update Hadoop

Prior to starting the SAS High-Performance Deployment of Hadoop update, perform the following steps:

Note: The following steps also apply when you are upgrading SAS LASR adapter components only.

1. If one does not already exist, create a SAS Software Depot that contains the installation software that you will use to update Hadoop.

   For more information, see "Creating a SAS Software Depot" in the *SAS Intelligence Platform: Installation and Configuration Guide*, available at **http://support.sas.com/documentation/cdl/en/biig/63852/HTML/default/p03intellplatform00installgd.htm**.

2. Log on to the Hadoop NameNode as the hdfs user.

3. Run the following command to make sure that the Hadoop file system is healthy: **hadoop fsck /**

   Correct any issues before proceeding.

4. Stop any other processes, such as YARN, running on the Hadoop cluster.

   Confirm that all processes have stopped across all the cluster machines. (You might have to become another user to have the necessarily privileges to stop all processes.)

5. As the Hadoop user, run the command **$HADOOP_HOME/bin/stop-dfs.sh** to stop HDFS daemons, and confirm that all processes have ceased on all the machines in the cluster.

   TIP Check that there are no Java processes owned by **hadoop** running on any machine: **ps -ef | grep hadoop**. If you find any Java processes owned by the **hadoop** user account, terminate them.You can issue a single **simsh** command to simultaneously check all the machines in the cluster: ***/HPA-environment-installation-directory*/bin/simsh ps -ef | grep hadoop**.

6. Back up the Hadoop name directory (**hadoop-name** by default).

   Perform a file system backup using tar (or whatever tool or process that your site uses for backups).

Update SAS High-Performance Deployment of Hadoop (SAS LASR Adapter Components Only)

The Hadoop install script gives you the option of upgrading of the LASR Analytic Server Hadoop adapter components (JAR files and shared libraries) only. When you choose this option, the install script does *not* modify Apache Hadoop and the HDFS file system.

To update LASR Analytic Server Hadoop adapter components (JAR files and shared libraries) only, follow these steps:

1. Make sure that you have performed the steps listed in the section "Preparing to Update Hadoop" on page 107.
2. Log on to the Hadoop NameNode as the user ID that owns your current Hadoop installation directories.
3. Copy the sashadoop.tar.gz file to a temporary location and extract it:

```
cp sashadoop.tar.gz /tmp
cd /tmp
tar xzf sashadoop.tar.gz
```

   A directory that is named **sashadoop** is created.

4. Change directory to the **sashadoop** directory and run the **hadoopInstall** command:

```
cd sashadoop
./hadoopInstall
```

5. Respond to the prompts from the configuration program:

Table A2.1 *SAS High-Performance Deployment of Hadoop Configuration Parameters*

Parameter	Description
Do you wish to use an existing Hadoop installation? (y/N)	Enter **y** and press Enter.
Enter path to existing Hadoop installation.	Specify the value of HADOOP_HOME (for example, **/opt/hadoop/hadoop-0.23.1**) and press Enter.
Supported version of Hadoop found at: '/opt/hadoop/hadoop-0.23.1' Updating Hadoop install at: '/opt/hadoop/hadoop-0.23.1' Stop Hadoop server at: '/opt/hadoop/hadoop-0.23.1', and Hit Return.	Be sure that the Hadoop server is stopped ($HADOOP_HOME/sbin/stop-dfs.sh) and press Enter.

The install script outputs messages similar to the following:

```
Verify that the following lines are in '/opt/hadoop/hadoop-0.23.1/etc/
hadoop/hdfs-site.xml'.

<property>
  <name>dfs.permissions.enabled</name>
  <value>true</value>
</property>
<property>
  <name>dfs.namenode.plugins</name>
  <value>com.sas.lasr.hadoop.NameNodeService</value>
</property>
<property>
  <name>dfs.datanode.plugins</name>
  <value>com.sas.lasr.hadoop.DataNodeService</value>
</property>
<property>
  <name>com.sas.lasr.hadoop.fileinfo</name>
  <value>ls -l {0}</value>
```

```
  <description>The command used to get the user, group, and permission
  information for a file.
  </description>
</property>
<property>
  <name>com.sas.lasr.service.allow.put</name>
  <value>true</value>
  <description>Flag indicating whether the PUT command is enabled when
  running as a service. The default is false.
  </description>
</property>


Installation complete. Please restart your Hadoop server.
```

6. Verify that each on node, the hdfs-site.xml file contains the earlier listed properties.
7. Restart hadoop by entering the following command:

   $HADOOP_HOME/sbin/start-dfs.sh

Update SAS High-Performance Deployment of Hadoop

Version 2.6 of SAS High-Performance Deployment of Hadoop represents a version upgrade of Apache Hadoop (version 0.23.1 to version 2.4). This newer version includes new features such as YARN. During an upgrade, the install script installs a new version of Hadoop. Your data that resides in your current version of Hadoop will be upgraded in place. The new version of Hadoop will access that data.

Before you update Hadoop, you must gather the following information listed in Table A2.2. You can find most of this information in your current Hadoop configuration file, **$HADOOP_HOME/etc/hadoop/hdfs-site.xml**:

Table A2.2 *Hadoop Installation Checklist*

Installation Prompt	Requirement / How to Locate	Actual Value
Hadoop install directory	One level above the current HADOOP_HOME value. For example: **/hadoop**	
Replication factor	Refer to hdfs-site.xml.	
Port for fs.defaultFS	Refer to core-site.xml.	
Port for mapred.job.tracker	Refer to mapred_site.xml	
Port for dfs.datanode.address	Refer to hdfs-site.xml.	
Port for dfs.namenode.backup.address	Refer to hdfs-site.xml.	
Port for dfs.namenode.https-address	Refer to hdfs-site.xml.	

Installation Prompt	Requirement / How to Locate	Actual Value
Port for dfs.datanode.https.address	Refer to hdfs-site.xml.	
Port for dfs.datanode.ipc.address	Refer to hdfs-site.xml.	
Port for dfs.namenode.http-address	Refer to hdfs-site.xml.	
Port for dfs.datanode.http.address	Refer to hdfs-site.xml.	
Port for dfs.secondary.http.address	Refer to hdfs-site.xml.	
Port for dfs.namenode.backup.addres	Refer to hdfs-site.xml.	
Port for dfs.namenode.backup.http-address	Refer to hdfs-site.xml.	
Port for com.sas.lasr.hadoop.service.namenode.port	Refer to hdfs-site.xml.	
Port for com.sas.lasr.hadoop.service.datanode.port	Refer to hdfs-site.xml.	
HDFS server process user	Must be the same user as current Hadoop user.	
Path for JAVA_HOME directory	Location of your JRE installation (default: `/usr/lib/jvm/jre`).	
Path for Hadoop data directory	Same as the current Hadoop data directory. Refer to hdfs-site.xml.	
Path for Hadoop name directory	Same as the current Hadoop name directory. Refer to hdfs-site.xml.	
Path to machine list	See "List the Machines in the Cluster or Appliance".	

To update SAS High-Performance Deployment of Hadoop, follow these steps:

1. Make sure that you have performed the steps listed in the section "Preparing to Update Hadoop" on page 107.
2. Log on to the Hadoop NameNode as the root user.
3. Copy the sashadoop.tar.gz file to a temporary location and extract it:

```
cp sashadoop.tar.gz /tmp
cd /tmp
tar xzf sashadoop.tar.gz
```

A directory that is named **sashadoop** is created.

4. Change directory to the **sashadoop** directory and run the **hadoopInstall** command:

```
cd sashadoop
./hadoopInstall
```

5. Using the information that you gathered earlier in Table A2.2, respond to the prompts from the configuration program:

Table A2.3 *SAS High-Performance Deployment of Hadoop Configuration Parameters*

Parameter	Description
Do you wish to use an existing Hadoop installation? (y/N)	Enter **n** and press Enter. *Note:* Enter **n** because you have decided to use a new version of Hadoop binaries. The utility installs a newer Apache Hadoop with updated SAS LASR adapter components. Subsequent steps will instruct you to upgrade your data in place.
Enter path to install Hadoop. The directory 'hadoop-2.4.0' will be created in the path specified.	Specify the directory one level above the current HADOOP_HOME and press Enter. Refer to Table A2.2.
Do you wish to use Yarn and MR Jobhistory Server? (y/N)	If you plan to use YARN and MapReduce, specify **y** and press Enter. If you are using YARN, be sure to review "Preparing for YARN (Experimental)" on page 23 before proceeding. Otherwise, specify **n** and press Enter.
Enter replication factor. Default 2	Specify the replication factor used for your current Hadoop deployment and press Enter. Refer to Table A2.2.

Parameter	Description
Enter port number for fs.defaultFS. Default 54310 Enter port number for dfs.namenode.https-address. Default 50470 Enter port number for dfs.datanode.https.address. Default 50475 Enter port number for dfs.datanode.address. Default 50010 Enter port number for dfs.datanode.ipc.address. Default 50020 Enter port number for dfs.namenode.http-address. Default 50070 Enter port number for dfs.datanode.http.address. Default 50075 Enter port number for dfs.secondary.http.address. Default 50090 Enter port number for dfs.namenode.backup.address. Default 50100 Enter port number for dfs.namenode.backup.http-address. Default 50105 Enter port number for com.sas.lasr.hadoop.service.namenode.port. Default 15452 Enter port number for com.sas.lasr.hadoop.service.datanode.port. Default 15453	Specify each port and press Enter. Refer to Table A2.2.
[The following port prompts are displayed when you choose to deploy YARN:] Enter port number for mapreduce.jobhistory.admin.address. Default 10033 Enter port number for mapreduce.jobhistory.webapp.address. Default 19888 Enter port number for mapreduce.jobhistory.address. Default 10021 Enter port number for yarn.resourcemanager.scheduler.address. Default 8030 Enter port number for yarn.resourcemanager.resource-tracker.address. Default 8031 Enter port number for yarn.resourcemanager.address. Default 8032 Enter port number for yarn.resourcemanager.admin.address. Default 8033 Enter port number for yarn.resourcemanager.webapp.address. Default 8088 Enter port number for yarn.nodemanager.localizer.address. Default 8040 Enter port number for yarn.nodemanager.webapp.address. Default 8042 Enter port number for yarn.web-proxy.address. Default 10022	Specify each port and press Enter. Refer to Table A2.2.

Parameter	Description
Enter maximum memory allocation per Yarn container. Default 5905	This is the maximum amount of memory (in MB) that YARN can allocate on a particular machine in the cluster. Press Enter to accept the default or specify a different value and press Enter.
Enter user that will be running the HDFS server process.	Specify the user name (for example, hdfs) and press Enter. Refer to Table A2.2.
Enter user that will be running Yarn services.	Specify the user name (for example, yarn) and press Enter. For more information, see "Preparing for YARN (Experimental)" on page 23.
Enter user that will be running the Map Reduce Job History Server.	Specify the user name (for example, mapred) and press Enter. For more information, see "Preparing for YARN (Experimental)" on page 23.
Enter common primary group for users running Hadoop services.	Apache recommends that the hdfs, mapred, and YARN users share the same primary Linux group. Enter a group name and press Enter. For more information, see "Preparing for YARN (Experimental)" on page 23.
Enter path for JAVA_HOME directory. (Default: /usr/lib/jvm/jre)	Specify the path to the JRE or JDK and press Enter. Refer to Table A2.2. *Note:* The configuration program does not verify that a JRE is installed at **`/usr/lib/jvm/jre`**, which is the default path for some Linux vendors.
Enter path for Hadoop data directory. This should be on a large drive. Default is '/hadoop/hadoop-data'. Enter path for Hadoop name directory. Default is '/hadoop/hadoop-name'.	Specify the paths to your current Hadoop data and name directories and press Enter. Refer to Table A2.2. *Note:* The data directory cannot be the root directory of a partition or mount. *Note:* If you have more than one data device, enter one of the data directories now, and refer to "(Optional) Deploy with Multiple Data Devices" on page 47 after the installation.
Enter full path to machine list. The NameNode '**`host`**' should be listed first.	Specify the path to your current machine list and press Enter. Refer to Table A2.2.

6. You will see failure to create directory errors for a directory other than the **`hadoop-2.4.0`** directory. These are normal, since the directories being created already exist. These errors occur on all nodes after you confirm that you want the installation program to copy the installation to all nodes.

   CAUTION:
   After the installation is complete, do *not* reformat the NameNode. Reformatting the Hadoop NameNode deletes your data in the HDFS cluster.

7. Log out as the root user. Log in as the hdfs user.
8. Run this command to define HADOOP_HOME in the Hadoop user's (hdfs) environment:

export HADOOP_HOME=/*installation-directory*/hadoop/hadoop-2.4.0

where *installation-directory* is the location where you installed Hadoop (for example, **/opt/hadoop/hadoop-2.4.0**).

9. Run the following command to start Hadoop: **$HADOOP_HOME/sbin/start-dfs.sh -upgrade**.
10. Run the following command: **$HADOOP_HOME/bin/hadoop fsck /**.

    You should see a healthy file system and the correct number of DataNodes.
11. The initial-sas-hdfs-setup.sh script makes modifications required for Hadoop, such as creating some new directories that support YARN and applying permissions that improve security. Review the **hdfs fs** commands that are listed in **$HADOOP_HOME/sbin/initial-sas-hdfs-setup.sh** and then run this script once.

    Alternatively, you can run individual commands from the script if you understand how the commands modify HDFS.
12. Confirm that Hadoop is running successfully by opening a browser to **http://*namenode*:50070/dfshealth.html**. Review the information in the cluster summary section of the page. Confirm that the number of live nodes equals the number of DataNodes and that the number of dead nodes is zero.
13. If you do *not* plan to update the SAS High-Performance Analytics environment, then you must manually update the analytics environment to reflect the new HADOOP_HOME value. Do this by editing **$GRIDINSTALLLOC/tkmpirsh.sh**. Then copy this file to the same location across all the machines in the cluster. For example:

    ```
    /opt/TKGrid/bin/simcp $GRIDINSTALLLOC/tkmpirsh.sh $GRIDINSTALLLOC/tkmpirsh.sh
    ```

Update the Analytics Environment

You have the following options for managing updates to the SAS High-Performance Analytics environment:

- Delete the SAS High-Performance Analytics environment and install the newer version.

  See the procedure later in this topic.
- Rename the root installation directory for the current SAS High-Performance Analytics environment, and install the newer version under the previous root installation directory.

  See “Install the Analytics Environment” on page 78.
- Do nothing to the current SAS High-Performance Analytics environment, and install the new version under a new installation directory.

  See “Install the Analytics Environment” on page 78.

  When you change the path of the SAS High-Performance Analytics environment, you have to also have to reconfigure the SAS LASR Analytic Server to point to the new path. See “Add a SAS LASR Analytic Server” in Chapter 5 of *SAS Visual*

Analytics: Administration Guide available at **http://support.sas.com/documentation/solutions/va/index.html**.

Updating your deployment of the SAS High-Performance Analytics environment consists of deleting the deployment and reinstalling the newer version. To update the SAS High-Performance Analytics environment, follow these steps:

1. Check that there are no analytics environment processes running on any machine:

   ```
   ps -ef | grep TKGrid
   ```

   If you find any TKGrid processes, terminate them.

   TIP You can issue a single **simsh** command to simultaneously check all the machines in the cluster: ***/HPA-environment-installation-directory*/bin/simsh ps -ef | grep TKGrid**.

2. Delete the analytics environment installation directory on every machine in the cluster:

   ```
   rm -r -f /HPA-environment-install-dir
   ```

   TIP You can issue a single **simsh** command to simultaneously remove the environment install directories on all the machines in the cluster: ***/HPA-environment-installation-directory*/bin/simsh rm -r -f /*HPA-environment-installation-directory***.

3. Re-install the analytics environment using the shell script as described in "Install the Analytics Environment" on page 78.

Appendix 3

SAS High-Performance Analytics Infrastructure Command Reference

The **simsh** and **simcp** commands are installed with SAS High-Performance Computing Management Console and the SAS High-Performance Analytics environment. The default path to the commands is ***/HPCMC-installation-directory*/webmin/utilbin** and ***/HPA-environment-installation-directory*/bin**, respectively. Any user account that can access the commands and has passwordless secure shell configured can use them.

TIP Add one of the earlier referenced installation paths to your system PATH variable to make invoking **simsh** and **simcp** easier.

The **simsh** command uses secure shell to invoke the specified command on every machine that is listed in the **/etc/gridhosts** file. The following command demonstrates invoking the **hostname** command on each machine in the cluster:

```
/HPCMC-install-dir/webmin/utilbin/simsh hostname
```

TIP You can use SAS High-Performance Computing Management Console to create and manage your grid hosts file. For more information, see *SAS High-Performance Computing Management Console: User's Guide*, available at **http://support.sas.com/documentation/onlinedoc/va/index.html**.

The **simcp** command is used to copy a file from one machine to the other machines in the cluster. Passwordless secure shell and an **/etc/gridhosts** file are required. The following command is an example of copying the **/etc/hosts** file to each machine in the cluster:

```
/HPA-environment-installation-directory/bin/simcp /etc/hosts /etc
```

Appendix 4

SAS High-Performance Analytics Environment Client-Side Environment Variables

The following environment variables can be used on the client side to control the connection to the SAS High-Performance Analytics environment. You can set these environment variables in the following ways:

- invoke them in your SAS program using **`options set=`**
- add them to your shell before running the SAS program
- add them to your sasenv_local configuration file, if you want them used in all SAS programs

GRIDHOST=
identifies the root node on the SAS High-Performance Analytics environment to which the client connects.

The values for GRIDHOST and GRIDINSTALLLOC can both be specified in the GRIDHOST variable, separated by a colon (similar to the format used by **`scp`**). For example:

```
GRIDHOST=my_machine_cluster_001:/opt/TKGrid
```

GRIDINSTALLLOC=
identifies the location on the machine cluster where the SAS High-Performance Analytics environment is installed. For example:

```
GRIDINSTALLLOC=/opt/TKGrid
```

GRIDMODE=SYM | ASYM
toggles the SAS High-Performance Analytics environment between symmetric (default) and asymmetric mode.

GRIDRSHCOMMAND= " " | " *ssh-path*"
(optional) specifies **`rsh`** or **`ssh`** used to launch the SAS High-Performance Analytics environment.

If unspecified or a null value is supplied, a SAS implementation of the SSH protocol is used.

ssh-path specifies the path to the SSH executable that you want to use. This can be useful in deployments where export controls restrict SAS from delivering software that uses cryptography. For example:

```
option set=GRIDRSHCOMMAND="/usr/bin/ssh";
```

GRIDPORTRANGE=
identifies the port range for the client to open. The root node connects back to the client using ports in the specified range. For example:

```
option set=GRIDPORTRANGE=7000-8000;
```

GRIDREPLYHOST=

specifies the name of the client machine to which the SAS High-Performance Analytics environment connects. **GRIDREPLYHOST** is used when the client has more than one network card or when you need to specify a full network name.

GRIDREPLYHOST can be useful when you need to specify a fully qualified domain name, when the client has more than one network interface card, or when you need to specify an IP address for a client with a dynamically assigned IP address that domain name resolution has not registered yet. For example:

```
GRIDREPLYHOST=myclient.example.com
```

Appendix 5

Deploying on SELinux and IPTables

Overview of Deploying on SELinux and IPTables

This document describes how to prepare Security Enhanced Linux (SELinux) and IPTables for a SAS High-Performance Analytics infrastructure deployment.

Security Enhanced Linux (SELinux) is a feature in some versions of Linux that provides a mechanism for supporting access control security policies. IPTables is a firewall—a combination of a packet-filtering framework and generic table structure for defining rulesets. SELinux and IPTables is available in most new distributions of Linux, both community-based and enterprise-ready. For sites that require added security, the use of SELinux and IPTables is an accepted approach for many IT departments.

Because of the limitless configuration possibilities, this document is based on the default configuration for SELinux and IPTables running on RedHat Enterprise Linux (RHEL) 6.3. You might need to adjust the directions accordingly, especially for complex SELinux and IPTables configurations.

Prepare the Management Console

SELinux Modifications for the Management Console

After generating and propagating root's SSH keys throughout the cluster or data appliance, you must run the following command on every machine or blade to restore the security context on the files in **/root/.ssh**:

```
restorecon -R -v /root/.ssh
```

IPTables Modifications for the Management Console

Add the following line to **/etc/sysconfig/iptables** to allow connections to the port on which the management console is listening (10020 by default). Open the port only on the machine on which the management console is running:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 10020 -j ACCEPT
```

Prepare Hadoop

SELinux Modifications for Hadoop

After generating and propagating root's SSH keys throughout the cluster or data appliance, you must run the following command on every machine or blade to restore the security context on the files in **/root/.ssh**:

```
restorecon -R -v /root/.ssh
```

IPTables Modifications for Hadoop

The SAS High-Performance Deployment of Hadoop has a number of ports on which it communicates. To open these ports, place the following lines in **/etc/sysconfig/iptables**:

Note: The following example uses default ports. Modify as necessary for your site.

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 54310 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 54311 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50470 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50475 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50010 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50020 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50070 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50075 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50090 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50100 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50105 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50030 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50060 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 15452 -j ACCEPT
```

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 15453 -j ACCEPT
```

Edit **/etc/sysconfig/iptables** and then copy this file across the machine cluster or data appliance. Lastly, restart the IPTables service.

Prepare the Analytics Environment

SELinux Modifications for the Analytics Environment

After generating and propagating root's SSH keys throughout the cluster or data appliance, you must run the following command on every machine or blade to restore the security context on the files in **/root/.ssh**:

```
restorecon -R -v /root/.ssh
```

IPTables Modifications for the Analytics Environment

If you are deploying the SAS LASR Analytic Server, then you must define one port per server in **/etc/sysconfig/iptables**. (The port number is defined in the SAS code that starts the SAS LASR Analytic server.)

If you have more than one server running simultaneously, you need all these ports defined in the form of a range.

The following is an example of an iptables entry for a single server (one port):

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 10010 -j ACCEPT
```

The following is an example of an iptables entry for five servers (port range):

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 10010:10014 -j ACCEPT
```

MPICH_PORT_RANGE must also be opened in IPTables by editing the **/etc/sysconfig/iptables** file and adding the port range.

The following is an example for five servers:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 10010:10029 -j ACCEPT
```

Edit **/etc/sysconfig/iptables** and then copy this file across the machine cluster or data appliance. Lastly, restart the IPTables service.

Analytics Environment Post-Installation Modifications

The SAS High-Performance Analytics environment uses Message Passing Interface (MPI) communications, which requires you to define one port range per active job across the machine cluster or data appliance.

(A port range consists of a minimum of four ports per active job. Every running monitoring server counts as a job on the cluster or appliance.)

For example, if you have five jobs running simultaneously across the machine cluster or data appliance, you need a minimum of 20 ports in the range.

The following example is an entry in tkmpirsh.sh for five jobs:

```
export MPICH_PORT_RANGE=18401:18420
```

Edit tkmpirsh.sh using the number of jobs appropriate for your site. (tkmpirsh.sh is located in ***/installation-directory/TKGrid/***.) Then, copy tkmpirsh.sh across the machine cluster or data appliance.

iptables File

This topic lists the complete **/etc/sysconfig/iptables** file. The additions to iptables described in this document are highlighted.

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
# Needed by SAS HPC MC
-A INPUT -m state --state NEW -m tcp -p tcp --dport 10020 -j ACCEPT
# Needed for HDFS (Hadoop)
A INPUT -m state --state NEW -m tcp -p tcp --dport 54310 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 54311 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50470 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50475 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50010 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50020 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50070 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50075 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50090 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50100 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50105 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50030 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 50060 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 15452 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 15453 -j ACCEPT
# End of HDFS Additions
# Needed for LASR Server Ports.
-A INPUT -m state --state NEW -m tcp -p tcp --dport 17401:17405 -j ACCEPT
# End of LASR Additions
# Needed for MPICH.
-A INPUT -m state --state NEW -m tcp -p tcp --dport 18401:18420 -j ACCEPT
# End of MPICH additions.
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
```

Glossary

data set
See SAS data set

encryption
the act or process of converting data to a form that is unintelligible except to the intended recipients.

foundation services
See SAS Foundation Services

grid host
the machine to which the SAS client makes an initial connection in a SAS High-Performance Analytics application.

Hadoop Distributed File System
a framework for managing files as blocks of equal size, which are replicated across the machines in a Hadoop cluster to provide fault tolerance.

HDFS
See Hadoop Distributed File System

identity
See metadata identity

Integrated Windows authentication
a Microsoft technology that facilitates use of authentication protocols such as Kerberos. In the SAS implementation, all participating components must be in the same Windows domain or in domains that trust each other.

Internet Protocol Version 6
See IPv6

IPv6
a protocol that specifies the format for network addresses for all computers that are connected to the Internet. This protocol, which is the successor of Internet Protocol Version 4, uses hexadecimal notation to represent 128-bit address spaces. The format can consist of up to eight groups of four hexadecimal characters, delimited by colons, as in FE80:0000:0000:0000:0202:B3FF:FE1E:8329. As an alternative, a group of consecutive zeros could be replaced with two colons, as in FE80::0202:B3FF:FE1E:8329. Short form: IPv6

IWA
See Integrated Windows authentication

JAR file
a Java Archive file. The JAR file format is used for aggregating many files into one file. JAR files have the file extension .jar.

Java
a set of technologies for creating software programs in both stand-alone environments and networked environments, and for running those programs safely. Java is an Oracle Corporation trademark.

Java Database Connectivity
See JDBC

Java Development Kit
See JDK

JDBC
a standard interface for accessing SQL databases. JDBC provides uniform access to a wide range of relational databases. It also provides a common base on which higher-level tools and interfaces can be built. Short form: JDBC.

JDK
a software development environment that is available from Oracle Corporation. The JDK includes a Java Runtime Environment (JRE), a compiler, a debugger, and other tools for developing Java applets and applications. Short form: JDK.

localhost
the keyword that is used to specify the machine on which a program is executing. If a client specifies localhost as the server address, the client connects to a server that runs on the same machine.

login
a SAS copy of information about an external account. Each login includes a user ID and belongs to one SAS user or group. Most logins do not include a password.

Message Passing Interface
is a message-passing library interface specification. SAS High-Performance Analytics applications implement MPI for use in high-performance computing environments.

metadata identity
a metadata object that represents an individual user or a group of users in a SAS metadata environment. Each individual and group that accesses secured resources on a SAS Metadata Server should have a unique metadata identity within that server.

metadata object
a set of attributes that describe a table, a server, a user, or another resource on a network. The specific attributes that a metadata object includes vary depending on which metadata model is being used.

middle tier
in a SAS business intelligence system, the architectural layer in which Web applications and related services execute. The middle tier receives user requests,

applies business logic and business rules, interacts with processing servers and data servers, and returns information to users.

MPI
See Message Passing Interface

object spawner
a program that instantiates object servers that are using an IOM bridge connection. The object spawner listens for incoming client requests for IOM services. When the spawner receives a request from a new client, it launches an instance of an IOM server to fulfill the request. Depending on which incoming TCP/IP port the request was made on, the spawner either invokes the administrator interface or processes a request for a UUID (Universal Unique Identifier).

planned deployment
a method of installing and configuring a SAS business intelligence system. This method requires a deployment plan that contains information about the different hosts that are included in the system and the software and SAS servers that are to be deployed on each host. The deployment plan then serves as input to the SAS Deployment Wizard.

root node
in a SAS High-Performance Analytics application, the role of the software that distributes and coordinates the workload of the worker nodes. In most deployments the root node runs on the machine that is identified as the grid host. SAS High-Performance Analytics applications assign the highest MPI rank to the root node.

SAS Application Server
a logical entity that represents the SAS server tier, which in turn comprises servers that execute code for particular tasks and metadata objects.

SAS authentication
a form of authentication in which the target SAS server is responsible for requesting or performing the authentication check. SAS servers usually meet this responsibility by asking another component (such as the server's host operating system, an LDAP provider, or the SAS Metadata Server) to perform the check. In a few cases (such as SAS internal authentication to the metadata server), the SAS server performs the check for itself. A configuration in which a SAS server trusts that another component has pre-authenticated users (for example, Web authentication) is not part of SAS authentication.

SAS configuration directory
the location where configuration information for a SAS deployment is stored. The configuration directory contains configuration files, logs, scripts, repository files, and other items for the SAS software that is installed on the machine.

SAS data set
a file whose contents are in one of the native SAS file formats. There are two types of SAS data sets: SAS data files and SAS data views.

SAS Deployment Manager
a cross-platform utility that manages SAS deployments. The SAS Deployment Manager supports functions such as updating passwords for your SAS deployment, rebuilding SAS Web applications, and removing configurations.

SAS Deployment Wizard
a cross-platform utility that installs and initially configures many SAS products. Using a SAS installation data file and, when appropriate, a deployment plan for its initial input, the wizard prompts the customer for other necessary input at the start of the session, so that there is no need to monitor the entire deployment.

SAS Foundation Services
a set of core infrastructure services that programmers can use in developing distributed applications that are integrated with the SAS platform. These services provide basic underlying functions that are common to many applications. These functions include making client connections to SAS application servers, dynamic service discovery, user authentication, profile management, session context management, metadata and content repository access, activity logging, event management, information publishing, and stored process execution.

SAS installation data file
See SID file

SAS installation directory
the location where your SAS software is installed. This location is the parent directory to the installation directories of all SAS products. The SAS installation directory is also referred to as SAS Home in the SAS Deployment Wizard.

SAS IOM workspace
in the IOM object hierarchy for a SAS Workspace Server, an object that represents a single session in SAS.

SAS Metadata Server
a multi-user server that enables users to read metadata from or write metadata to one or more SAS Metadata Repositories.

SAS Pooled Workspace Server
a SAS Workspace Server that is configured to use server-side pooling. In this configuration, the SAS object spawner maintains a collection of workspace server processes that are available for clients.

SAS Software Depot
a file system that consists of a collection of SAS installation files that represents one or more orders. The depot is organized in a specific format that is meaningful to the SAS Deployment Wizard, which is the tool that is used to install and initially configure SAS. The depot contains the SAS Deployment Wizard executable, one or more deployment plans, a SAS installation data file, order data, and product data.

SAS Stored Process Server
a SAS IOM server that is launched in order to fulfill client requests for SAS Stored Processes.

SAS Workspace Server
a SAS IOM server that is launched in order to fulfill client requests for IOM workspaces.

SASHDAT file
the data format used for tables that are added to HDFS by SAS. SASHDAT files are read in parallel by the server.

SASHOME directory
the file location where an instance of SAS software is installed on a computer. The location of the SASHOME directory is established at the initial installation of SAS software by the SAS Deployment Wizard. That location becomes the default installation location for any other SAS software you install on the same machine.

server context
a SAS IOM server concept that describes how SAS Application Servers manage client requests. A SAS Application Server has an awareness (or context) of how it is being used and makes decisions based on that awareness. For example, when a SAS Data Integration Studio client submits code to its SAS Application Server, the server determines what type of code is submitted and directs it to the correct physical server for processing (in this case, a SAS Workspace Server).

server description file
a file that is created by a SAS client when the LASR procedure executes to create a server. The file contains information about the machines that are used by the server. It also contains the name of the server signature file that controls access to the server.

SID file
a control file containing license information that is required in order to install SAS.

spawner
See object spawner

worker node
in a SAS High-Performance Analytics application, the role of the software that receives the workload from the root node.

workspace
See SAS IOM workspace

Index

R

S

U