SAS® High-Performance Analytics Infrastructure 3.9: Installation and Configuration Guide

SAS® Documentation
January 12, 2023

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2018. *SAS® High-Performance Analytics Infrastructure 3.9: Installation and Configuration Guide*. Cary, NC: SAS Institute Inc.

SAS® High-Performance Analytics Infrastructure 3.9: Installation and Configuration Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

January 2023

3.9-P2:hpaicg

Contents

What’s New

What's New in Installation and Configuration for SAS High-Performance Analytics Infrastructure 3.9

Overview

The *SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide* explains how to install and initially configure the SAS High-Performance Analytics infrastructure. This infrastructure consists of the following products:

- SAS High-Performance Computing Management Console 2.9
- SAS Plug-ins for Hadoop, version 1.02
- SAS High-Performance Analytics environment 3.9

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

SAS High-Performance Analytics Infrastructure 3.9 includes the following changes and enhancements:

- “Security Fixes for SAS Plug-ins for Hadoop” on page v
- “New Grid Monitor” on page vi

Security Fixes for SAS Plug-ins for Hadoop

Version 1.02 of SAS Plug-ins for Hadoop contains important security fixes. The changes are compatible with previous SAS 9 and SAS Viya software, with the following exceptions:

- HDFS browsing feature of SAS Visual Analytics Administrator

  SAS Visual Analytics 7.4 sites that want to continue using the HDFS browsing feature in SAS Visual Analytics Administrator must install a hot fix to re-enable HDFS browsing.

- SAS Event Stream Processing file and socket connectors and adapters when used to write SASHDAT files to HDFS

  Contact SAS Technical Support for guidance.

  For more information, see HDAT Subscribe Socket Connector Notes in *SAS Event Stream Processing: Connectors and Adapters*.

New Grid Monitor

There is a new grid monitor console or terminal application, gridmon.sh, that can be run from a Linux terminal or a terminal emulator such as PuTTY. For more information, see Appendix 4, “gridmon.sh Usage and Reference Guide,” on page 87.

Accessibility

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

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 Plug-ins for 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 components:

- SAS High-Performance Computing Management Console, version 2.9
- SAS Plug-ins for Hadoop, version 1.02
- SAS High-Performance Analytics environment, version 3.9

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

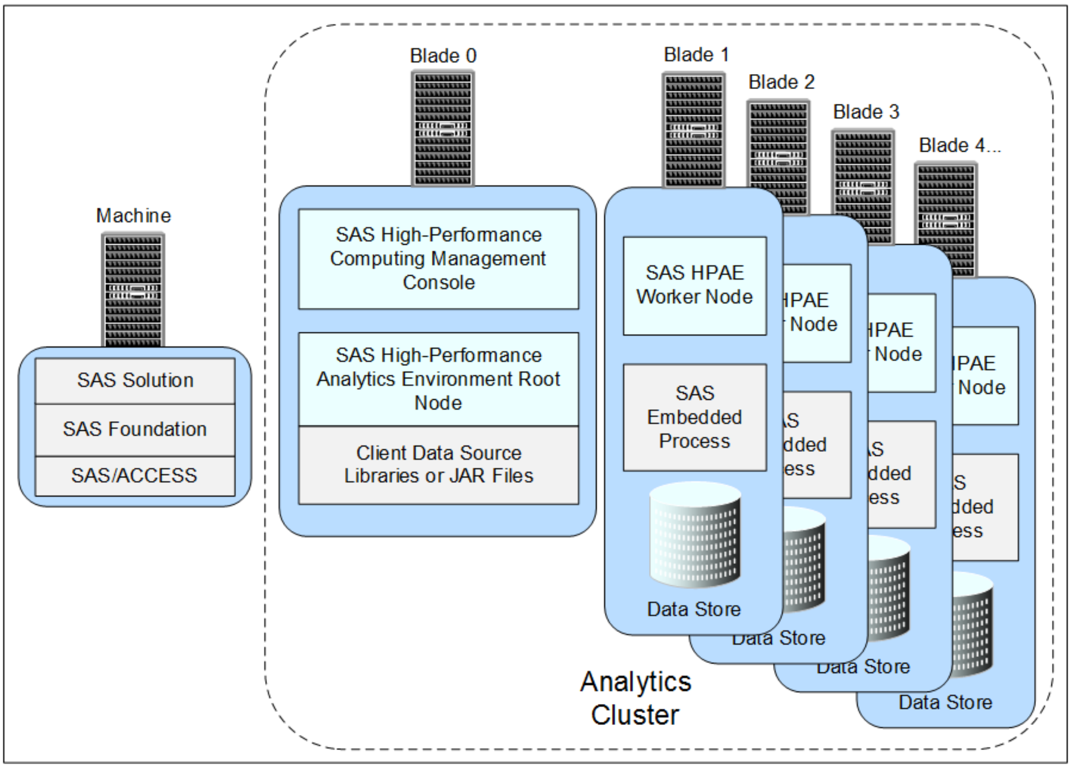
Refer to your order summary to determine the specific version of the infrastructure that is included in your SAS order. Your order summary resides in your SAS Software Depot for your respective order under the **`install_doc`** directory (for example, **`C:\SAS Software Depot\install_doc\my-order\ordersummary.html`**).

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 analytics cluster. The analytics cluster is a Hadoop cluster or a data appliance.

- (Optional) SAS Plug-ins for Hadoop

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

  If you already have one of the supported Hadoop distributions, you can modify it with files from the SAS Plug-ins for Hadoop package. Hadoop modified with SAS Plug-ins for Hadoop enables the SAS High-Performance Analytics environment to write SASHDAT file blocks evenly across the HDFS file system. This even distribution provides a balanced workload across the machines in the cluster and enables SAS analytic processes to read SASHDAT tables very quickly.

  For more information, see “Deploying SAS Plug-ins for Hadoop” on page 39.

- (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 analytics cluster are all made easier by the management console.

Other software on the analytics cluster includes 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 analytics cluster. These components are contained in a deployment package that is specific for your data source.

  For more information, refer to the *SAS and SAS Viya Embedded Process: Deployment Guide* and the *SAS/ACCESS for Relational Databases: Reference*.

  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
  - SAS Customer Intelligence
  - SAS High-Performance Risk
  - SAS Visual Analytics

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 analytics cluster placement:

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

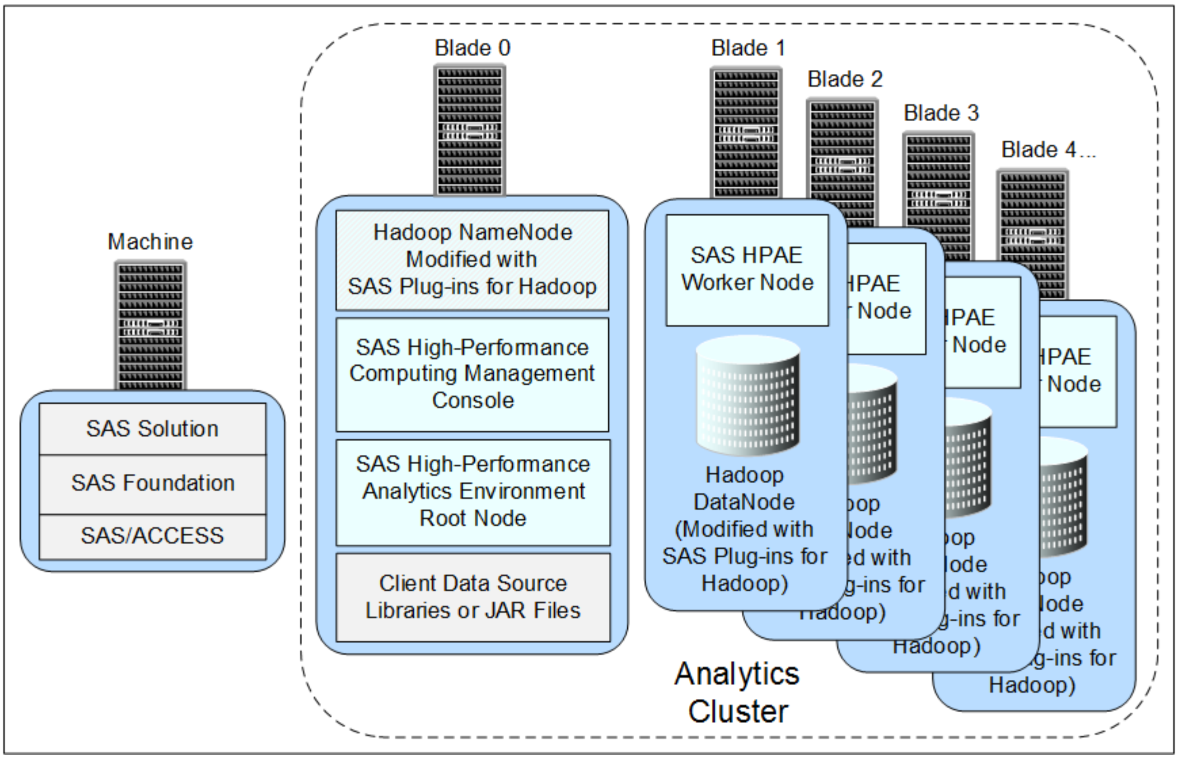
TIP Where you locate your cluster depends on a number of criteria. Your SAS representative will know the latest supported configurations and can work with you to help you determine which cluster placement option works best for your site. Also, there might be solution-specific criteria that you should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

Analytics Cluster Co-Located with Your Hadoop Cluster

Note: In a co-located configuration, the SAS High-Performance Analytics environment supports the Apache, Cloudera, Hortonworks, and MapR distributions of Hadoop. For more specific version information, see the SAS 9.4 Supported Hadoop Distributions.

The following figure shows the analytics cluster co-located on your Hadoop cluster:

Figure 1.2 *Analytics Cluster Co-Located with the Hadoop Cluster*

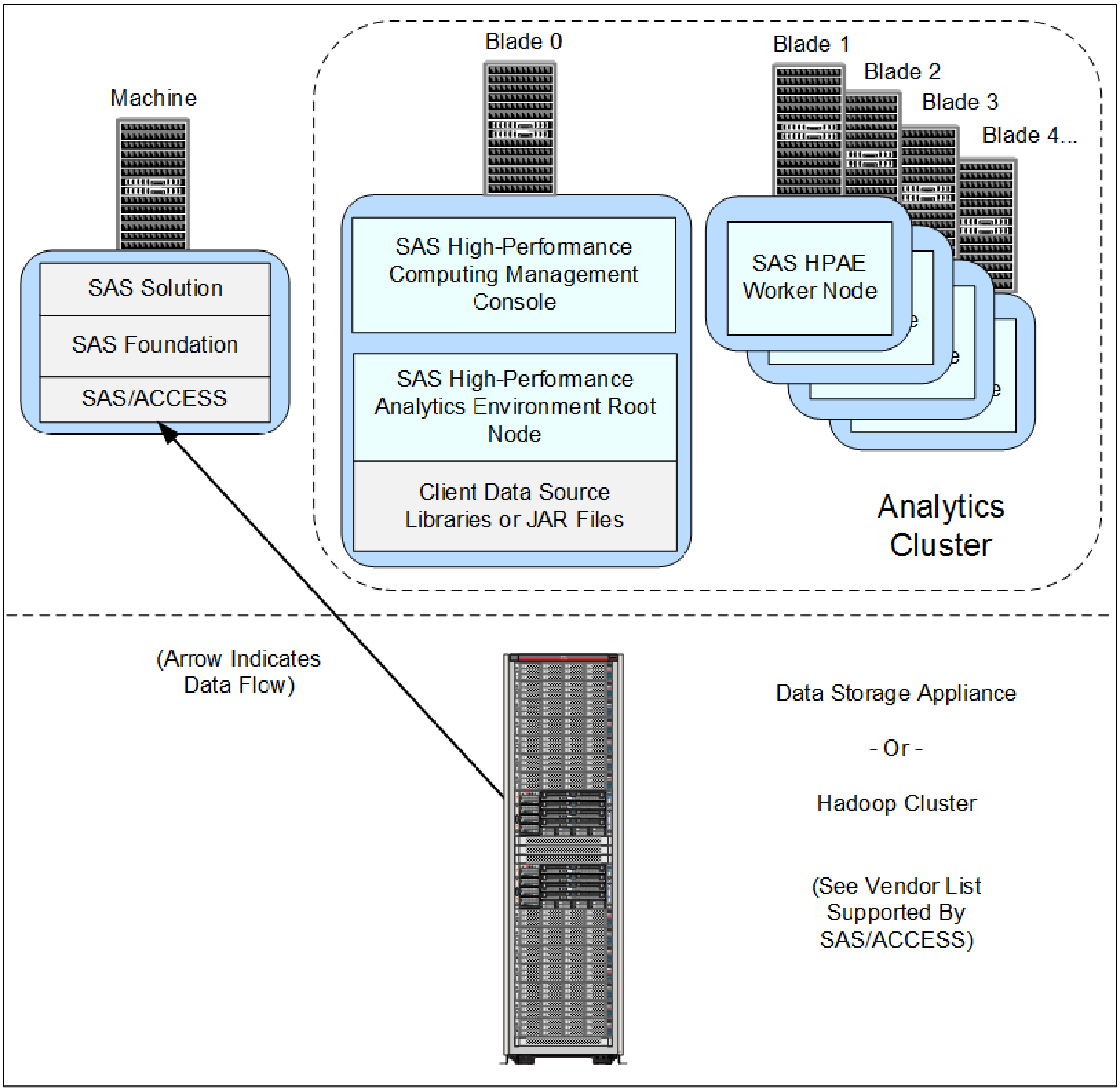

Blade 0
Blade 1
Blade 2
Blade 3
Blade 4 ...
Machine
SAS Solution
SAS Foundation
SAS/ACCESS
Hadoop NameNode Modified with SAS Plug-ins for Hadoop
SAS High-Performance Computing Management Console
SAS High-Performance Analytics Environment Root Node
Client Data Source Libraries or JAR Files
SAS HPAE Worker Node
Hadoop DataNode (Modified with SAS Plug-ins for Hadoop)
Analytics Cluster

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.

Analytics 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)


Machine
SAS Solution
SAS Foundation
SAS/ACCESS
Blade 0
SAS High-Performance Computing Management Console
SAS High-Performance Analytics Environment Root Node
Client Data Source Libraries or JAR Files
Blade 1
Blade 2
Blade 3
Blade 4...
SAS HPAE Worker Node
Analytics Cluster
(Arrow Indicates Data Flow)
Data Storage Appliance
- Or -
Hadoop Cluster
(See Vendor List Supported By SAS/ACCESS)

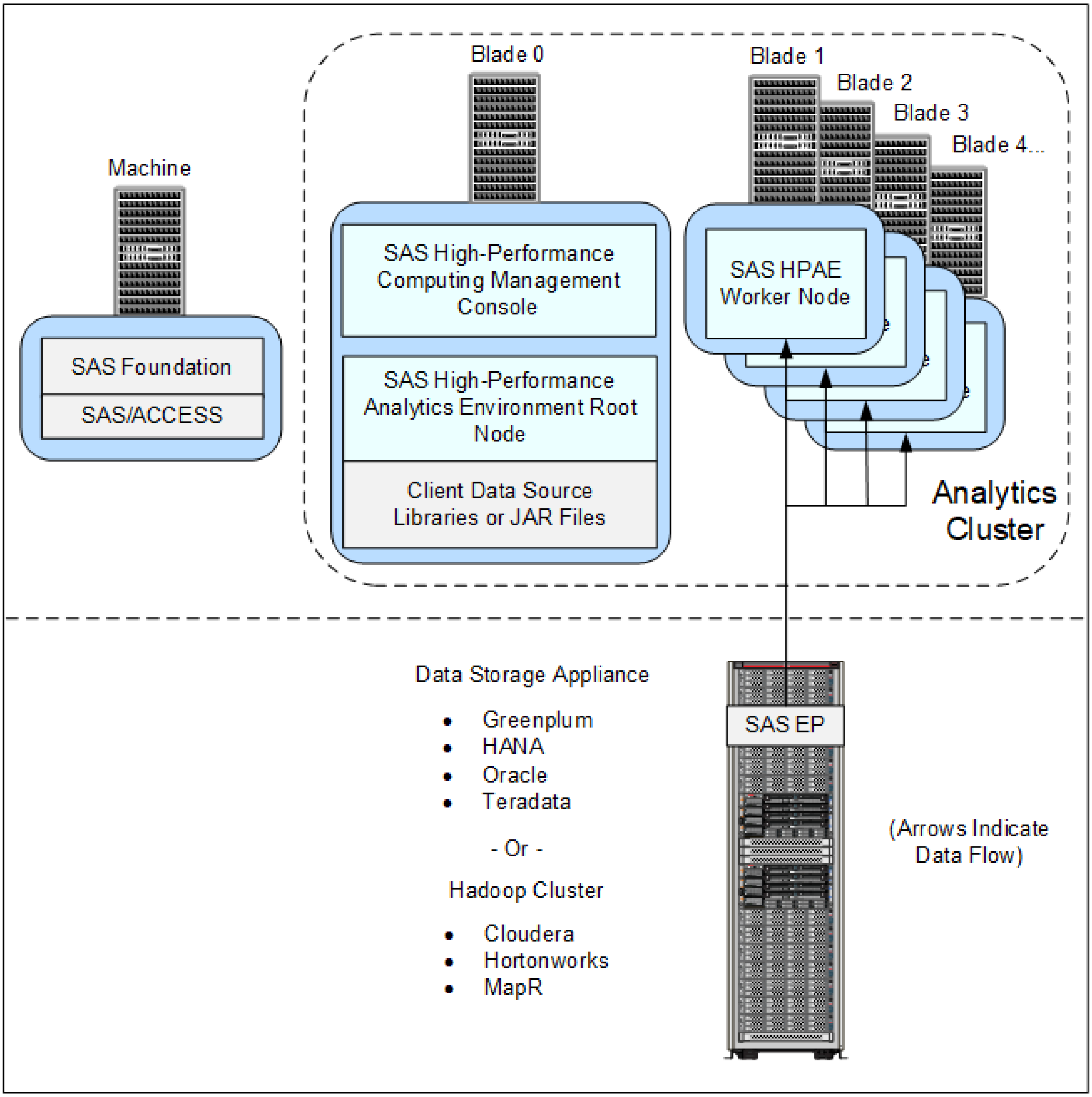
The serial connection between the analytics cluster and your data store is achieved by using the SAS/ACCESS Interface. SAS/ACCESS is orderable in a deployment package that is specific for your data source. For more information, refer to the *SAS/ACCESS for Relational Databases: Reference*.

Analytics Cluster Remote from Your Data Store (Parallel Connection)

Note: The SAS Embedded Process supports the Cloudera, Hortonworks, and MapR distributions of Hadoop. For more specific version information, see the SAS 9.4 Support for Hadoop.

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 analytics cluster. These components are contained in a deployment package that is specific for your data source. For more information, refer to the *SAS and SAS Viya Embedded Process: Deployment Guide*.

Hadoop Deployment Comparison

The following table compares various deployment Hadoop scenarios.

Table 1.1 *Hadoop Deployment Comparison*

	Co-located with Hadoop **SASHDAT: Yes** **SAS Embedded Process: No**	**Co-located with Hadoop** **SASHDAT: No** **SAS Embedded Process: Yes**	**Remote Data Provider** **SASHDAT: Not Supported** **SAS Embedded Process: No**	**Remote Data Provider** **SASHDAT: Not Supported** **SAS Embedded Process: Yes**
SASHDAT Support	Yes	No	No, SASHDAT is co-located or MapR NFS only.	No. SASHDAT is co-located or MapR NFS only.
Parallel R/W	Yes for SASHDAT and CSV. No for SAS/ACCESS because there is no SAS Embedded Process.	Yes. (At least for PROC HDMD.)	No, SAS/ACCESS can perform a serial read through the root node.	Yes. SAS/ACCESS and SAS Embedded Process enable this.
Asymmetric*	No for SASHDAT. No for SAS/ACCESS. SAS/ACCESS can perform a serial read through the root node.	Yes. SAS Embedded Process on all the machines can deliver data to a fewer or greater number of machines.	No. SAS/ACCESS can perform a serial read through the root node.	Yes. SAS Embedded Process on all the machines can deliver data to a fewer or greater number of machines.
Serial Reads for SAS/ACCESS	SAS/ACCESS reads are always serial without SAS Embedded Process.	(If something is misconfigured, SAS/ACCESS performs a serial read.)	SAS/ACCESS reads are always serial without SAS Embedded Process.	(If something is misconfigured, SAS/ACCESS performs a serial read.)
Popularity	This is the SAS Visual Analytics configuration.	Rare.	Rare.	Popular. (Can be combined with a co-located Hadoop configuration.)

* *Asymmetric* refers to a deployment where the total number of SAS High-Performance Analytics environment worker nodes is *not* equal to the total number of Hadoop data nodes. *Symmetric* refers to an equal number of worker nodes and data nodes.

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) Modify co-located Hadoop.
6. Deploy the SAS High-Performance Analytics environment.
7. (Optional) Deploy the SAS Embedded Process for Hadoop.
8. (Optional) Configure the analytics environment for a remote parallel connection

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 *SAS Intelligence Platform: Installation and Configuration Guide*.

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.

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 values. 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 13.

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 27.

Step 5: (Optional) Modify Co-Located Hadoop

If your site wants to use Hadoop as the co-located data store, then you can modify a supported pre-existing Hadoop distribution.

For more information, see Chapter 4, “Modifying Co-Located Hadoop with SAS Plug-ins for Hadoop,” on page 39.

Step 6: 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 5, “Deploying the SAS High-Performance Analytics Environment,” on page 51.

Step 7: (Optional) Deploy the SAS Embedded Process for Hadoop

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 analytics cluster. These components are contained in a deployment package that is specific for your data source.

For information about installing the SAS Embedded Process, see the *SAS and SAS Viya Embedded Process: Deployment Guide*.

Step 8: (Optional) Configure the Analytics Environment for a Remote Parallel Connection

You can optionally configure the SAS High-Performance Analytics Environment for a remote parallel connection.

For more information, see Chapter 6, “Configuring the Analytics Environment for a Remote Parallel Connection,” on page 65.

2

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

Infrastructure Deployment Process Overview

Preparing your analytics cluster is the third of eight 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) Modify co-located Hadoop.

6. Deploy the SAS High-Performance Analytics environment.

7. (Optional) Deploy the SAS Embedded Process for Hadoop.

8. (Optional) Configure the analytics environment for a remote parallel connection.

Install Required Libraries

Two required libraries are not installed by default on Red Hat Enterprise Linux 8.x. Take the following steps before you start the installation:

1. Install ncurses-compat-libs. Run the following command on every machine in the cluster or blade in the data appliance:

   ```
   dnf install ncurses-compat-libs
   ```

2. Install libnsl.so.1. Run the following command on every machine in the cluster or blade in the data appliance:

   ```
   dnf install libnsl
   ```

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:

- Edit the **/etc/sudoers** file to disable the requirement for a terminal:

  ```
  Defaults !requiretty
  ```

  Also, when issuing a sudo command with the simultaneous commands (simcp, simsh) be sure to place `sudo` first. For example:

  ```
  sudo /opt/TKGrid/bin/simsh ls -dl /root
  ```

  These steps address a known bug in Red Hat Linux:

  https://bugzilla.redhat.com/show_bug.cgi?id=1020147

- 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
  ```

- The SAS High-Performance Analytics components require approximately 1.1 GB of disk space. 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.

  It is important to review the system requirements for your SAS solution. Navigate to the SAS 9.4 System Requirements page on the SAS Support website to find system requirements for the following products:

  - □ SAS Visual Analytics and SAS Visual Statistics
  - □ SAS High-Performance Anti-Money Laundering
  - □ SAS High-Performance Customer Link Analytics
  - □ SAS High-Performance Data Mining
  - □ SAS High-Performance Econometrics
  - □ SAS High-Performance Forecasting
  - □ SAS High-Performance Marketing Optimization
  - □ SAS High-Performance Optimization
  - □ SAS High-Performance Risk
  - □ SAS High-Performance Statistics
  - □ SAS High-Performance Text Mining

List the Machines in the Cluster or Appliance

Before the SAS High-Performance Analytics infrastructure can be installed on the analytics cluster, you must create a file named gridhosts that lists all of the machines in the cluster. The SAS High-Performance Analytics environment, SAS Plug-ins for Hadoop, and the SAS High-Performance Computing Management Console all use the gridhosts file for Message Passing Interface (MPI) communication. (The gridhosts file is copied to each machine in the cluster during the installation process. For more information, see Chapter 3, “Deploying SAS High-Performance Computing Management Console,” on page 27.)

TIP You can use SAS High-Performance Computing Management Console to create and manage your gridhosts file. For more information, see the SAS High-Performance Computing Management Console: User's Guide.

During deployment, the installation script uses **/etc/gridhosts** to set up your analytics cluster. As a part of the deployment process, the script creates **TKGrid/grid.hosts** from **/etc/gridhosts**.

After deployment, the SAS High-Performance Analytics environment uses **`TKGrid/grid.hosts`** to manage machines on the cluster, while SAS High-Performance Computing Management Console uses **`/etc/gridhosts`**.

On blade 0, create a file named gridhosts in **`/etc`**. (On Greenplum, blade 0 is known as the Master Server.)

In the gridhosts file, list one machine per line. You can use IP addresses or fully qualified domain names (FQDNs). However, all FQDNs must resolve to IP addresses and must be in the same DNS domain and sub-domain.

CAUTION! The gridhosts file must contain only those machines that are members of your analytics cluster or data appliance. These machines are the NameNode (or Root Node) and its DataNodes (or Worker Nodes). If the management console is located on a machine that is not a member of the analytics cluster, then the console machine must also contain a copy of **`/etc/gridhosts`** with its FQDN added to the list of machines.

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

- Supported Hadoop distributions: NameNode (blade 0)
- Greenplum Data Computing Appliance: Master Server

Here is an example of a gridhosts file:

```
machine001.example.com
machine002.example.com
machine003.example.com
machine004.example.com
...
```

Note: Make sure that there are no whitespace characters in your gridhosts file. The SAS High-Performance Analytics environment can skip entries when it encounters whitespace characters (such as tabs).

Review Passwordless Secure Shell (SSH) Requirements

TIP If you are not familiar with passwordless Secure Shell (SSH), please see Appendix 6, "Setting Up Passwordless Secure Shell (SSH)," on page 101.

Secure Shell (SSH) has the following requirements:

- To support Kerberos, enable Generic Security Services Application Programming Interface (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 20 .

- Passwordless Secure Shell (SSH) is required on all machines in the cluster or on the data appliance.

  Note the following items:

- - If your SAS compute server is on a separate system from the SAS High-Performance Analytics environment name node, then you need to copy the ~/.ssh directory on the compute server machine as well.
  - Passwordless SSH must be bi-directional between all of the analytics environment worker nodes.
- The following user accounts require passwordless SSH:
  - 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 “Requirements for Co-Located Hadoop” on page 22.
  - SAS High-Performance Analytics environment user account

    Passwordless SSH must be configured for every user that runs a SAS High-Performance procedure or interacts with SAS LASR Analytic Server or SAS data (sashdat format).

    For more information about the environment’s user accounts, see “Preparing to Deploy the SAS High-Performance Analytics Environment” on page 23.

    TIP Users’ home directories must be located in the same directory on each machine in the analytics cluster. For example, you will experience problems if user foo has a home directory at `/home/foo` on blade one and blade two, and a home directory at `/mnt/user/foo` on blade three.

Configure Client Connections to the Analytics Environment

Set GRIDRSHCOMMAND for your SAS programs on the client:

- SAS programs (OPTIONS statement):

  ```
  options set=GRIDRSHCOMMAND="/path-to-file/ssh -q -o
  StrictHostKeyChecking=no";
  ```

  Here is an example on Linux:

  ```
  options set=GRIDRSHCOMMAND="/usr/bin/ssh -q -o
  StrictHostKeyChecking=no";
  ```
- SAS configuration files (SET statement):

  ```
  -SET GRIDRSHCOMMAND "/path-to-file/ssh -q -o
  StrictHostKeyChecking=no"
  ```

  Here is an example on Linux:

```
-SET GRIDRSHCOMMAND "/usr/bin/ssh -q -o
StrictHostKeyChecking=no"
```

Note: Windows does not ship with SSH. If your site does not have SSH, you must download and install SSH from a website like Cygwin or PuTTY.

TIP Adding GRIDRSHCOMMAND to your sasv9_usermods.cfg file preserves the setting during SAS upgrades and avoids having to manually set that environment variable on the client before starting SAS.

For more information, see Appendix 3, “SAS High-Performance Analytics Environment Client-Side Environment Variables,” on page 85.

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 that 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.

The list of Kerberos prerequisites is:

- 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: Example

This topic provides an example of setting up hosts using MIT Kerberos. There are other implementations of Kerberos, such as Microsoft Active Directory, that the SAS High-Performance Analytics infrastructure supports.

Every machine in the analytics 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 +ok_to_delegate 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 contains only 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 Validate your configuration in a temporary credential cache (ccache) to avoid overwriting any ccache in your user session with the host's credentials:

```
kinit -kt /etc/krb5.keytab -c ~/testccache host/
machine.name@REALM.NAME
```

6 Because kinit obtains only a krbtgt ticket for a given principal, also validate that Kerberos is able to issue service tickets for the host principal:

```
kvno -c ~/testccache machine.name@REALM.NAME
```

7 Run the `klist` command to check the status of your Kerberos ticket:

```
klist -efac ~/testccache
```

Your klist output should resemble the following:

```
Ticket cache: FILE:/home/myacct/testccache
Default principal: host/myserver.example.com@NA.EXAMPLE.COM

Valid starting     Expires            Service principal
07/07/15 15:33:32  07/08/15 01:33:32  krbtgt/NA.EXAMPLE.COM@NA.EXAMPLE.COM
    renew until 07/14/15 15:33:32, Flags: FRIA
    Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: (none)
07/07/15 15:34:09  07/08/15 01:33:32  host/myserver.example.com@NA.EXAMPLE.COM
    renew until 07/14/15 15:33:32, Flags: FRAO
    Etype (skey, tkt): aes256-cts-hmac-sha1-96, aes256-cts-hmac-sha1-96
    Addresses: (none)
```

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.

8 Delete your ccache:

```
kdestroy -c ~/testccache
```

Configure Passwordless SSH to Use Kerberos

TIP If you are not familiar with passwordless Secure Shell (SSH), please see Appendix 6, "Setting Up Passwordless Secure Shell (SSH)," on page 101.

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 start-up, the Message Passing Interface (MPI) sends a user's Kerberos credentials cache (KRB5CCNAME) that 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
```

Note: Enter the above option on one line. Do not add any carriage returns or other whitespace characters.

For more information, see Table 5.2 on page 56.

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";
```

TIP Adding GRIDRSHCOMMAND to your sasv9_usermods.cfg preserves the setting during SAS upgrades and avoids having to manually set that environment variable on the client before starting SAS.

Third-party Kerberos libraries can change the default Kerberos library that TKSSH_GSSAPI is using. To prevent this from happening, make sure that you set the TKSSH_GSSAPI_LIB environment variable to SECUR32, which forces TKSSH_GSSAPI to use Windows single sign-on:

```
set=TKSSH_GSSAPI_LIB ="SECUR32";
```

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 from 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 15.
- Locate the software.

  Make sure that your SAS Software Depot has been created. (For more information, see "Creating a SAS Software Depot" in *SAS Intelligence Platform: Installation and Configuration Guide*.)

Requirements for Co-Located Hadoop

If you already have one of the supported Hadoop distributions, you can modify it with files from the SAS Plug-ins for Hadoop package. Hadoop modified with SAS Plug-ins for Hadoop enables the SAS High-Performance Analytics environment to write SASHDAT file blocks evenly across the HDFS file system.

The following is required for existing Hadoop clusters with which the SAS High-Performance Analytics environment can be co-located:

- Your Hadoop distribution must be supported.
- Each machine in the cluster must be able to resolve the host name of all the other machines.
- The machine configured as the NameNode cannot also be configured as a DataNode.
- These Hadoop directories must reside on local storage:
  - the directory on the file system where the Hadoop NameNode stores the namespace and transactions logs persistently
  - the directory on the file system where temporary MapReduce data is written
  - the directory on the file system where the MapReduce framework writes system files

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

Preparing to Deploy the SAS High-Performance Analytics Environment

If you are using Kerberos, see also "Preparing for Kerberos" on page 18.

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 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 analytics 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 *SAS Intelligence Platform: Data Administration Guide*.)

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 61.

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 the *SAS and SAS Viya Embedded Process: Deployment Guide*.

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.

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 uses 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 actually use.

We recommend that you document each SAS port you reserve in the following standard location on each machine: `/etc/services`. This practice helps to 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 2.1 Pre-installation Checklist for SAS Ports

SAS Component	Default Port	Data Direction	Actual Port
SAS High-Performance Computing Management Console server	10020	Inbound	
SAS Plug-in on the NameNode	15452	Inbound	
SAS Plug-in on the DataNode	15453	Inbound	

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 eight 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) Modify co-located Hadoop.

6. Deploy the SAS High-Performance Analytics environment.

7. (Optional) Deploy the SAS Embedded Process for Hadoop.

8. (Optional) Configure the analytics environment for a remote parallel connection.

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 Analytics Server, passwordless SSH is required to run jobs on the machines in the cluster.

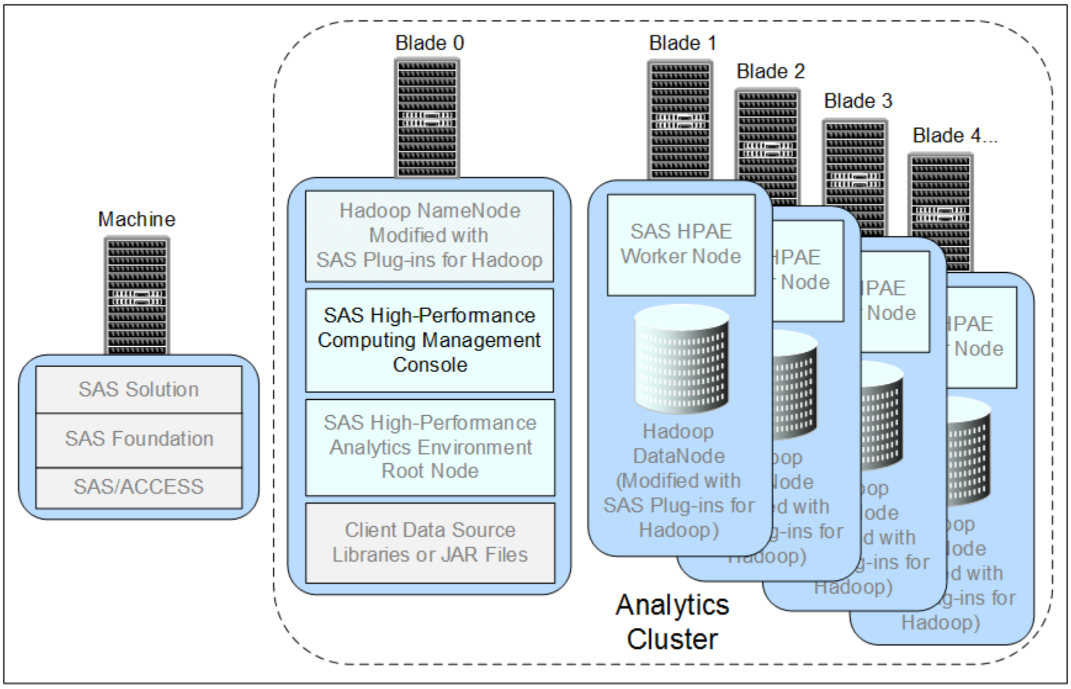
Also, users of some SAS solutions must have an operating system (external) account on all the machines in the cluster and their keys must be distributed across the cluster. For more information, see "Create the First User Account and Propagate the SSH Key" on page 34.

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 the SAS High-Performance Computing Management Console: User's Guide.

Overview of Deploying the Management Console

The SAS High-Performance Computing Management Console is deployed on the machine where the SAS High-Performance Analytics environment is deployed. In this document, that machine is blade 0.

Figure 3.1 Management Console Deployed with a Hadoop Cluster


Machine
SAS Solution
SAS Foundation
SAS/ACCESS
Blade 0
Hadoop NameNode Modified with SAS Plug-ins for Hadoop
SAS High-Performance Computing Management Console
SAS High-Performance Analytics Environment Root Node
Client Data Source Libraries or JAR Files
Blade 1
Blade 2
Blade 3
Blade 4...
SAS HPAE Worker Node
Hadoop DataNode (Modified with SAS Plug-ins for Hadoop)
Analytics Cluster

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 Appendix 1, "Updating the SAS High-Performance Analytics Infrastructure," on page 79.

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_9/Linux_for_x64`** directory.

4 Enter one of the following commands:

- To install in the default location of **`/opt`**:

```
rpm -ivh sashpcmc*.rpm
```

- To install in a location of your choice:

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

where *directory* is an absolute path where you want to install the console.

5 Proceed to the topic "Configure the Management Console" on page 30.

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_9/Linux_for_x64`** directory.

4 Copy sashpcmc-2.8.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-2.8.tar.gz
```

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

6 Proceed to the topic "Configure the Management Console" on page 30.

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 30. 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.

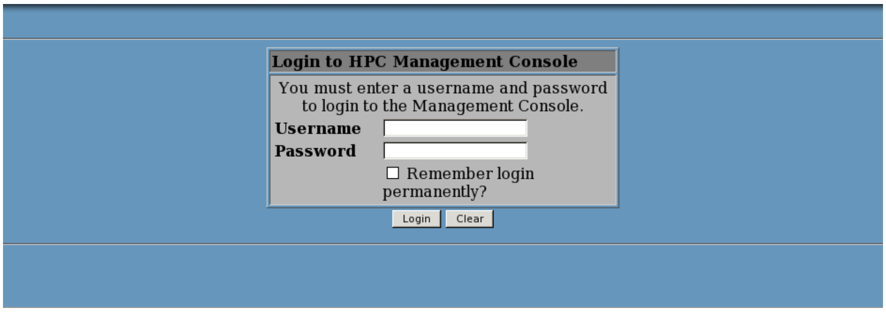
7 To start the SAS High-Performance Computing Management Console server, enter the following command from any directory:

```
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.



Login to HPC Management Console

You must enter a username and password to login to the Management Console.

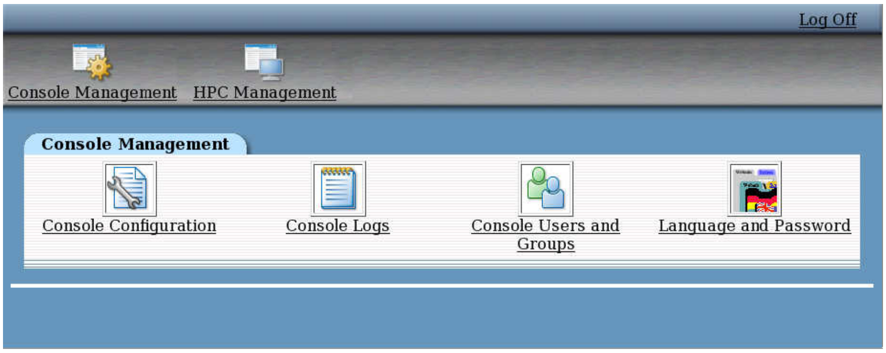
Username

Password

□ Remember login permanently?

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:

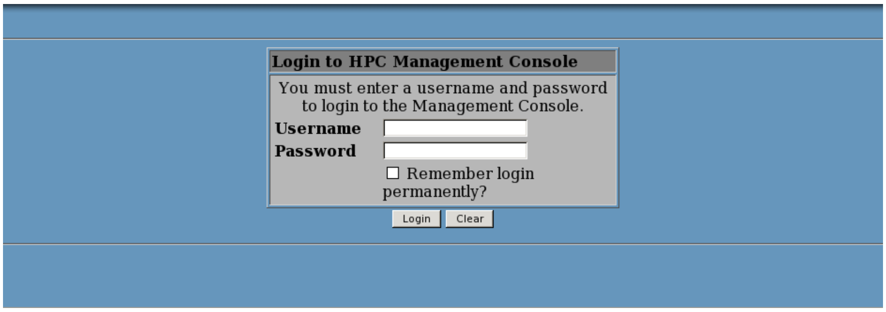
```
service sashpcmc status
```

(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 30.

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.



Login to HPC Management Console

You must enter a username and password to login to the Management Console.

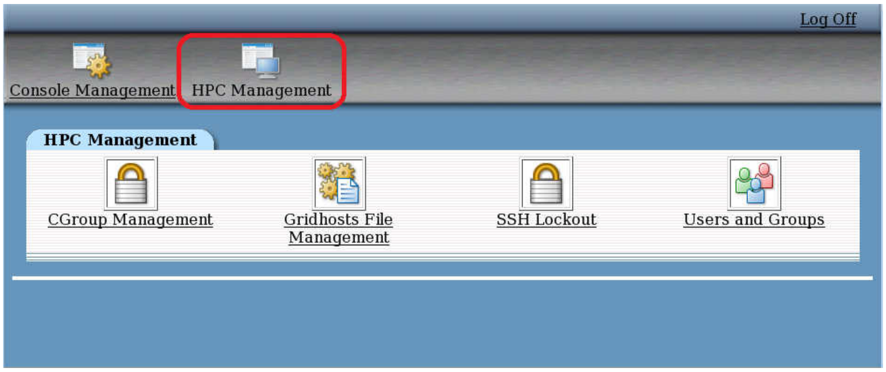
Username

Password

□ Remember login permanently?

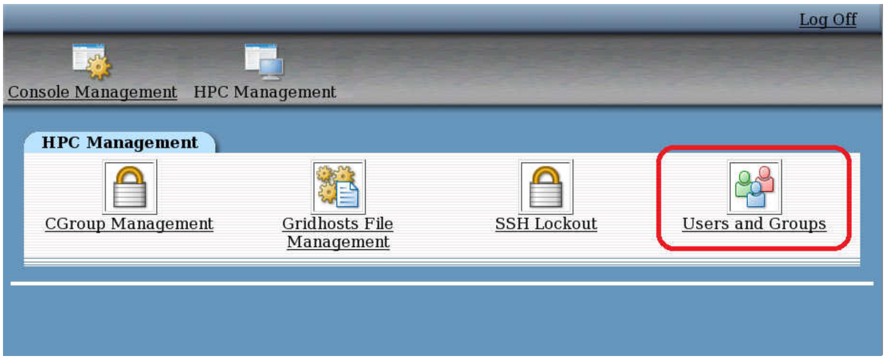
3 Log on to SAS High-Performance Computing Management Console.

The Console Management page appears.



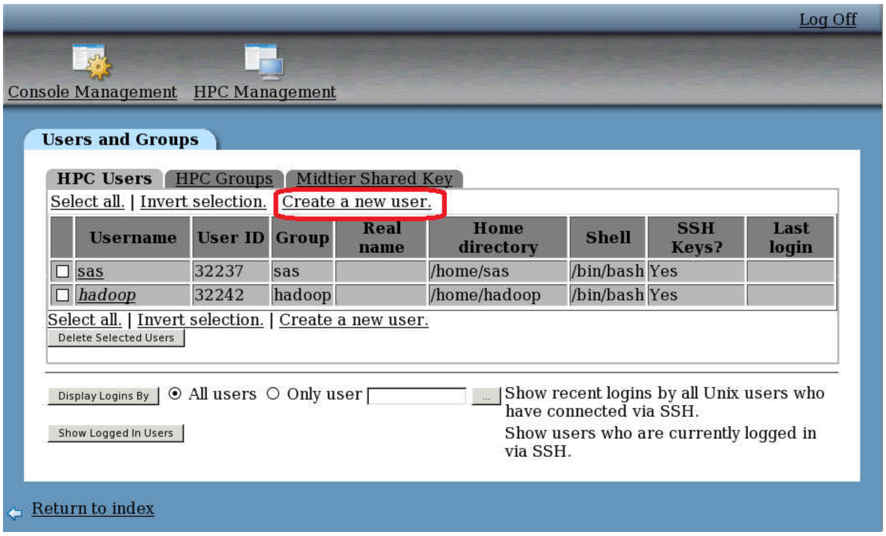
4 Click **HPC Management**.

The HPC Management page appears.



5 Click **Users and Groups**.

The Users and Groups page appears.



Log Off

Console Management HPC Management

Users and Groups

HPC Users | HPC Groups | Midtier Shared Key

Select all. | Invert selection. | Create a new user.

	Username	User ID	Group	Real name	Home directory	Shell	SSH Keys?	Last login
□	sas	32237	sas		/home/sas	/bin/bash	Yes	
□	*hadoop*	32242	hadoop		/home/hadoop	/bin/bash	Yes	

Select all. | Invert selection. | Create a new user.

Delete Selected Users

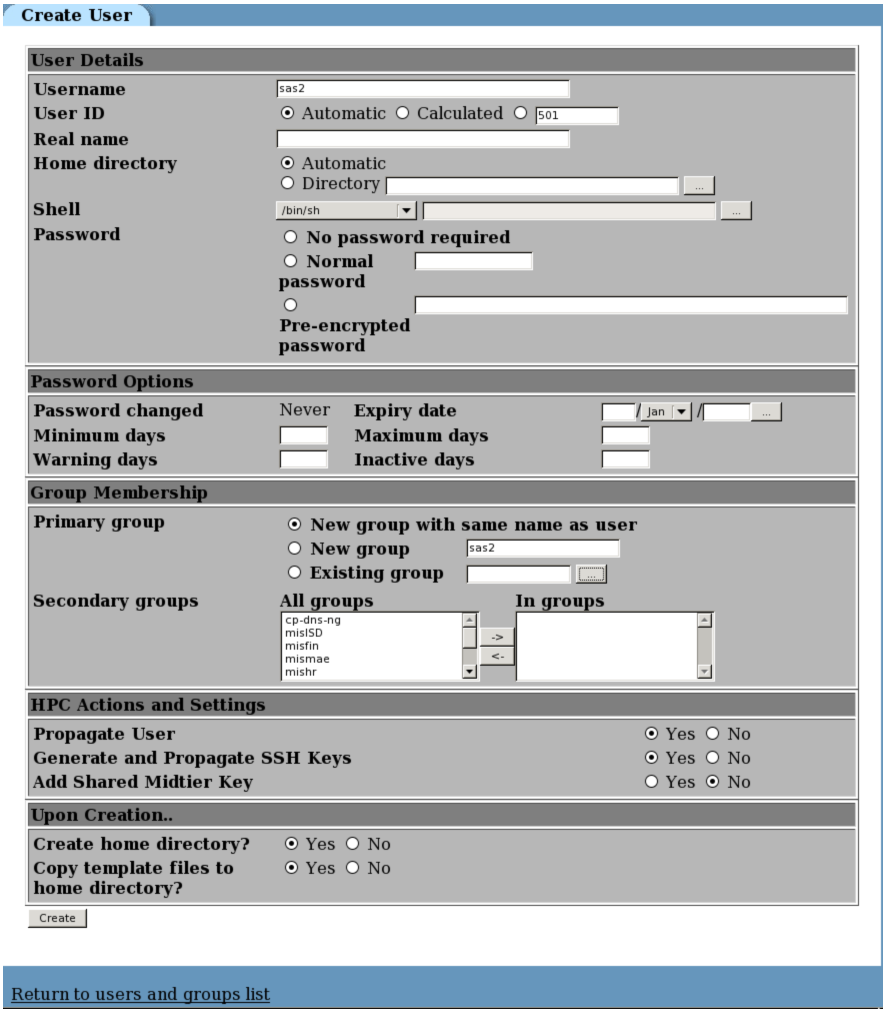
Display Logins By ● All users ○ Only user [ ] ... Show recent logins by all Unix users who have connected via SSH.

Show Logged In Users Show users who are currently logged in via SSH.

Return to index

6 Click **Create a new user**.

The Create User page appears.



Create User

User Details

Username	sas2
User ID	● Automatic ○ Calculated ○ 501
Real name	
Home directory	● Automatic ○ Directory [ ] ...
Shell	/bin/sh [ ] ...
Password	○ No password required ○ Normal password [ ] ○ Pre-encrypted password [ ]

Password Options

Password changed	Never	Expiry date	[ ] / Jan / [ ] ...
Minimum days		Maximum days	
Warning days		Inactive days	

Group Membership

Primary group	● New group with same name as user ○ New group sas2 ○ Existing group [ ] ...
Secondary groups	All groups: cp-dns-ng, misISD, misfin, mismae, mishr -> <- In groups:

HPC Actions and Settings

Propagate User	● Yes ○ No
Generate and Propagate SSH Keys	● Yes ○ No
Add Shared Midtier Key	○ Yes ● No

Upon Creation..

Create home directory?	● Yes ○ No
Copy template files to home directory?	● Yes ○ No

Create

Return to users and groups list

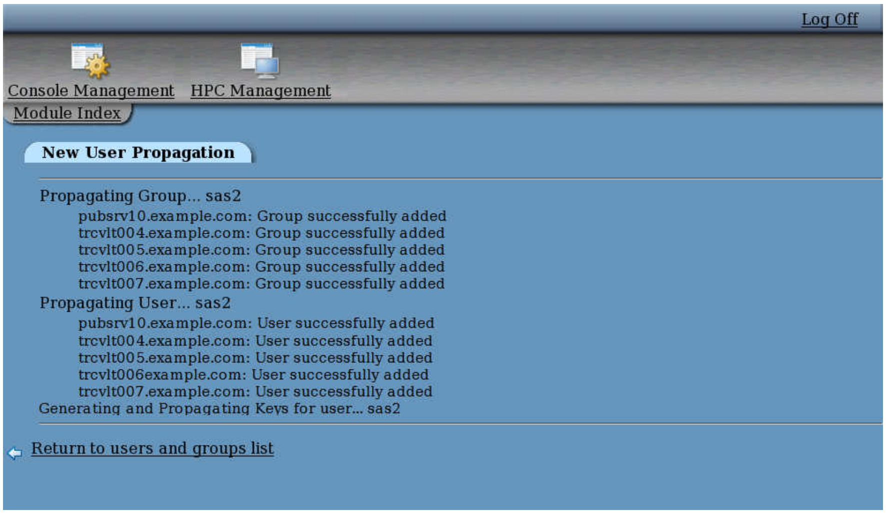
7 Enter information for the new user, using the security policies in place at your site.

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.



```
Propagating Group... sas2
      pubsrv10.example.com: Group successfully added
      trcvlt004.example.com: Group successfully added
      trcvlt005.example.com: Group successfully added
      trcvlt006.example.com: Group successfully added
      trcvlt007.example.com: Group successfully added
Propagating User... sas2
      pubsrv10.example.com: User successfully added
      trcvlt004.example.com: User successfully added
      trcvlt005.example.com: User successfully added
      trcvlt006example.com: User successfully added
      trcvlt007.example.com: User successfully added
Generating and Propagating Keys for user... sas2
```

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:

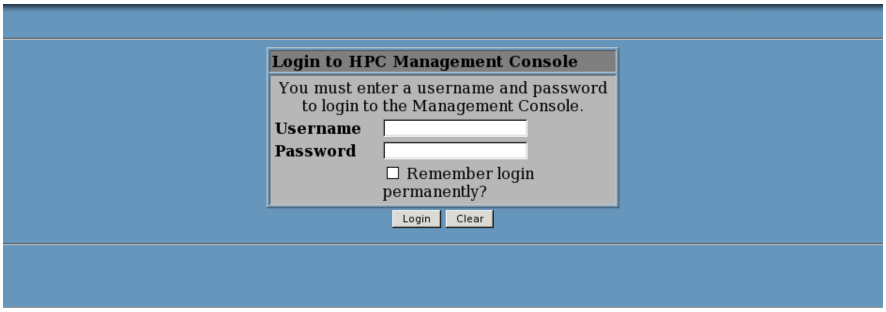
```
service sashpcmc status
```

For more information, see To start the SAS High-Performance Computing Management Console server on page 30.

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.



Login to HPC Management Console

You must enter a username and password to login to the Management Console.

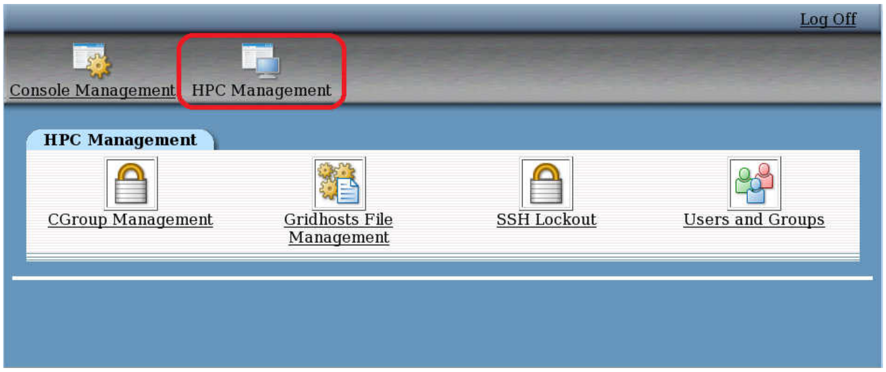
Username

Password

□ Remember login permanently?

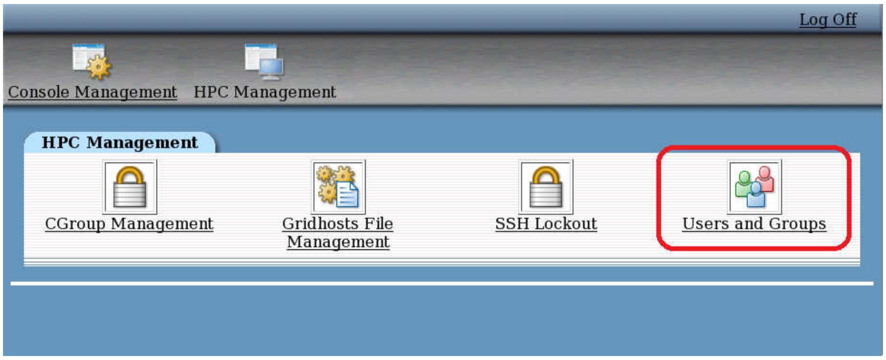
3 Log on to SAS High-Performance Computing Management Console.

The Console Management page appears.



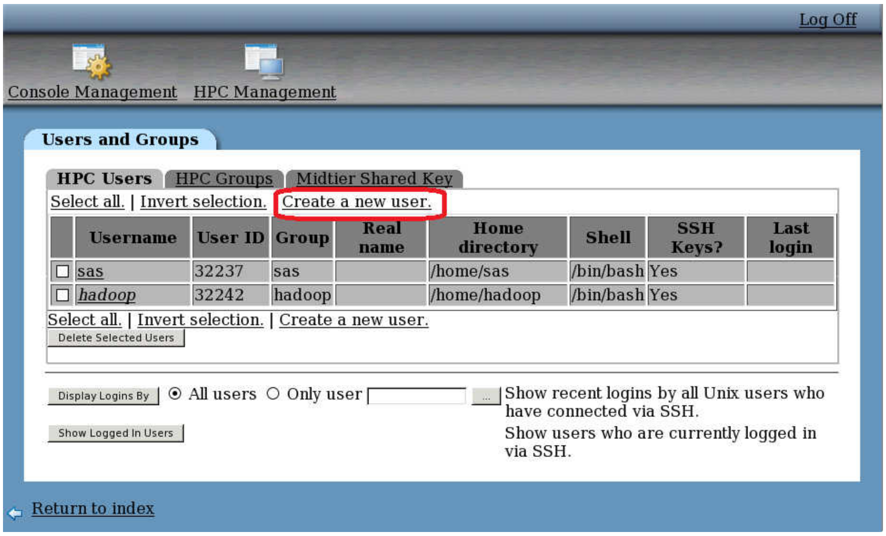
4 Click **HPC Management**.

The Console Management page appears.



5 Click **Users and Groups**.

The Users and Groups page appears.



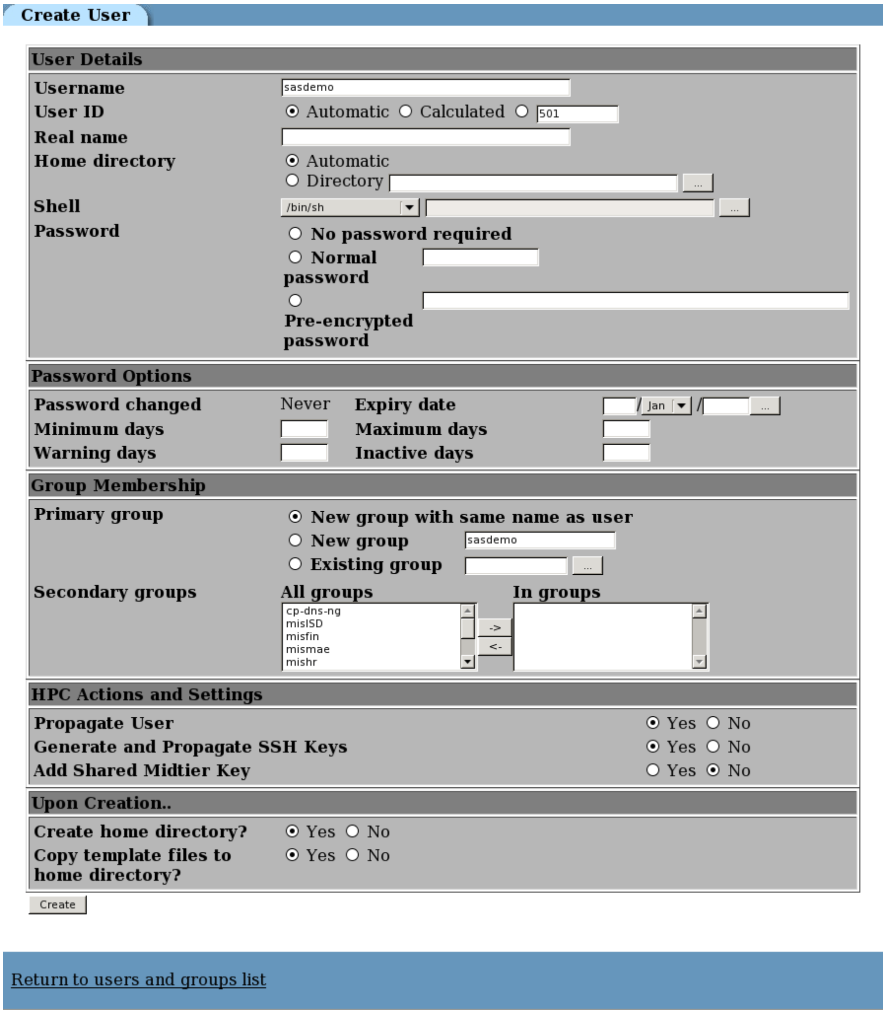
	Username	User ID	Group	Real name	Home directory	Shell	SSH Keys?	Last login
□	sas	32237	sas		/home/sas	/bin/bash	Yes	
□	hadoop	32242	hadoop		/home/hadoop	/bin/bash	Yes	

Display Logins By ● All users ○ Only user ... Show recent logins by all Unix users who have connected via SSH.

Show Logged In Users Show users who are currently logged in via SSH.

6 Click **Create a new user**.

The Create User page appears.



User Details

Username	sasdemo
User ID	● Automatic ○ Calculated ○ 501
Real name	
Home directory	● Automatic ○ Directory ...
Shell	/bin/sh ...
Password	○ No password required ○ Normal password ○ Pre-encrypted password

Password Options

Password changed	Never	Expiry date	/ Jan / ...
Minimum days		Maximum days	
Warning days		Inactive days	

Group Membership

Primary group	● New group with same name as user ○ New group sasdemo ○ Existing group ...
Secondary groups	All groups: cp-dns-ng, misISD, misfin, mismae, mishr -> <- In groups

HPC Actions and Settings

Propagate User	● Yes ○ No
Generate and Propagate SSH Keys	● Yes ○ No
Add Shared Midtier Key	○ Yes ● No

Upon Creation..

Create home directory?	● Yes ○ No
Copy template files to home directory?	● Yes ○ No

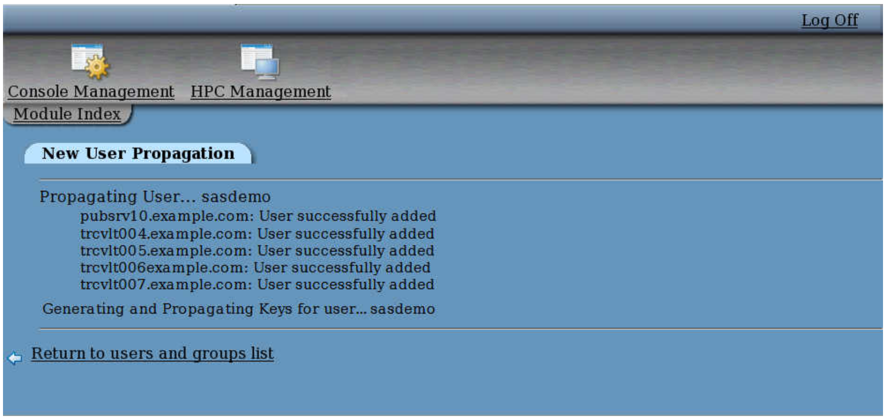
7 Enter information for the new user, using the security policies in place at your site.

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.



```
Propagating User... sasdemo
        pubsrv10.example.com: User successfully added
        trcvlt004.example.com: User successfully added
        trcvlt005.example.com: User successfully added
        trcvlt006example.com: User successfully added
        trcvlt007.example.com: User successfully added
Generating and Propagating Keys for user... sasdemo
```

4

Modifying Co-Located Hadoop with SAS Plug-ins for Hadoop

Infrastructure Deployment Process Overview

Modifying a co-located Hadoop cluster is an optional fifth of eight 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) Modify co-located Hadoop.**

6. Deploy the SAS High-Performance Analytics environment.

7. (Optional) Deploy the SAS Embedded Process for Hadoop.

8. (Optional) Configure the analytics environment for a remote parallel connection.

Deploying SAS Plug-ins for Hadoop

Overview of Deploying SAS Plug-ins for Hadoop

To deploy SAS Plug-ins for Hadoop:

1 Ensure that Python is installed on all nodes where you intend to install the SAS plug-ins for Hadoop.

2 Copy the SAS Plug-ins for Hadoop files to the Hadoop cluster.

3 Install SAS Plug-ins for Hadoop to the Hadoop NameNode and to each DataNode by using one of the following methods:

   a the sashdat-install.sh script (supplied by SAS)

   Note: The sashdat-install.sh script installs SAS Plug-ins for Hadoop on all supported Hadoop distributions.

   b parcel with Cloudera Manager

   c stack with Ambari

4 Configure HDFS Service properties.

Copying SAS Plug-ins for Hadoop Files to the Hadoop Cluster

Follow these steps to copy SAS Plug-ins for Hadoop files to the Hadoop cluster:

1 The software that is needed for SAS Plug-ins for Hadoop is available from within the SAS Software Depot that was created by your site's depot administrator:

   ***depot-installation-location*/standalone_installs/SAS_Plug-ins_for_Hadoop/1_02/Linux_for_x64/hdatplugins.tar.gz**

2 Copy the hdatplugins.tar.gz file to a temporary location on one of the following:

   - the NameNode host of the Hadoop cluster (to use the sashdat-install.sh script)
   - the Cloudera Manager host of the Cloudera Hadoop cluster
   - the Ambari Server host of the Hortonworks Hadoop cluster for Ambari

   and untar it:

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

   A directory that is named **/tmp/hdatplugins** is created.

   Note: Ensure that the file permissions are set to 0755.

3 Go to one of the following sections:

   a "sashdat-install.sh" on page 41

   b "Cloudera Manager" on page 41

   c "Ambari" on page 42

Installing SAS Plug-ins for Hadoop

Depending on your Hadoop distribution, you can install SAS Plug-ins for Hadoop by using sashdat-install.sh, Cloudera Manager, or Ambari.

sashdat-install.sh

You can use the sashdat-install.sh script that is supplied by SAS to install SAS Plug-ins for Hadoop on all supported Hadoop distributions.

1 Log on to the Hadoop NameNode machine (blade 0) with a UNIX account that has sudo privileges and passwordless SSH access to every machine in the Hadoop cluster.

2 Change to the directory that was specified in Step 2 and run the sashdat-install.sh script using one of the following commands:

   a Deploy by using the ‘hdfs’ account to query the hdfs service for the list of machines:

   sashdat-install.sh -add

   Here is an example:

   ```
   ./sashdat-install.sh -add
   ```

   b Deploy by supplying your own list of machines:

   sashdat-install.sh -add -hostfile *host-list-filename*

   or:

   sashdat-install.sh -add -host “*list of hosts*”

   Here are examples:

   ```
   ./sashdat-install.sh -add -hostfile /tmp/my_hosts
   ```

   ```
   ./sashdat-install.sh -add -host "host1, host2, host3"
   ```

   c Deploy by specifying a different parent installation path:

   ```
   ./sashdat-install.sh -add -hdathome /opt/my_path/
   ```

   For more information, see “sashdat-install.sh Reference”.

Cloudera Manager

You can use Cloudera Manager with parcel to install SAS Plug-ins for Hadoop on all supported Cloudera Hadoop distributions.

1 On the Hadoop Cloudera Manager host, navigate to the **/tmp/hdatplugins/parcel/** directory.

2 Run the following script.

   Note: The user account that you use to run the script must have super user (sudo) or root access.

   Here is an example:

   ```
   install_parcel.sh -v redhat6
   ```

3 Log on to Cloudera Manager as administrator.

4 Activate the parcel.

   a Click **Distribute** to copy the parcel to all nodes.

   b Click **Activate**. You are prompted to restart the cluster or to close the window.

   c When prompted, click **Close**.

   CAUTION! Do not restart the cluster.

Ambari

You can use Ambari with stack to install SAS Plug-ins for Hadoop on all supported Cloudera Hadoop distributions.

Note: The following deployment steps assume that the hdatplugins rpm package is installed directly on one of the following machines:

1 the Ambari server

2 a machine in the network that is accessible to the Ambari server

CAUTION! When the Hortonworks Hadoop stack is upgraded, the HDATPlugins stack must be deactivated and then reactivated. If the Hortonworks Hadoop level is upgraded in **Express** mode on Ambari, the HDATPlugins stack must be restarted. If the Hortonworks Hadoop level is upgraded in **Rolling** mode, a restart of the HDATPlugins stack is not required.

Follow these steps to install SAS Plug-ins for Hadoop using Ambari with stack:

1 To launch the script, on the Hadoop Ambari Server host, navigate to the **/tmp/hdatplugins/stack/** directory and run the following command:

```
./install_hdatplugins.sh Ambari-admin-username
```

   After the script finishes running, this message is displayed: `You can install the HDATPLUGINS stack now from Ambari Cluster Manager.`

2 Log on to Ambari. On the Ambari server, deploy the services as follows:

   a Click **Actions** and select **+ Add Service**. The Add Service Wizard page and the Choose Services panel open.

   b In the Choose Services panel, select **SASHDAT**. Click **Next**. The Assign Slaves and Clients panel opens.

   c In the Assign Slaves and Clients panel under **Client** , select all data nodes and all name nodes where you want the stack to be deployed. The Customize Services panel opens. The SASHDAT stack is listed.

   d Do not change any settings on the Customize Services panel. Click **Next**.

   Note: If your cluster is secured with Kerberos, the Configure Identities panel opens. Enter your Kerberos credentials in the **admin_principal** text box and the **admin_password** text box. Click **Next**. The Review panel opens.

e. Review the information in the panel. If the values are correct, click **Deploy**. The Install, Start, and Test panel opens. After the stack is installed on all nodes, click **Next**. The Summary panel opens.

f. Click **Complete**. The stacks are now installed on all nodes of the cluster. SASHDAT is displayed on the Ambari dashboard.

g. On every node, all files in the **`/usr/hdp/`*`Hadoop-version`*`/hadoop/bin`** directory must be executable with file permissions of 755.

Configuring HDFS Service Properties

Configure HDFS service properties for SAS Plug-ins for Hadoop based on your Hadoop distribution.

Cloudera Hadoop

Note: If Cloudera Manager provides a choice between classic and new layouts, use classic layout.

To use Cloudera Manager to configure HDFS service properties for SAS Plug-ins for Hadoop, perform the following steps:

1. Log on to Cloudera Manager as an administrator.

2. Search for the string `hdfs_service_config_safety_valve` in the HDFS configuration filter. Or navigate to the Service-Wide group.

   Under **Advanced**, add the following lines to the HDFS Service Advanced Configuration Snippet (Safety Valve) for the hdfs-site.xml property.

   Note: Specify each property on a single line. Multiple lines are used here to improve readability.

   ```
   <property><name>dfs.namenode.plugins</name> <value>com.sas.cas.hadoop.NameNodeService
        </value></property>
   <property><name>dfs.datanode.plugins</name> <value>com.sas.cas.hadoop.DataNodeService
        </value></property>
   <property><name>com.sas.cas.hadoop.service.namenode.port</name> <value>15452
        </value></property>
   <property><name>com.sas.cas.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>com.sas.cas.hadoop.short.circuit.command</name>
   <value>/opt/sas/HDATHome/bin/sascasfd</value></property>
   ```

   Note: You can change the port for the SAS name node and data node plug-ins. This example shows the default ports (15452 and 15453, respectively).

   Note: The SAS Plug-ins for Hadoop installation directory, **`HDATHome`**, is deployed under **`/opt/sas/`** by default. If you have chosen a different installation path, use that different path where necessary in this step and in later steps.

3. Search for the string `hdfs_client_config_safety_valve` in the HDFS configuration filter. Or navigate to the Gateway Default Group.

Under **Advanced**, add the following lines to the HDFS Client Advanced Configuration Snippet (Safety Valve) for the hdfs-site.xml property.

Note: Specify each property on a single line. Multiple lines are used here to improve readability.

```
<property><name>com.sas.cas.hadoop.service.namenode.port</name> <value>15452
     </value></property>
<property><name>com.sas.cas.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>com.sas.cas.hadoop.short.circuit.command</name> <value>
     /opt/sas/HDATHome/bin/sascasfd</value></property>
```

4. Search for the string `hdfs_service_env_safety_valve` in the HDFS configuration filter. Or navigate to the Service-Wide group.

   Under **Advanced**, add the following line to the HDFS Service Environment Advanced Configuration Snippet (Safety Valve) property:

   ```
   HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/sas/HDATHome/lib/*
   ```

5. Search for the string `hdfs_client_env_safety_valve` in the HDFS configuration filter. Or navigate to the Gateway Default Group.

   Under **Advanced**, add the following property in the HDFS Client Environment Advanced Configuration Snippet (Safety Valve) for hadoop-env.sh:

   ```
   HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/sas/HDATHome/lib/*
   ```

6. Click **Cloudera Manager Home**, and then select the Yarn service. Within the Yarn service, search for the string `mapreduce_client_env_safety_valve` in the Yarn configuration filter. Or navigate to the Gateway Default Group by clicking **Configuration and Gateway Default Group** ▶ **Advanced**.

   Add the following properties in Gateway Client Environment Advanced Configuration Snippet (Safety Valve) for hadoop-env.sh:

   ```
   JAVA_HOME=Java-home-path
   HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/sas/HDATHome/lib/*
   ```

   Note: The value of the JAVA_HOME variable should be a valid path to the Java installation that is used by the Cloudera Hadoop system on all Hadoop nodes. For example:

   ```
   JAVA_HOME=/usr/java/jdk1.7.0_67-cloudera
   ```

7. Save changes.

8. From the Cloudera Manager home, select the drop-down list for your cluster and select **Deploy Client Configuration**. In the dialog box, select **Deploy Client Configuration**, and then click **Close**.

9. Restart the HDFS service and any dependencies in Cloudera Manager.

10. Verify that you can access HDFS. For details, see “Verifying Access to HDFS” on page 46.

Hortonworks Data Platform Hadoop

To use Ambari to configure Hortonworks HDFS service properties for SAS Plug-ins for Hadoop perform the following steps:

1 Log on to Ambari.

2 Click **HDFS Service**.

3 Choose **Config Section**.

4 Click **Advanced**.

5 Select **Custom hdfs-site** and add the following properties:

dfs.namenode.plugins
`com.sas.cas.hadoop.NameNodeService`

dfs.datanode.plugins
`com.sas.cas.hadoop.DataNodeService`

com.sas.cas.hadoop.service.namenode.port
`15452`

Note: You can change the port for the SAS name node and data node plug-ins. The following example shows the default ports (15452 and 15453, respectively):

com.sas.cas.hadoop.service.datanode.port
`15453`

dfs.namenode.fs-limits.min-block-size
`0`

com.sas.cas.hadoop.short.circuit.command
`/opt/sas/HDATHome/bin/sascasfd`

Note: The SAS Plug-ins for Hadoop installation directory, `HDATHome`, is deployed under `/opt/sas/` by default. If you have chosen a different installation path, use that different path where necessary in this step and in later steps.

6 Save the properties.

7 Add the following statement to the **hadoop-env template** of HDFS on the **Advanced hadoop-env** tab, in the section, `# Set Hadoop-specific environment variables here`:

```
export HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/sas/HDATHome/lib/*
```

Note: Ensure that the export command is on a single line.

8 Restart all Hortonworks Data Platform (HDP) services and MapReduce services.

9 Verify that you can access HDFS. For details, see "Verifying Access to HDFS" on page 46.

Apache Hadoop

To configure Apache Hadoop HDFS service properties for SAS Plug-ins for Hadoop, perform the following steps:

On every machine in the cluster, in `/etc/hadoop/hadoop-env.sh`, in the section,

```
# Set Hadoop-specific environment variables here
```

set the HADOOP_CLASSPATH to the following value:

```
export
HADOOP_CLASSPATH=$HADOOP_CLASSPATH:/opt/sas/HDATHome/lib/*
```

Note: Ensure that the export command is on a single line.

The following properties have been added to your hdfs-site.xml file:

```
<property>
<name>dfs.namenode.plugins</name>
<value>com.sas.cas.hadoop.NameNodeService</value>
</property>
<property>
<name>dfs.datanode.plugins</name>
<value>com.sas.cas.hadoop.DataNodeService</value>
</property>
<property>
<name>com.sas.cas.hadoop.service.namenode.port</name>
<value>15452</value>
</property>
<property>
<name>com.sas.cas.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>com.sas.cas.hadoop.short.circuit.command</name>
<value>/opt/sas/HDATHome/bin/sascasfd</value>
</property>
```

Verifying Access to HDFS

To test that your connection to Hadoop is working, perform the following steps:

1. In the Folders pane of SAS Visual Analytics:
   a. Expand **SAS Folders**.
   b. Expand **Products**.
   c. Expand **SAS Visual Analytics**.
   d. Expand **Samples**.
   e. Select VA_SAMPLE_BANKCAMPAIGN.
   f. Right-click and select **Add to HDFS**.
   g. In the Add Table dialog box, click **OK**.
2. When the table is loaded to Hadoop you receive the following message:

   ```
   Table was added successfully
   ```

   If you receive a different message, click **Show Details** to troubleshoot the problem.

sashdat-install.sh Reference

Overview and Requirements

The sashdat-install.sh script enables you to deploy SAS Plug-ins for Hadoop. The script provides an alternative to Cloudera parcels and Ambari stacks.

The UNIX account with which the script is run requires sudo privileges and passwordless SSH access to every machine in the Hadoop cluster when adding and removing SAS Plug-ins for Hadoop. No sudo access is required when you are checking whether the plug-ins are correctly installed on all data nodes.

When adding or removing SAS Plug-ins for Hadoop, the sashdat-install.sh script attempts to query the Hadoop configuration to automatically discover the machine name for all of the nodes in the cluster. In order to query the hdfs service for machine names, the script assumes that your site uses the default Hadoop user account ‘hdfs.’ You can provide a different Hadoop account with execution permissions for the `hdfs` command, or you can provide your own list of machine names.

Note: It is recommended that you provide your own list of machine names using the -host on page 48 or -hostfile on page 48 options.

You must provide a list of machine names under the following conditions:

- the hdfs service is down
- you are adding new machines to the cluster
- you want to use a list of machines that is different from what is in the Hadoop configuration

Syntax

- Add SAS Plug-ins for Hadoop:

```
sashdat-install.sh -add <-hostfile host-list-filename | -
host "host-list"> <-hdfsuser user-ID> <-hdathome parent-
installation-path>
```

- Remove SAS Plug-ins for Hadoop:

```
sashdat-install.sh -remove <-hostfile host-list-filename | -
host “host-list”> <-hdfsuser user-ID> <-hdathome parent-
installation-path>
```

- Check whether SAS Plug-ins for Hadoop is properly installed:

```
sashdat-install.sh -version <-hdathome parent-installation-
path>
```

Options

-add

installs SAS Plug-ins for Hadoop on all machines in the cluster, or on the user-supplied list of machines.

Requirements The UNIX user account with which you run the sashdat-install.sh script must have sudo permissions and passwordless SSH access to every machine in the Hadoop cluster.

The script assumes that your site uses the default Hadoop user account, ‘hdfs,’ with which the script automatically retrieves the list of data nodes from the Hadoop configuration. If your site does not use the ‘hdfs’ user account, then you must use the `-hdfsuser user-ID` option to provide a valid Hadoop user account with execution permissions for the `hdfs` command. Or, you can provide your own list of machines using either the `-hostfile` or `-host` option.

-hdathome *parent-installation-path*

(optional) specifies a custom parent installation path for the plug-ins instead of the default **`/opt/sas`** path. The subdirectory **`HDATHome`** will be created under the specified `-hdathome` path.

-hdfsuser *user-ID*

(optional) specifies the user ID that has execution permissions for hdfs to run the `hdfs dfsadmin -report` command to retrieve the machine names of the nodes in the Hadoop cluster.

The `-hdfsuser` option is not required when the default Hadoop account, ‘hdfs,’ is present, or when you supply your own list of machines in the Hadoop cluster using the `-hostfile` or `-host` option.

-hostfile *host-list-filename*

(optional) specifies the full path of the file that contains the list of machine names for all of the cluster nodes on which the plug-ins are installed or removed.

Requirement The host list file must contain one fully qualified machine name per line.

Example

```
machine001.example.com
machine002.example.com
machine003.example.com
machine004.example.com
```

-host "*host-list*"

(optional) specifies the list of machine names for all of the cluster nodes on which the plug-ins are installed or removed.

Requirement If you specify more than one machine, then the names must be separated by spaces or commas.

Examples

```
-host "server1 server2 server3"
```

```
-host "blue1,blue2,blue3"
```

-remove

removes the plug-ins on all machines in the cluster, or on a user-supplied the list of machines.

Requirements The UNIX user account with which you run the sashdat-install.sh script must have sudo permissions and passwordless SSH access to every machine in the Hadoop cluster.

The script assumes that your site uses the default Hadoop user account, hdfs, with which the script automatically retrieves the list of data nodes from the Hadoop configuration. If your site does not use the hdfs user account, then you must use the -`hdfsuser user-ID` option to provide a valid Hadoop user account with execution permissions for the `hdfs` command. Or, you can provide your own list of machines using either the -`hostfile` or -`host` option.

-version <-hdathome *parent-installation-path*>
displays the version of the plug-ins that are installed.

Example
```
./sashdat-install.sh -version
```

-x -check <-hdathome *parent-installation-path*>
checks whether the plug-ins are installed correctly on all data nodes.

Tip You can specify the hosts for which you want to check the plug-ins by using the -`hostfile` or -`host` option.

Example
```
./sashdat-install.sh -x -check
```

Add Examples

This section demonstrates various ways to use sashdat-install.sh to add SAS Plug-ins for Hadoop to your supported Hadoop cluster:

Add using the ‘hdfs’ account to query Hadoop for a list of machines:
```
./sashdat-install.sh -add
```

Add using the ‘my-hdfs’ account to query Hadoop for a list of machines:
```
./sashdat-install.sh -add -hdfsuser my-hdfs
```

Add specifying a user-supplied list of machines:
```
./sashdat-install.sh -add -hostfile /tmp/my_hosts
```

Add specifying a user-supplied installation path:
```
./sashdat-install.sh -add -hdathome /var/my_sasplugins/
```

Add specifying a user-supplied list of machines:
```
./sashdat-install.sh -add -host "host1, host2, host3"

./sashdat-install.sh -add -hostfile /tmp/my_hosts
```

Remove Examples

This section demonstrates various ways to use sashdat-install.sh to remove SAS Plug-ins for Hadoop to your supported Hadoop cluster:

Remove using the ‘hdfs’ account to query Hadoop for a list of machines:
```
./sashdat-install.sh -add
```

Remove using the ‘my-hdfs’ account to query Hadoop for a list of machines:
```
./sashdat-install.sh -remove -hdfsuser my-hdfs
```

Remove specifying a user-supplied list of machines:
```
./sashdat-install.sh -remove -hostfile /tmp/my_hosts
```

Remove specifying a user-supplied installation path:
```
./sashdat-install.sh -remove -hdathome /var/my_sasplugins/
```

Remove specifying a user-supplied list of machines:

```
./sashdat-install.sh -remove -host "host1, host2, host3"

./sashdat-install.sh -remove -hostfile /tmp/my_hosts
```

5

Deploying the SAS High-Performance Analytics Environment

Infrastructure Deployment Process Overview

Installing and configuring the SAS High-Performance Analytics environment is the sixth of eight 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) Modify co-located Hadoop.

▶ **6. Deploy the SAS High-Performance Analytics environment.**

7. (Optional) Deploy the SAS Embedded Process for Hadoop.

8. (Optional) Configure the analytics environment for a remote parallel connection.

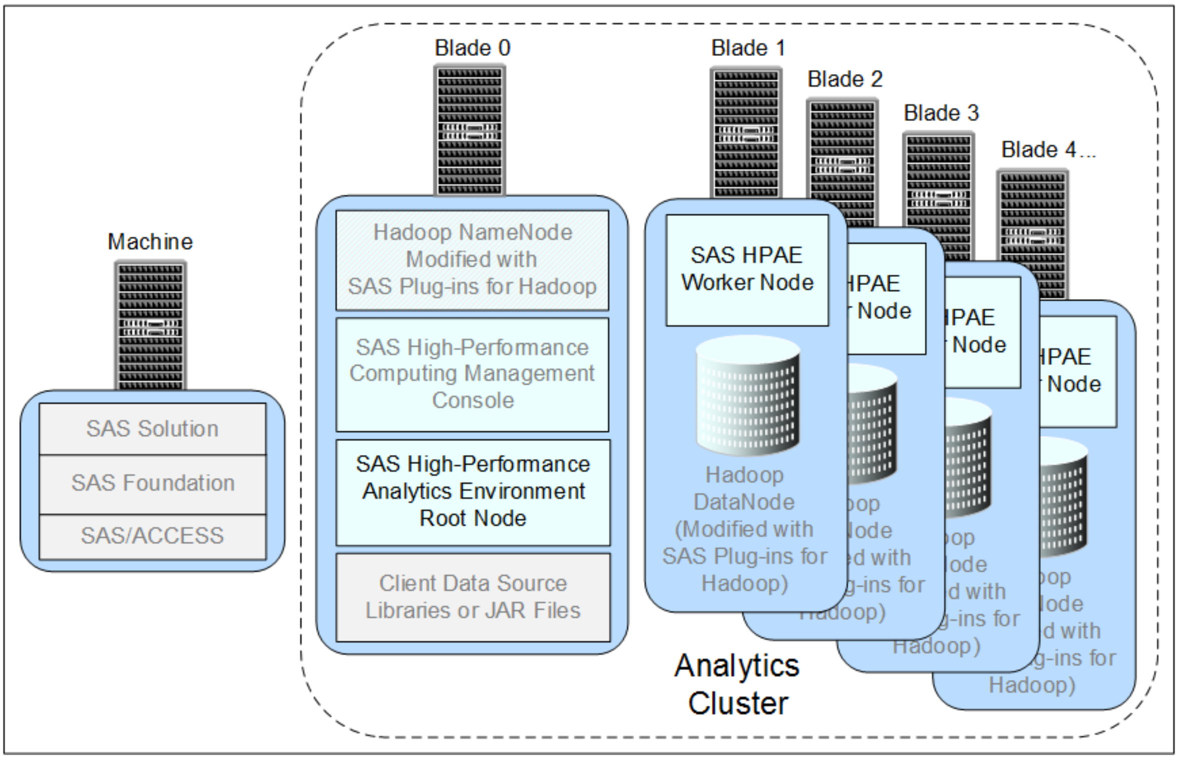
This chapter describes how to install and configure all of the components for the SAS High-Performance Analytics environment on the machines in the cluster.

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.

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

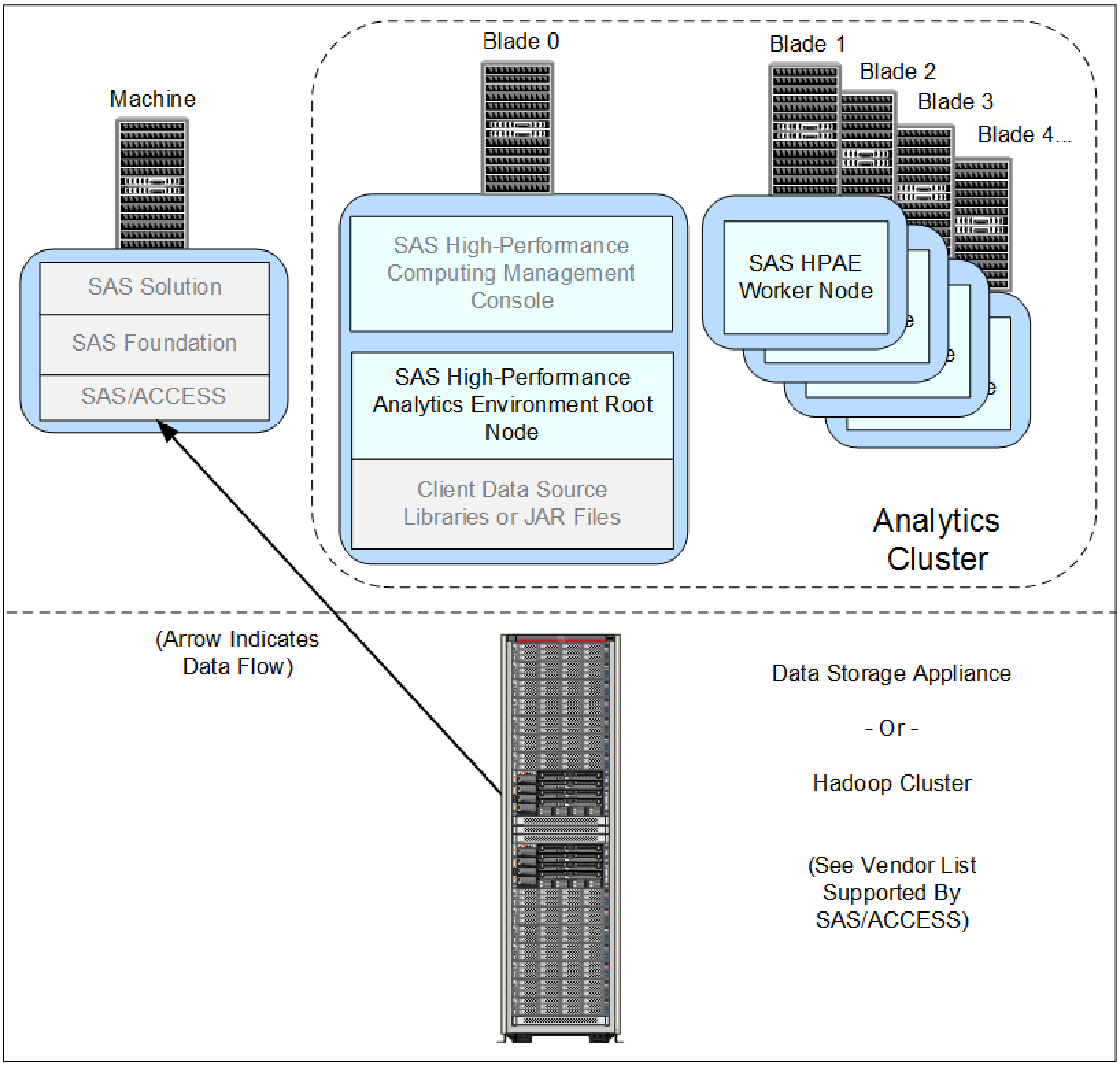
Figure 5.1 Analytics Environment Co-Located on the Hadoop Cluster


Machine
SAS Solution
SAS Foundation
SAS/ACCESS
Blade 0
Hadoop NameNode Modified with SAS Plug-ins for Hadoop
SAS High-Performance Computing Management Console
SAS High-Performance Analytics Environment Root Node
Client Data Source Libraries or JAR Files
Blade 1
Blade 2
Blade 3
Blade 4 ...
SAS HPAE Worker Node
Hadoop DataNode (Modified with SAS Plug-ins for Hadoop)
Analytics Cluster

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:

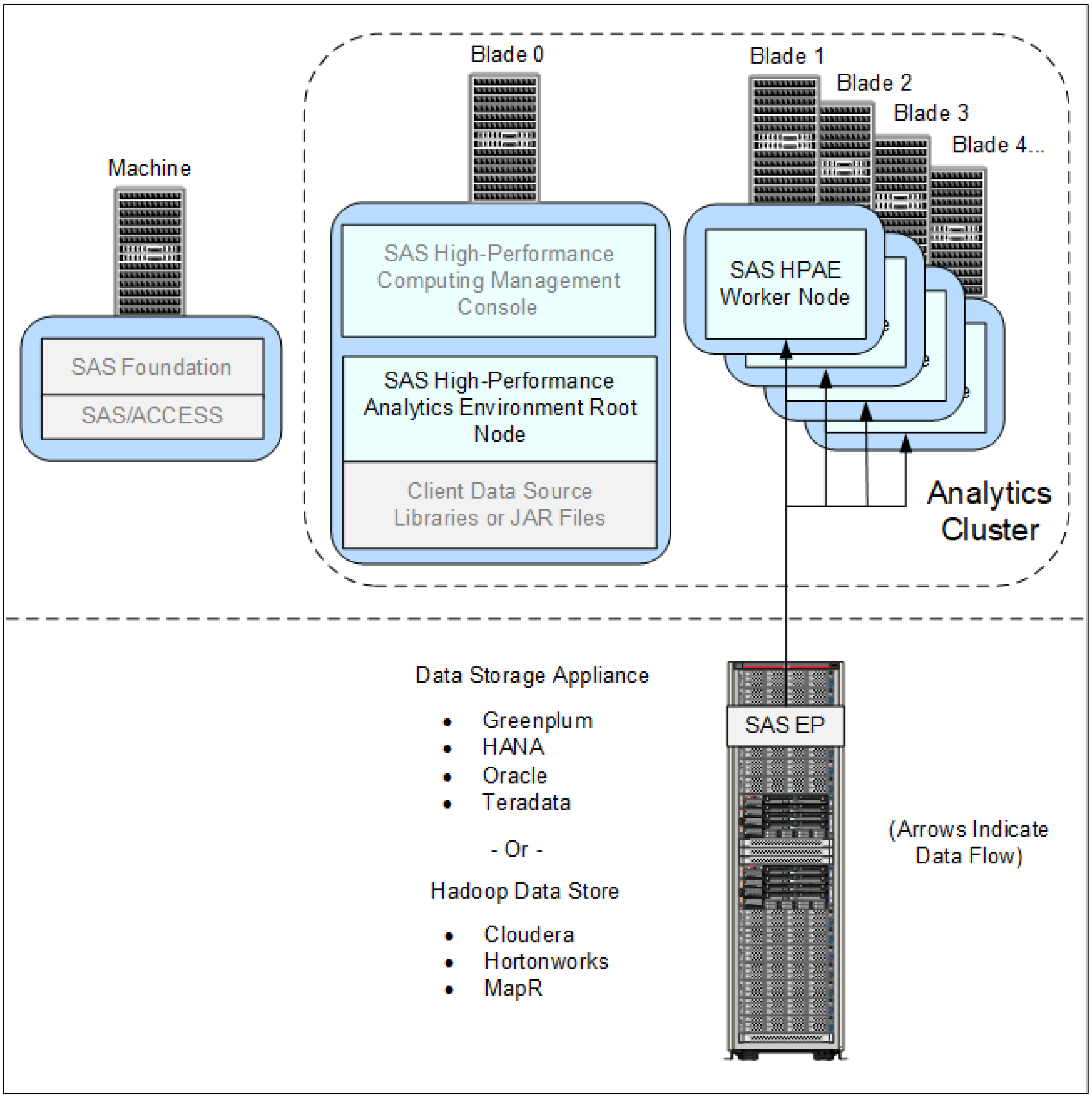
Figure 5.2 *Analytics Environment Remote from Your Data Store (Serial Connection)*


Machine
SAS Solution
SAS Foundation
SAS/ACCESS
Blade 0
SAS High-Performance Computing Management Console
SAS High-Performance Analytics Environment Root Node
Client Data Source Libraries or JAR Files
Blade 1
Blade 2
Blade 3
Blade 4...
SAS HPAE Worker Node
Analytics Cluster
(Arrow Indicates Data Flow)
Data Storage Appliance
- Or -
Hadoop Cluster
(See Vendor List Supported By SAS/ACCESS)

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 5.3 Analytics Environment Remote from Your Data Store (Parallel Connection)



The SAS High-Performance Analytics environment is packaged in separate executables. Refer to the following table for more information:

Table 5.1 Installation and Configuration Packages for the SAS High-Performance Analytics Environment

Order to install	Filename	Purpose
1	TKGrid_Linux_x86_64.sh	Analytics environment installation script for Red Hat Linux 6 and other equivalent, kernel-level Linux systems.

Order to install	Filename	Purpose
2	TKTGDat.sh	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.
3 (optional)	TKGrid_SEC_x86_64.sh	Installation script for enabling the analytics environment to read and write encrypted SASHDAT files.
4 (optional)	TKGrid_REP_x86_64.sh	Script for configuring the SAS High-Performance Analytics environment with a SAS Embedded Process for Red Hat Linux 6 and other equivalent, kernel-level Linux systems.

Encrypting SASHDAT Files

Starting with release 2.94, the SAS High-Performance Analytics environment supports reading and writing files using AES encryption with 256-bit keys. (This feature is very similar to the AES encryption provided by the Base SAS Engine.) SASHDAT encryption is designed to bolster privacy protection for data at rest—that is, data stored in SASHDAT for analytic purposes.

Remember that SASHDAT data is typically not the system of record, but rather a copy of operational data that has been arranged for the purposes of analytics. In addition to encrypting data, many SAS users also anonymize their data when preparing it for analytics.

To enable the SAS High-Performance Analytics environment to read and write SASHDAT using encryption, you must install the TKGrid_SEC package. For more information, see “Configuring the Analytics Environment for SASHDAT Encryption” on page 59.

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 23.

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/3_91/Linux_for_x64`.

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

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 is 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 23.

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 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 5.2 Configuration Parameters for the TKGrid Shell Script

Parameter	Description
TKGrid Configuration Utility. Running on '*machine-name*' Using stdin for options. 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 NFS mount to MAPR directory (ie: /mapr/my.cluster.com, default is none).	If you want the analytics environment to be able to read and write MapR data directly, enter the NFS mount here (for example, `/mapr/my.cluster.com`). The mount point must exist on all nodes, including the name node. The TKGrid script sets the environment variable, TKMPI_MAPRHDFSPREFIX, to point to this share. For more information, see http://doc.mapr.com/display/MapR/Accessing+Data+with+NFS.

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`** **Note:** Enter the above option on one line. Do not add any carriage returns or other whitespace characters. 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.
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 specify *no* umask value, press Enter. Or, specify a umask value and press Enter. For more information, see "Consider Umask Settings" on page 23.

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 5.3 Configuration Prompts for the TKTGDat Shell Script

TKTG Configuration Utility. Running on '*machine-name*' Using stdin for options. 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 If you are planning to use the High-Performance Analytics environment in a locale other than English, you must copy the appropriate locale files from **SASFoundation/9.4/misc/tktg** to the **TKTGDat** directory on every machine in the analytics cluster.

In this example, the simultaneous command, `simcp`, is used to copy the Japanese locale files to the **TKTGDat** directory on each machine in the analytics cluster:

```
/opt/TKGrid/bin/simcp /opt/SASHome/SASFoundation/9.4/misc/tktg/jp* /opt/TKTGDat
```

14 Make one of the following choices:

- To enable the SAS High-Performance Analytics environment to read and write SASHDAT using encryption, proceed to "Configuring the Analytics Environment for SASHDAT Encryption" on page 59.
- To configure the analytics environment for a SAS Embedded Process, proceed to "Configuring for Access to a Data Store with a SAS Embedded Process" on page 73.
- To validate your analytics environment, proceed to "Validating the Analytics Environment Deployment" on page 60.

Configuring the Analytics Environment for SASHDAT Encryption

In release 2.94, the SAS High-Performance Analytics environment supports reading and writing files using AES encryption with 256-bit keys. (This feature is very similar to the AES encryption provided by the SAS BASE Engine.)

Note: For U.S. export purposes, SAS designates each product based on the encryption algorithms and the product's functional capability. The ability to encrypt SASHDAT files is available to most commercial and government users inside and outside the U.S. However, some countries (for example, Russia, China, and France) have import restrictions on products that contain encryption, and the U.S. prohibits the export of encryption software to specific embargoed or restricted destinations.

To enable the SAS High-Performance Analytics environment to read and write SASHDAT using encryption, follow these steps:

1 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_Encryption_Installation/3_91/Linux_for_x64`.

2 Copy TKGrid_SEC_x86_64.sh to the `/tmp` directory of the root node of the cluster.

3 Log on to the machine that is 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 23.

4 Change directories to the desired installation location, such as `/opt`.

5 Run the TKGrid_SEC_x86_64 shell script in this directory.

6 Respond to the prompts from the shell script:

Table 5.4 Configuration Prompts for the TKGrid_SEC_x86_64 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.

7 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.

Note: The contents of TKGrid_SEC must be distributed to every machine in the analytics cluster.

The shell script creates a `lib2` subdirectory and a file named VERSION2.

> **TIP** If you are using Hadoop as your data provider, make sure that you follow the steps described for your distribution of Hadoop in Chapter 4, "Modifying Co-Located Hadoop with SAS Plug-ins for Hadoop," on page 39.

8. To validate your analytics environment, proceed to "Validating the Analytics Environment Deployment" on page 60.

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 60.
- "Use MPI to Validate" on page 61.

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 one of the machines in the analytics cluster.

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 Chapter 6, "Configuring the Analytics Environment for a Remote Parallel Connection," 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 Chapter 6, “Configuring the Analytics Environment for a Remote Parallel Connection,” on page 65.

Resource Management for the Analytics Environment

Resource Settings File

You can specify 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 values in resource.settings are commented. Uncomment the variables and add values that make sense for your site. For more information, see “Using CGroups to Manage CPU” 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 analytics environment to participate in the resource accounting that YARN performs. For more information, see “Managing Resources” in *SAS LASR Analytic Server: Reference Guide*.

resource.settings consists of the following:

```
# VM limit (in KBytes). Default is unlimited
#export TKMPI_ULIMIT="-v 50000000"
```

```
# Location for temporary files.
#export TKOPT_ENV_UTILLOC=/tmp

# Maximum runtime for non-LASR TKGrid jobs.
#export TKMPI_MAXRUNTIME=3600

# UMask for any files created.
#export TKMPI_UMASK=0022

# Nice level for TKGrid jobs. If unset, defaults to 0 for LASR, 5 for non-LASR
#export TKMPI_NICE=5

# Time to wait for MPI to initialize. Default is 30s.
#export TKMPI_INIT_TIMEOUT=30

# Security token for MPICH. No socket authentication is performed if unset.
#export MPICH_SECURITY_TOKEN=$RANDOM$RANDOM

# Command to give load score for a node.
#export TKMPI_SCORENODE=$TKMPI_DIR/bin/scorenode.sh

# Time (in seconds) to wait between receiving start and end of security key.
# Defaults to 10. Valid range (0-1000)
#export MPICH_SECURITY_WAIT=10

# Memory allocation limit (in MBytes). Excludes mmapped files. Default is unlimited.
#export TKMPI_MEMSIZE=30000

# Cgroup to associate with TKGrid jobs.
#export TKMPI_CGROUP="cgexec -g cpu:50"

# Resource Manager.
#export TKMPI_YARN_PRIORITY=2
#export TKMPI_YARN_TIMEOUT=3600
#export TKMPI_YARN_CORES=1
#export TKMPI_RESOURCEMANAGER="java -Xmx256m -Xms256m -cp \"`$HADOOP_HOME/bin/hadoop
classpath`\" com.sas.grid.provider.yarn.tkgrid.JobLauncher --masterMem 2000 --javaMem 500
--hostlist \$TKMPI_YARN_HOSTS --cores \$TKMPI_YARN_CORES --memory \$TKMPI_MEMSIZE
--priority \$TKMPI_YARN_PRIORITY --timeout \$TKMPI_YARN_TIMEOUT --jobname $TKMPI_APPNAME"


# 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

#    if [ "$TKMPI_INFO" = "LASRLOAD" ]; then
#      TKMPI_INFO is an environment variable that will be passed from
#                 MVA SAS to the grid. It can be used to distinguish a
#                 proc lasr create from a proc lasr add, by including
#                 this line before the proc lasr add:
#                 options set=TKMPI_INFO="LASRLOAD";
#       To exclude from YARN resource manager.
#       unset TKMPI_RESOURCEMANAGER
#    fi

# 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
```

Request Memory with TKMPI_INFO

When programmers use TKMPI_INFO in their SAS code, the SAS High-Performance Analytics environment can better decide how much memory to request.

Consider this example: the $TKMPI_APPNAME variable is set to `lasr`for both a SAS Analytic LASR Server (PROC LASR CREATE) and for a SAS Analytic LASR Server Proxy used when loading a table (PROC LASR ADD). This makes it impossible to specify a YARN memory limit differently for these two cases. Most likely, a SAS Analytic LASR Server would want a large amount of memory and the proxy server would require a smaller amount.

Here is an example of how you might use TKMPI_INFO in a SAS program to solve the memory issue:

```
options set=TKMPI_INFO="LASRSTART";
proc lasr create port=17761;
performance nodes=2; run;

options set=TKMPI_INFO="LASRLOAD";
proc lasr add data=sashelp.cars port=17761; run
```

In resource.settings, you might add an entry similar to the following:

```
if [ "$TKMPI_APPNAME" = "lasr" ]; then
  if  [ "$TKMPI_INFO" = "LASRSTART" ];
     export TKMPI_MEMSIZE=60000
  fi
  if  [ "$TKMPI_INFO" = "LASRLOAD" ];
     export TKMPI_MEMSIZE=4000
  fi
fi
```

Note that TKMPI_INFO is not limited to SAS Analytic LASR Server. TKMPI_INFO can also be used for any other HPA PROC. You could use the variable to pass any type of information you need to resource.settings (for example, SMALL, MEDIUM, LARGE classes).

6

Configuring the Analytics Environment for a Remote Parallel Connection

Infrastructure Deployment Process Overview

Configuring your data storage is the last of eight 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) Modify co-located Hadoop.

6. Deploy the SAS High-Performance Analytics environment.

7. (Optional) Deploy the SAS Embedded Process for Hadoop.

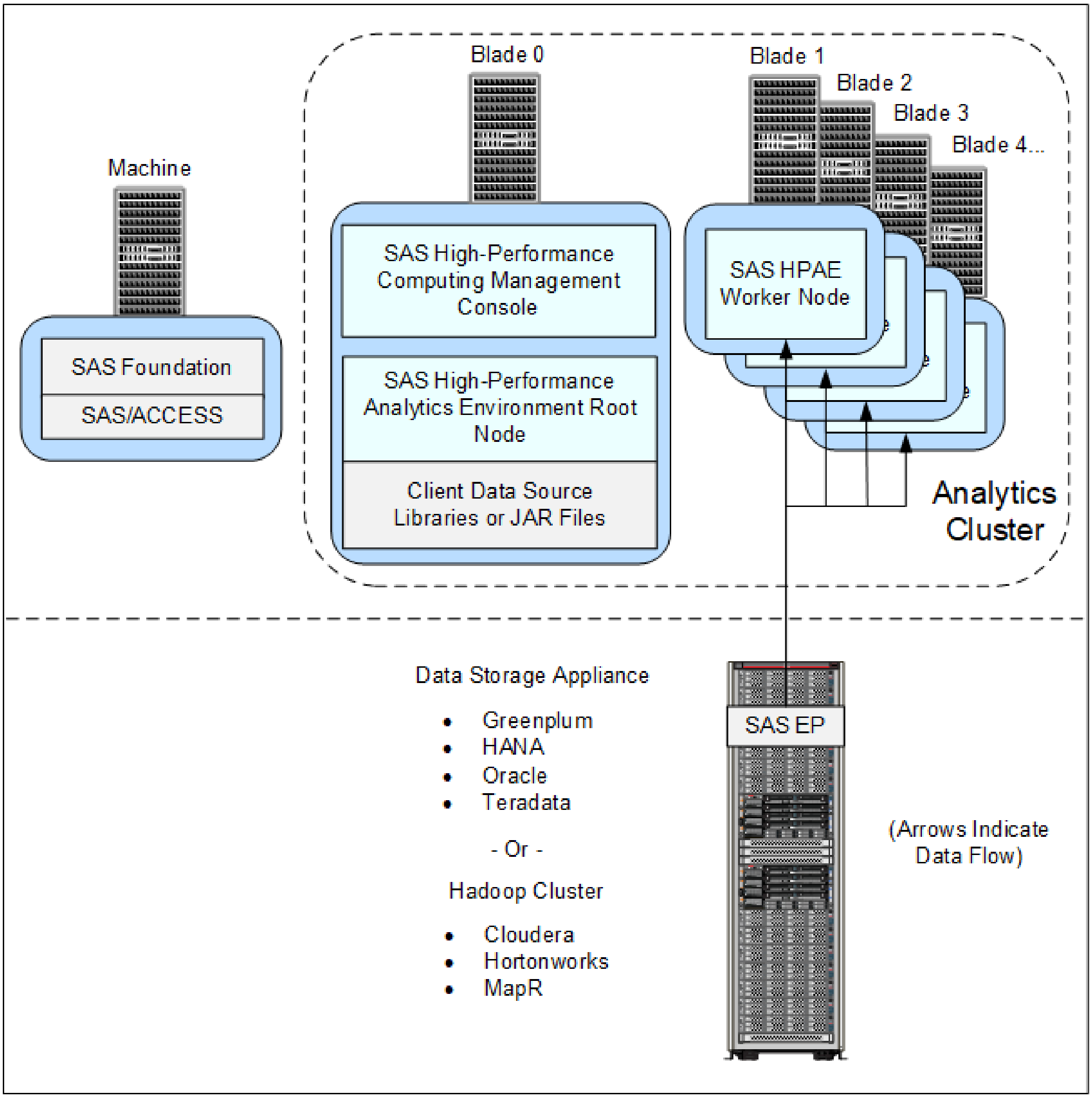
▶ **8. (Optional) Configure the analytics environment for a remote parallel connection.**

Overview of Configuring the Analytics Environment for a Remote Parallel Connection

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. After you have installed SAS/ACCESS and its embedded process, you configure the analytics environment for the particular access interface that you will use with a shell script, TKGrid_REP.

For information about installing the SAS Embedded Process, see the *SAS and SAS Viya Embedded Process: Deployment Guide*.

Figure 6.1 *Analytics Cluster Remote from Your Data Store (Parallel Connection)*



Preparing for a Remote Parallel Connection

Overview of Preparing for a Remote Parallel Connection

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.

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

1 "Prepare for Hadoop" on page 68 .

2 "Prepare for a Greenplum Data Computing Appliance" on page 71.

3 "Prepare for a HANA Cluster" on page 71.

4 "Prepare for an Oracle Exadata Appliance" on page 72.

5 "Prepare for a Teradata Managed Server Cabinet" on page 72.

Prepare for Hadoop

Overview of Preparing for Hadoop

Note: SAS Embedded Process supports the Cloudera, Hortonworks, and MapR distributions of Hadoop. For more specific version information, see the SAS 9.4 Support for Hadoop.

Preparing for a remote parallel connection to Hadoop consists of the following steps:

1 downloading and running a tracer script, hadooptracer.py, supplied by SAS.

hadooptracer.py traces the system calls of various Hadoop client tools and uses this data to identify the appropriate client JAR files that the SAS High-Performance Analytics environment requires for Hadoop client machine to Hadoop server environment connectivity.

The script also copies these Hadoop client JAR files to a location on the Hadoop Hive node that you specify. You then manually copy this directory to the machine where you will deploy the SAS High-Performance environment root node. See "Run the Hadoop Tracer Script" on page 69.

2 recording information about your Hadoop deployment that you will need when configuring the analytics environment for a remote data store.

See "Hadoop Checklists" on page 70.

3 determining the version of the JRE used by Hadoop and installing that same JRE on the analytics cluster.

See "Install the JRE on the Analytics Cluster" on page 71.

Prerequisites for the Hadoop Tracer Script

In order to run the Hadoop tracer script, you must meet these prerequisites:

- Python 2.6 (or later) and strace must be installed.
- We recommend that your Hadoop Hive node machine must have a temporary directory named `tmp` under the root directory (`/tmp`).

  By default, the script uses a single directory (`/tmp`) as its temporary and output directories. However, you can change these using various script options.
- The user running the script must have the following:
  - □ a Linux account with SSH access (password or private key), to the Hive node or NameNode.

- □ authorization to issue HDFS and Hive commands.

  Before the script is executed, a simple validation test is run to see whether HDFS (`hadoop`) and Hive (`hive`) commands can be issued. If the validation test fails, the script does not execute.

- □ a ticket (Kerberos only).

Run the Hadoop Tracer Script

To download and run the Hadoop tracer script, follow these steps:

1. Make sure that your system meets the prerequisites.

   See "Prerequisites for the Hadoop Tracer Script".

2. On a machine from which you can also access your Hadoop Hive node, create a temporary directory to download the script.

   Here is an example:

   ```
   mkdir hadoopfiles_temp
   ```

3. Download the hadooptracer.zip file from the following FTP site to the directory that you created: ftp://ftp.sas.com/techsup/download/blind/access/hadooptracer.zip.

4. If there is not a **`/tmp`** directory on your Hadoop Hive node, create it.

   By default, the script uses a single directory (**`/tmp`**) as its temporary and output directories. However, you can change these using various script options. See Step 8.

5. Using a transfer method, such as PSFTP, SFTP, SCP, or FTP, transfer the ZIP file to the Hive node on your Hadoop cluster.

   Here is an example:

   ```
   scp /opt/sas/hadoopfiles_temp/hadooptracer.zip
   root@hive_node.example.com:/tmp
   ```

6. On the Hadoop Hive node, change to the **`/tmp`** directory and unzip hadooptracer.zip.

   Here is an example:

   ```
   cd /tmp
   unzip hadooptracer.zip
   ```

7. Enter the following command to grant Execute permissions on the script file:

   ```
   chmod 755 ./hadooptracer.py
   ```

8. Run the script with these options:

   ```
   python hadooptracer.py --filterby=latest --postprocess
   ```

   Note: hadooptracer.py ignores --`postprocess` on Cloudera Hadoop clusters.

   > **TIP** For more information about script options, run the script with the -`h` option.

The script does the following:

a traces the system calls of various Hadoop client tools and uses this data to identify the appropriate client JAR files.

b copies the client JAR files to `/tmp/jars`.

c writes its log to `/tmp/hadhooptracer.log`.

Note: Some error messages in the console output for hadooptracer.py are normal and do not necessarily indicate a problem with the JAR and configuration file collection process. However, if the files are not collected as expected or if you experience problems connecting to Hadoop with the collected files, contact SAS Technical Support and include the hadooptracer.log file.

9 When the script finishes executing, delete the following files:

- derby*.jar
- spark-examples*.jar
- ranger-plugins-audit*.jar
- avatica*.jar
- hadoop-0.20.2-dev-core*.jar

10 Copy the JAR files that hadhooptracer.py writes in`/tmp/jars` to a directory on the SAS High-Performance Analytics environment root node machine. As the analytics environment configuration script prompts you for this directory later, be sure to note it in Table 6.1.

Hadoop Checklists

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.

Note: The SAS Embedded Process supports the Cloudera, Hortonworks, and MapR distributions of Hadoop. For more detailed information, see the SAS Foundation system requirements documentation for your operating environment.

1 Record the path to the Hadoop client JAR files required by the analytics environment in the table that follows:

Table 6.1 Record the Location of the Hadoop Client JAR Files

Example	Actual Path of the Required Hadoop JAR Files on Your Analytics Environment Root Node
/opt/hadoop_jars (Hadoop client JAR files)	

Note: The location of the common and core JAR files listed in Table 6.1 should be the same location that you copied the client JAR files to in Step 10 on page 70.

2 Record the location (JAVA_HOME) of the 64-bit Java Runtime Engine (JRE) used by the analytics environment in the table that follows:

Table 6.2 Record the Location of the JRE Used by the SAS High-Performance Analytics Environment

Example	Actual Path of the JRE on Your Analytics Environment Root Node
/opt/java/jre1.7.0_07	

Note: This is the location of the JRE that the SAS High-Performance Analytics environment uses, not the location of the JRE that Hadoop uses.

Install the JRE on the Analytics Cluster

The SAS High-Performance Analytics environment requires a 64-bit Java Runtime Engine (JRE) when the environment is configured for a remote parallel connection.

Note: The JRE used by the analytics environment must match the version of the JRE used by your Hadoop cluster.

As the analytics environment configuration script prompts you for the location of this JRE (JAVA_HOME), be sure to note it in Table 6.2.

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 6.3 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 6.4 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 6.5 Record the Location of the Oracle Client

Example	Actual Path of the Oracle Client on Your System
/usr/local/ora11gr2/product/11.2.0/client_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 6.6 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 6.7 Record the Location of the Teradata Client

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

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 with a SAS Embedded Process consists of the following steps:

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

For more information, see "Preparing for a Remote Parallel Connection" on page 67.

Note: Other third-party data providers besides Hadoop are supported. For more information, see the *SAS and SAS Viya Embedded Process: Deployment Guide*.

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 73.

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 74.

How the Configuration Script Works

You configure the SAS High-Performance Analytics environment with a SAS Embedded Process 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 for a Remote Parallel Connection" on page 67.
2. Make sure that you understand how the analytics environment configuration script works, as described in "How the Configuration Script Works" on page 73.
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/3_8/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 is 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 23.

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 6.8 Configuration Parameters for the TKGrid_REP Shell Script

Parameter	Description
TKGrid Remote EP Addon Configuration Utility. Running on '*machine-name*' Using stdin for options. 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.
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 6.7 on page 73.)
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.
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 6.3 on page 71.)
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.
Enter path of 64 bit JRE i.e.: /usr/java/jdk1.7.0_09/jre	If you chose `yes` in the previous step, specify the path where the JRE resides that analytics environment uses and press Enter. (This path was recorded earlier in Table 6.2 on page 71.)
Enter path of the directory (or directories separated by :) containing the Hadoop client JAR files.	Specify the path where the client Hadoop JAR files required by SAS reside and press Enter. (This path was recorded earlier in Table 6.1 on page 70.)
Enter any JRE Options you need added for the Java invocation, or just Enter if none.	If you need to add any JRE options, do so here (for example, `-Djava.security.auth.login.config=/opt/mapr/conf/mapr.login.conf -Djava.library.path=/opt/mapr/lib`).
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.

Parameter	Description
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 6.5 on page 72.)
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 6.6 on page 72.)
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 6.4 on page 72.)
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 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 56.
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 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 and SAS Viya Embedded Process: Deployment Guide*.

11 To validate your analytics environment, proceed to "Validating the Analytics Environment Deployment" on page 60.

Map Internal Network Names to External Network Names

If your data source is on an internal network and you want to load data using the SAS Embedded Process across a remote parallel connection on an external network, then you must create a grid.publichosts file.

Note: The grid.publichosts file pertains only to data sources that are running Hadoop distributions, not to relational database management systems.

A grid.publichosts file maps your cluster machines' internal network names to their external network names.

To create a grid.publichosts file, follow these steps:

1. Sign in to the SAS High-Performance Analytics environment root node machine.
2. Copy **`TKGrid/grid.hosts`** to **`TKGrid/grid.publichosts`**.
3. Using a text editor, modify grid.publichosts so that it contains two names per line. The machine's internal name should be listed first, followed by the machine's external name.

Here is an example:

```
grid001.example.com ext_grid001.example.com
grid002.example.com ext_grid002.example.com
grid003.example.com ext_grid003.example.com
grid004.example.com ext_grid004.example.com
...
```

4. Although only the root node uses grid.publichosts, you should copy it to all the worker nodes in the cluster for completeness.

Appendix 1

Updating the SAS High-Performance Analytics Infrastructure

Analytics Infrastructure Dependencies

Note the following dependencies when updating the SAS High-Performance Analytics infrastructure:

- If you update the analytics environment, you must also update your co-located Hadoop cluster.
- Update Hadoop first, followed by the analytics environment.

For information about updating SAS Plug-ins for Hadoop, see Chapter 4 on page 39.

Updating SAS Plug-ins for Hadoop

SAS Plug-ins for Hadoop provides files that you add to your pre-existing Hadoop distribution in order to write SASHDAT file blocks evenly across the HDFS file system. For information about updating SAS Plug-ins for Hadoop, see Chapter 4 on page 39.

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 55.

- 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 55.

  When you change the path of the SAS High-Performance Analytics environment, you also have to reconfigure the SAS LASR Analytic Server to point to the new path. For more information, see Add a SAS LASR Analytic Server in the *SAS Visual Analytics: Administration Guide*.

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 If you are also updating your co-located Hadoop deployment, update Hadoop first, and then the SAS High-Performance Analytics environment.

  For information about updating SAS Plug-ins for Hadoop, see Chapter 4 on page 39.

2 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`.

3 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 installation directories on all the machines in the cluster: `/HPA-environment-installation-directory/bin/simsh rm -r -f /HPA-environment-installation-directory`.

4 Re-install the analytics environment using the shell script as described in "Install the Analytics Environment" on page 55.

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 (or later), 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 implementing YARN, transfer these settings to YARN. For more information, see the *SAS LASR Analytic Server: Reference Guide*.

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 81.

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 --prefix=install-directory
      /SAS-Software-Depot-root-directory/standalone_installs/
      SAS_High-Performance_Computing_Management_Console/2_9/Linux_for_x64/
      sashpcmc-2.9.x86_64.rpm
   ```

   In this command, *install-directory* is the location where the management console is installed and *SAS-Software-Depot-root-directory* is the location where your SAS Software Depot resides.

4. Log on to the console to validate your update.

Appendix 2

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. To use the commands, a user account must be configured for passwordless secure shell.

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 the SAS High-Performance Computing Management Console: User's Guide.

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 3

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 4

gridmon.sh Usage and Reference Guide

gridmon.sh Usage

Overview

Note: gridmon.sh is supported only on Linux platforms.

gridmon.sh is a console or terminal application that can be run from a Linux terminal or a terminal emulator such as PuTTY. gridmon.sh displays data streamed from all the machines on your analytics cluster showing information about jobs, individual machines on the cluster and attached disks.

gridmon.sh enables you to perform several limited actions, such as killing a job, killing a rank, or running gstack. (For a complete list of functionality, see “gridmon.sh Reference”.) If an X Server resides on the SAS High-Performance Analytics environment root node, then you can launch an Xterm, a perf top session, or an Attach Debugger session directly from gridmon.sh.

Note: Attach Debugger is for use only when directed by SAS Technical Support or by SAS R&D.

If you run gridmon.sh in record mode, gridmon.sh captures this streamed data. Using the playback feature, you can investigate the state of your analytics cluster at the time it was recorded.

Use gridmon.sh

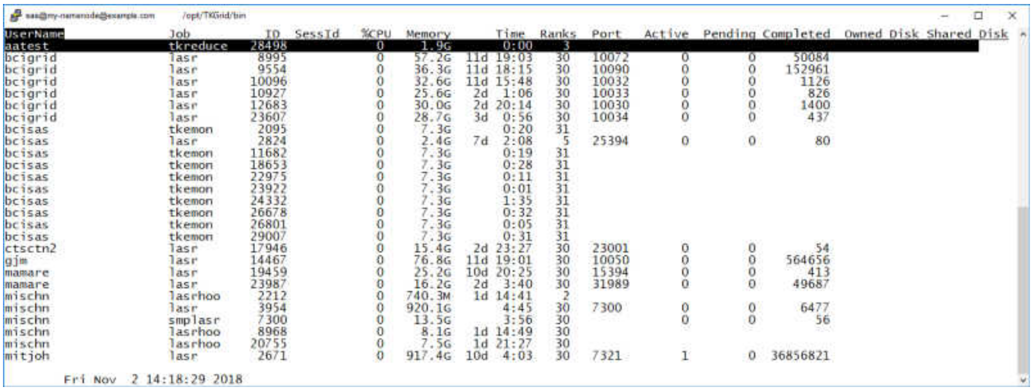
1 Log on to the SAS High-Performance Analytics environment root node machine as a user with passwordless SSH access to all analytics cluster nodes. The user also needs sudo privileges on all nodes on the analytics cluster to run Grid Monitor commands that require root access, such as viewing process limits and killing jobs.

2 To start gridmon.sh, run the following command:

```
/opt/TKGrid/bin/gridmon.sh
```

3 By default, gridmon.sh runs in job mode.

Figure A4.1 gridmon.sh Running in Job Mode



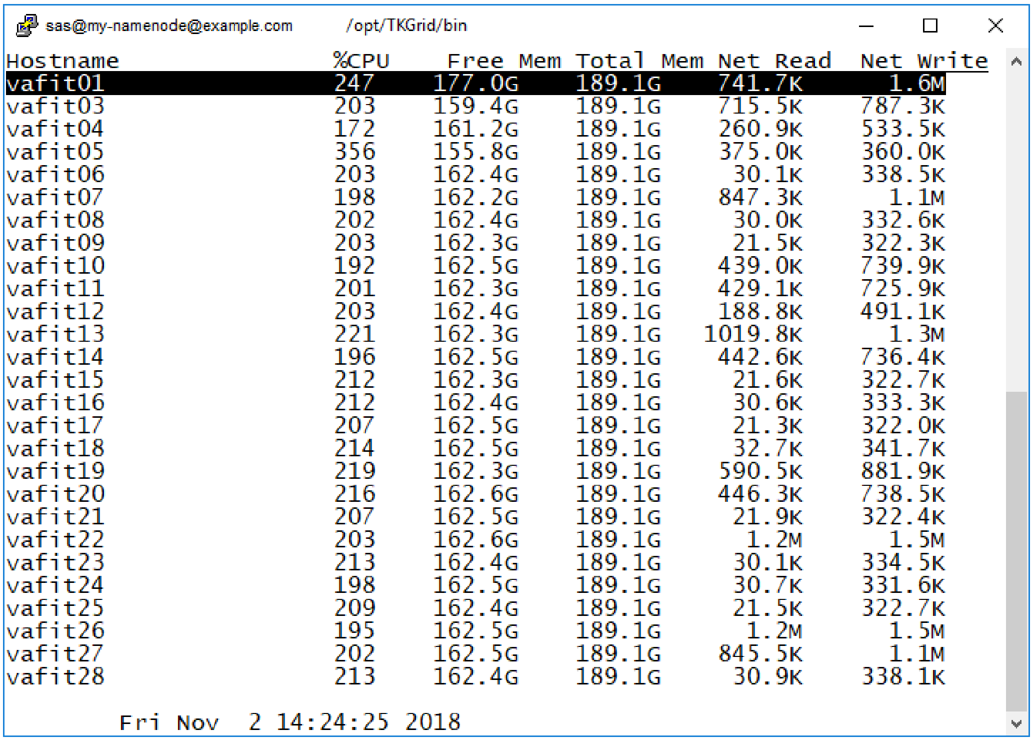
```
UserName            Job            ID  SessId  %CPU   Memory       Time Ranks  Port   Active  Pending Completed  Owned Disk  Shared Disk
aatest              tkreduce    28498             0     1.9G       0:00     3
bcigrid             lasr         8995             0    57.2G  11d 19:03    30  10072       0        0     50084
bcigrid             lasr         9554             0    36.3G  11d 18:15    30  10090       0        0    152961
bcigrid             lasr        10096             0    32.6G  11d 15:48    30  10032       0        0      1126
bcigrid             lasr        10927             0    25.6G   2d  1:06    30  10033       0        0       826
bcigrid             lasr        12683             0    30.0G   2d 20:14    30  10030       0        0      1400
bcigrid             lasr        23607             0    28.7G   3d  0:56    30  10034       0        0       437
bcisas              tkemon       2095             0     7.3G       0:20    31
bcisas              lasr         2824             0     2.4G   7d  2:08     5  25394       0        0        80
bcisas              tkemon      11682             0     7.3G       0:19    31
bcisas              tkemon      18653             0     7.3G       0:28    31
bcisas              tkemon      22975             0     7.3G       0:11    31
bcisas              tkemon      23922             0     7.3G       0:01    31
bcisas              tkemon      24332             0     7.3G       1:35    31
bcisas              tkemon      26678             0     7.3G       0:32    31
bcisas              tkemon      26801             0     7.3G       0:05    31
bcisas              tkemon      29007             0     7.3G       0:31    31
ctsctn2             lasr        17946             0    15.4G   2d 23:27    30  23001       0        0        54
gjm                 lasr        14467             0    76.8G  11d 19:01    30  10050       0        0    564656
mamare              lasr        19459             0    25.2G  10d 20:25    30  15394       0        0       413
mamare              lasr        23987             0    16.2G   2d  3:40    30  31989       0        0     49687
mischn              lasrhoo      2212             0   740.3M   1d 14:41     2
mischn              lasr         3954             0   920.1G       4:45    30  7300        0        0      6477
mischn              smplasr      7300             0    13.5G       3:56    30              0        0        56
mischn              lasrhoo      8968             0     8.1G   1d 14:49    30
mischn              lasrhoo     20755             0     7.5G   1d 21:27    30
mitjoh              lasr         2671             0   917.4G  10d  4:03    30  7321        1        0  36856821

       Fri Nov  2 14:18:29 2018
```

Note: **Owned Disk** and **Shared Disk** do not apply to the SAS High-Performance Analytics environment.

4 To run in machine mode, enter **m**.

Figure A4.2 gridmon.sh Running in Machine Mode

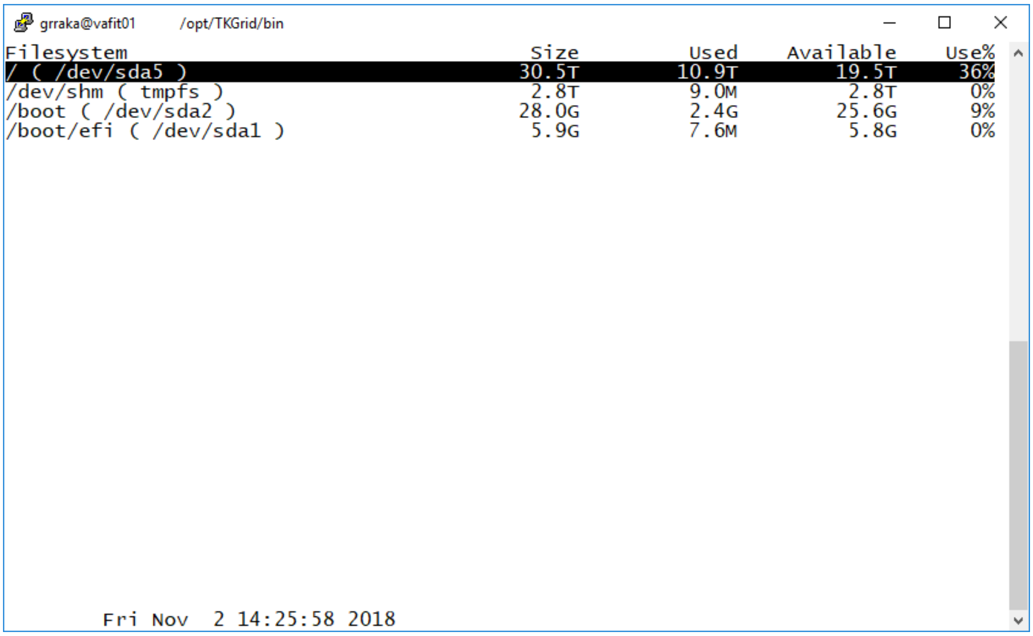


```
Hostname                    %CPU    Free Mem Total Mem Net Read  Net Write
vafit01                     247     177.0G   189.1G    741.7K     1.6M
vafit03                     203     159.4G   189.1G    715.5K   787.3K
vafit04                     172     161.2G   189.1G    260.9K   533.5K
vafit05                     356     155.8G   189.1G    375.0K   360.0K
vafit06                     203     162.4G   189.1G     30.1K   338.5K
vafit07                     198     162.2G   189.1G    847.3K     1.1M
vafit08                     202     162.4G   189.1G     30.0K   332.6K
vafit09                     203     162.3G   189.1G     21.5K   322.3K
vafit10                     192     162.5G   189.1G    439.0K   739.9K
vafit11                     201     162.3G   189.1G    429.1K   725.9K
vafit12                     203     162.4G   189.1G    188.8K   491.1K
vafit13                     221     162.3G   189.1G   1019.8K     1.3M
vafit14                     196     162.5G   189.1G    442.6K   736.4K
vafit15                     212     162.3G   189.1G     21.6K   322.7K
vafit16                     212     162.4G   189.1G     30.6K   333.3K
vafit17                     207     162.5G   189.1G     21.3K   322.0K
vafit18                     214     162.5G   189.1G     32.7K   341.7K
vafit19                     219     162.3G   189.1G    590.5K   881.9K
vafit20                     216     162.6G   189.1G    446.3K   738.5K
vafit21                     207     162.5G   189.1G     21.9K   322.4K
vafit22                     203     162.6G   189.1G      1.2M     1.5M
vafit23                     213     162.4G   189.1G     30.1K   334.5K
vafit24                     198     162.5G   189.1G     30.7K   331.6K
vafit25                     209     162.4G   189.1G     21.5K   322.7K
vafit26                     195     162.5G   189.1G      1.2M     1.5M
vafit27                     202     162.5G   189.1G    845.5K     1.1M
vafit28                     213     162.4G   189.1G     30.9K   338.1K

         Fri Nov  2 14:24:25 2018
```

5 To run in disk mode, enter `d`.

Figure A4.3 gridmon.sh Running in Disk Mode



```
Filesystem                                   Size        Used    Available   Use%
/ ( /dev/sda5 )                             30.5T       10.9T        19.5T    36%
/dev/shm ( tmpfs )                           2.8T        9.0M         2.8T     0%
/boot ( /dev/sda2 )                         28.0G        2.4G        25.6G     9%
/boot/efi ( /dev/sda1 )                      5.9G        7.6M         5.8G     0%

        Fri Nov  2 14:25:58 2018
```

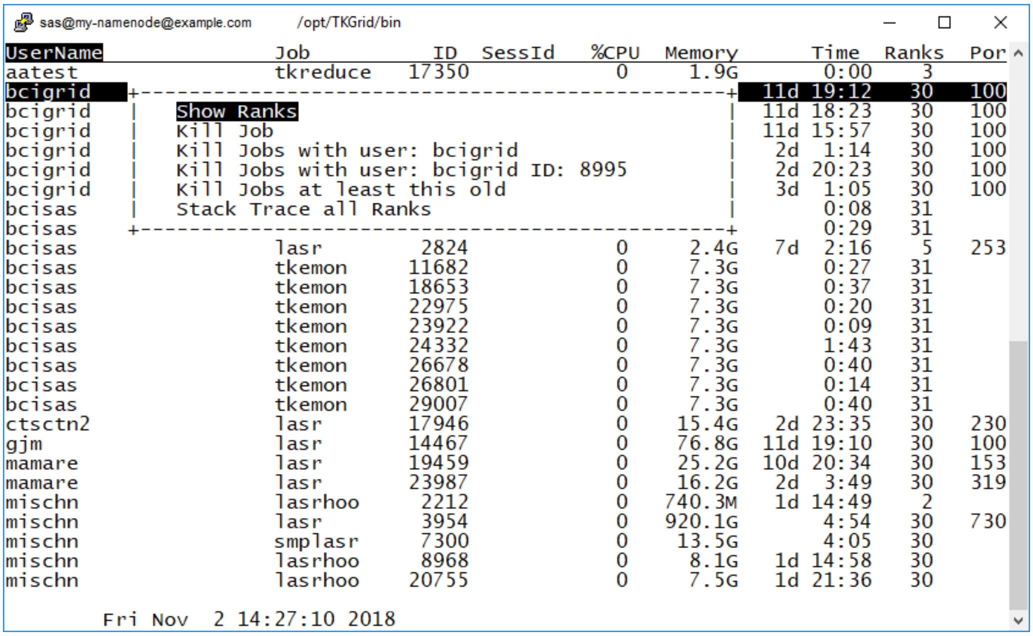
6 Refer to “Overview” in “gridmon.sh Reference” for the commands that you can use in each mode.

7 In job mode there are two menus.

Run in job mode (enter `j`), select a job, and press the Enter key.

The **Show Ranks** menu is displayed:

Figure A4.4 Show Ranks Menu



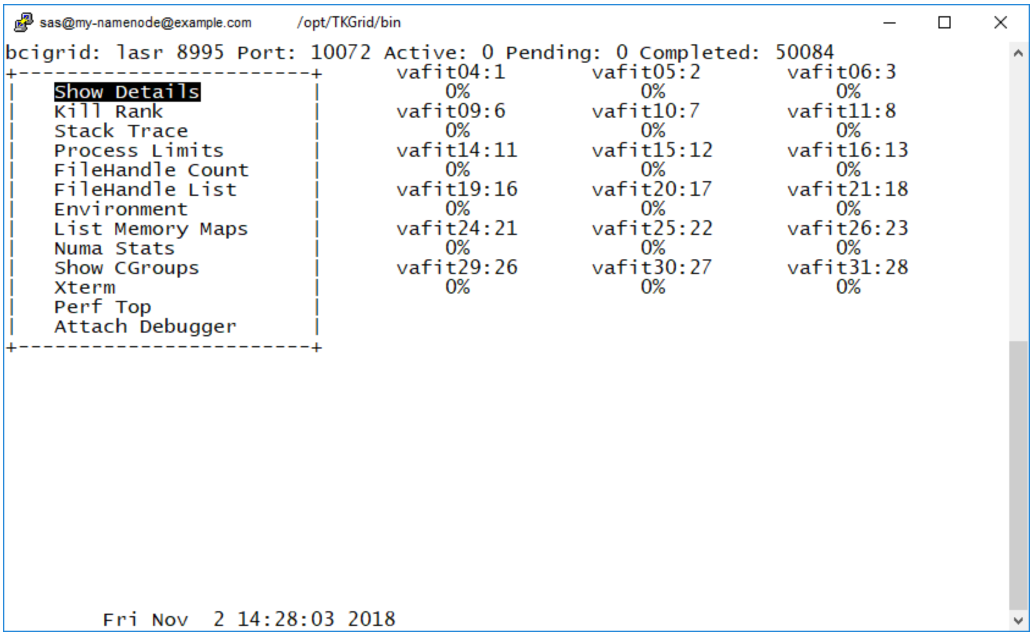
```
UserName              Job          ID  SessId   %CPU  Memory      Time  Ranks  Por
aatest                tkreduce    17350           0     1.9G       0:00    3
bcigrid   +------------------------------------------------+  11d 19:12   30   100
bcigrid   |   Show Ranks                                   |  11d 18:23   30   100
bcigrid   |   Kill Job                                     |  11d 15:57   30   100
bcigrid   |   Kill Jobs with user: bcigrid                 |   2d  1:14   30   100
bcigrid   |   Kill Jobs with user: bcigrid ID: 8995        |   2d 20:23   30   100
bcigrid   |   Kill Jobs at least this old                  |   3d  1:05   30   100
bcisas    |   Stack Trace all Ranks                        |       0:08   31
bcisas    +------------------------------------------------+       0:29   31
bcisas                lasr         2824           0     2.4G   7d  2:16    5   253
bcisas                tkemon      11682           0     7.3G       0:27   31
bcisas                tkemon      18653           0     7.3G       0:37   31
bcisas                tkemon      22975           0     7.3G       0:20   31
bcisas                tkemon      23922           0     7.3G       0:09   31
bcisas                tkemon      24332           0     7.3G       1:43   31
bcisas                tkemon      26678           0     7.3G       0:40   31
bcisas                tkemon      26801           0     7.3G       0:14   31
bcisas                tkemon      29007           0     7.3G       0:40   31
ctsctn2               lasr        17946           0    15.4G   2d 23:35   30   230
gjm                   lasr        14467           0    76.8G  11d 19:10   30   100
mamare                lasr        19459           0    25.2G  10d 20:34   30   153
mamare                lasr        23987           0    16.2G   2d  3:49   30   319
mischn                lasrhoo      2212           0   740.3M   1d 14:49    2
mischn                lasr         3954           0   920.1G       4:54   30   730
mischn                smplasr      7300           0    13.5G       4:05   30
mischn                lasrhoo      8968           0     8.1G   1d 14:58   30
mischn                lasrhoo     20755           0     7.5G   1d 21:36   30

        Fri Nov  2 14:27:10 2018
```

8 For specific information about each **Show Ranks** menu command, see "Show Ranks Menu Commands".

9 From the **Show Ranks** menu, select **Show Ranks** to display the Ranks window. Press **Enter** to display the **Show Details** menu.

Figure A4.5 Show Details Menu



```
bcigrid: lasr 8995 Port: 10072 Active: 0 Pending: 0 Completed: 50084
+------------------------+      vafit04:1       vafit05:2       vafit06:3
|   Show Details         |          0%              0%              0%
|   Kill Rank            |      vafit09:6       vafit10:7       vafit11:8
|   Stack Trace          |          0%              0%              0%
|   Process Limits       |      vafit14:11      vafit15:12      vafit16:13
|   FileHandle Count     |          0%              0%              0%
|   FileHandle List      |      vafit19:16      vafit20:17      vafit21:18
|   Environment          |          0%              0%              0%
|   List Memory Maps     |      vafit24:21      vafit25:22      vafit26:23
|   Numa Stats           |          0%              0%              0%
|   Show CGroups         |      vafit29:26      vafit30:27      vafit31:28
|   Xterm                |          0%              0%              0%
|   Perf Top             |
|   Attach Debugger      |
+------------------------+













        Fri Nov  2 14:28:03 2018
```

For specific information about each **Show Details** menu command, see "Show Details Menu Commands".

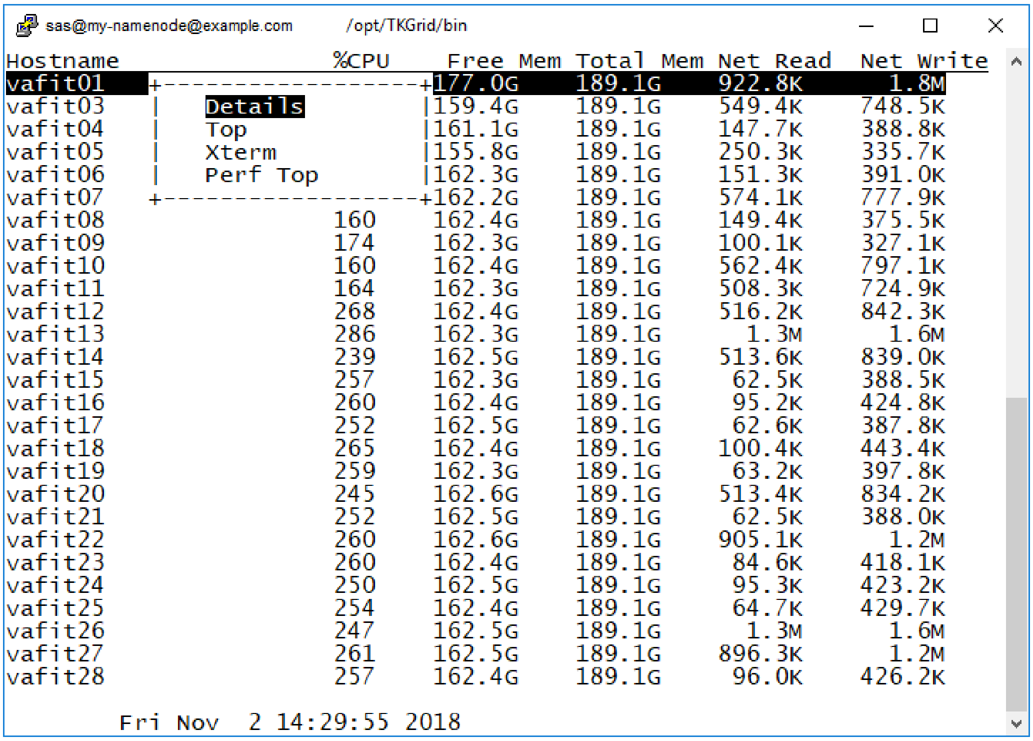
10 Press the Esc key to leave the **Show Details** menu.

11 In machine mode there is one menu.

Run in machine mode (enter `m`), select a machine, and press the Enter key.

The **Details** menu is displayed:

Figure A4.6 Details Menu



sas@my-namenode@example.com /opt/TKGrid/bin

```
Hostname                %CPU   Free Mem Total Mem Net Read  Net Write
vafit01    +------------------+177.0G   189.1G    922.8K      1.8M
vafit03    |   Details        |159.4G   189.1G    549.4K    748.5K
vafit04    |   Top            |161.1G   189.1G    147.7K    388.8K
vafit05    |   Xterm          |155.8G   189.1G    250.3K    335.7K
vafit06    |   Perf Top       |162.3G   189.1G    151.3K    391.0K
vafit07    +------------------+162.2G   189.1G    574.1K    777.9K
vafit08                  160   162.4G   189.1G    149.4K    375.5K
vafit09                  174   162.3G   189.1G    100.1K    327.1K
vafit10                  160   162.4G   189.1G    562.4K    797.1K
vafit11                  164   162.3G   189.1G    508.3K    724.9K
vafit12                  268   162.4G   189.1G    516.2K    842.3K
vafit13                  286   162.3G   189.1G      1.3M      1.6M
vafit14                  239   162.5G   189.1G    513.6K    839.0K
vafit15                  257   162.3G   189.1G     62.5K    388.5K
vafit16                  260   162.4G   189.1G     95.2K    424.8K
vafit17                  252   162.5G   189.1G     62.6K    387.8K
vafit18                  265   162.4G   189.1G    100.4K    443.4K
vafit19                  259   162.3G   189.1G     63.2K    397.8K
vafit20                  245   162.6G   189.1G    513.4K    834.2K
vafit21                  252   162.5G   189.1G     62.5K    388.0K
vafit22                  260   162.6G   189.1G    905.1K      1.2M
vafit23                  260   162.4G   189.1G     84.6K    418.1K
vafit24                  250   162.5G   189.1G     95.3K    423.2K
vafit25                  254   162.4G   189.1G     64.7K    429.7K
vafit26                  247   162.5G   189.1G      1.3M      1.6M
vafit27                  261   162.5G   189.1G    896.3K      1.2M
vafit28                  257   162.4G   189.1G     96.0K    426.2K

        Fri Nov  2 14:29:55 2018
```

For specific information about each **Details** menu command, see “Details Menu Commands”.

12 Enter `q` to exit gridmon.sh.

Run gridmon.sh in Record Mode

You can run gridmon.sh in record mode in order to capture data that is streamed from each machine on your analytics cluster at approximately one second intervals. You can review this captured data later by running gridmon.sh in playback mode.

1 Log on to the SAS High-Performance Analytics environment root node machine as a user with passwordless SSH access to all analytics cluster nodes. The user also needs sudo privileges on all analytics cluster nodes to run Grid Monitor commands that require root access, such as viewing process limits and killing jobs.

2 Change to the following directory:

```
cd /opt/TKGrid/bin/
```

3 To start gridmon.sh in record mode, run the following command:

```
./gridmon.sh -record path/input-filename
```

In this command, ***`path/input-filename`*** is the absolute path and filename for where gridmon.sh writes its output.

Here is an example:

```
./gridmon.sh -record /my_data/tkgridmon_output
```

Run gridmon.sh in Playback Mode

You can run gridmon.sh in playback mode to review data streamed from all the machines on your analytics cluster that you captured earlier while running gridmon.sh in record mode.

1. Log on to the SAS High-Performance Analytics environment root node machine as a user with passwordless SSH access to all analytics cluster nodes. The user also needs sudo privileges on all analytics cluster nodes to run Grid Monitor commands that require root access, such as viewing process limits and killing jobs.

2. Change to the following directory:

   ```
   cd /opt/TKGrid/bin/
   ```

3. To start gridmon.sh in playback mode, run the following command:

   ```
   ./gridmon.sh -playback path/output-filename
   ```

   Here is an example:

   ```
   ./gridmon.sh -gridhost -playback /my_data/tkgridmon_output
   ```

gridmon.sh Reference

Overview

This section describes commands that you can use to operate gridmon.sh. For usage information, see “Use gridmon.sh”.

- “Global Commands”
- “Job Mode Commands”
- “Machine Mode Commands”
- “Disk Mode Menu Commands”
- “Show Ranks Menu Commands”
- “Show Details Menu Commands”
- “Details Menu Commands”

Global Commands

Note: Menu options that produce lengthy results redirect the output to your vi editor. Closing vi returns to gridmon.sh.

Table A4.1 Global Commands

Command	Description
q	Exits gridmon.sh.
Up and down arrows Page Up and Page Down keys	Moves through the list of jobs, machines, or disks.
Backspace key Esc key	Cancels the current menu, prompt, or sub-mode.
?	Shows Help information for gridmon.sh.

For usage information, see "Use gridmon.sh".

Job Mode Commands

Table A4.2 Job Mode Commands

Command	Description
j	Runs gridmon.sh in job mode.
Left and right arrows	Changes the column for sorting the list.
h Home key	Moves to the top of the list.
Enter key	Shows the menu option for the selected job.

For usage information, see "Use gridmon.sh".

Machine Mode Commands

Table A4.3 Machine Mode Commands

Command	Description
m	Runs gridmon.sh in machine mode.
Enter key	Shows menu options for the selected machine.

For usage information, see "Use gridmon.sh".

Disk Mode Menu Commands

Table A4.4 Disk Mode Menu Commands

Command	Description
d	Runs gridmon.sh in disk mode.
Enter key	Shows selected disk use on machines where the disk is present.

For usage information, see "Use gridmon.sh".

Show Ranks Menu Commands

When gridmon.sh is in job mode, you display the **Show Ranks** menu by pressing the Enter key from the main window.

Table A4.5 Show Ranks Menu Commands

Command	Description
Show Ranks	Displays all the ranks belonging to the job and the machines on which they are running.
Kill job	Kills the selected job.
Kill jobs with user: *user-ID*	Kills all jobs of the selected user.
Kill jobs with user: *user-ID* ID: *process-ID*	Kills all jobs of the selected user and specific ID.
Kill jobs at least this old	Kills all jobs at least as old as the selected job.
Stack Trace all Ranks	Runs the gstack application on all processes in this job and collects results. gstack displays its results in your vi editor.

For usage information, see "Use gridmon.sh".

Show Details Menu Commands

When gridmon.sh is in job mode, you display the **Show Details** menu when you press the Enter key from the Ranks window (**Enter** ▶ **Show Ranks**).

Table A4.6 Show Details Menu Commands

Command	Description
Show Details	Shows process ID, CPU use, virtual memory, and if not zero, the following fields: ■ **CGroup Limit**: Size of memory cgroup. ■ **CGroup Usage**: Amount of the CGroup memory that is in use by all processes belonging to this server on the current machine. ■ **Faults/s**: The number of page faults per second for the process, most commonly caused by paging in table data. (Faults can help you determine whether the process is paging.)
Kill Rank	Kills the selected rank or process.
Stack Trace	Runs the gstack application on all processes in this job and collects results. gstack displays its results in your vi editor.
Process Limits	Displays the contents of **`/proc/`*`pid`*`/limits`**.
FileHandle Count	Counts the files owned by the process.
FileHandle List	Lists the files owned by the process.
Environment	Displays the process's environment handles from **`/proc/`*`pid`*`/environ`**.
List Memory Maps	Shows the process's memory maps from **`/proc/`*`pid`*`/maps`**.
Numa Stats	Shows the output from the Linux `numastat` command for this process.
Show CGroups	Shows the Linux cgroups that this process belongs to.
Xterm*	Starts an Xterm on the selected machine.
Perf Top*	Runs the perf top application on this process in a new Xterm window. **Note:** The perf top package must be installed.

Command	Description
Attach Debugger[*]	Attaches a debugger to the running process. Requires a new X window. **Note:** Attach Debugger is for use only when directed by SAS Technical Support or by SAS R&D.

[*] Requires that an X Server be running on the SAS High-Performance Analytics environment root node machine.

For usage information, see "Use gridmon.sh".

Details Menu Commands

When gridmon.sh is in machine mode, you display the **Details** menu by pressing the Enter key from the main window.

Table A4.7 Details Menu Commands

Command	Description
Details	Displays information about the machine, such as CPU utilization, free memory, and total memory.
Top	Runs the top application on all processes in this job and collects results. Top displays its results in your vi editor.
Xterm[*]	Starts an Xterm on the selected machine.
Perf Top[*]	Runs the perf top application on this process in a new Xterm window.

[*] Requires that an X Server be running on the SAS High-Performance Analytics environment root node machine.

For usage information, see "Use gridmon.sh".

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 Red Hat 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 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.

Here 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
```

Here 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.

Here 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
```

Appendix 6

Setting Up Passwordless Secure Shell (SSH)

What Is Passwordless SSH?

Secure Shell (SSH) is a network protocol that allows data to be exchanged using a secure channel between two networked devices. Passwordless SSH enables an identity to connect from one device to another without specifying a password. The identity can log on without a credential challenge, or it can invoke commands on the other device without a credential challenge.

Who Needs Passwordless SSH?

For a non-distributed server, passwordless SSH is not applicable.

For a distributed server, the requirements for passwordless SSH are as follows:

- Each user that needs to start and stop servers and load and unload tables must have an account that is configured for passwordless SSH on each machine in the cluster.
- If you use automated loading, the service account under which the scheduled task runs must be configured for passwordless SSH on each machine in the cluster. This is necessary to perform tasks such as starting and stopping the server and loading and unloading tables.
- For deployments that include SAS Visual Analytics, the service account for SAS LASR Analytic Server Monitor must be configured for passwordless SSH on each machine in the cluster. This is necessary to monitor hardware resources and processes for a distributed SAS LASR Analytic Server. This service account can be the same as the SAS installer account.

How to Set Up Passwordless SSH

You can use a point-and-click interface to generate SSH keys and configure them for passwordless SSH automatically for administrator accounts. See the SAS High-Performance Computing Management Console: User's Guide.

Here are some tips:

- In the SAS High-Performance Computing Management Console, be sure to select the **Generate and Propagate SSH Keys** option on the Create User page. This ensures that passwordless SSH is configured correctly for the account.

- After you add user or group accounts to the machines in the cluster, you must restart the HDFS service if it is co-located. An error message such as the following indicates that a user is not recognized:

  ```
  ERROR: host02.example.com (192.168.1.240) User does not belong to  .
  ```

Generate SSH Keys Manually

The recommended method is to use the SAS High-Performance Computing Management Console to generate SSH keys (as described in the preceding topic).

If you must generate SSH keys manually (for example, for existing user IDs), use the following steps:

1 Generate a private and public key pair on a Linux system. Enter the following command to generate the keys without requiring a passphrase:

```
ssh-keygen -t rsa -P ""
```

2 After the keys are generated, if passwordless SSH is required, then add the public key to the list of authorized keys by entering this command on the command line:

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

3 Check permissions on the **.ssh** directory and the files in your **.ssh** directory. The directory must be readable and writable by you only. The id_rsa file must be readable by you only. To verify access, enter the following command, and check the results:

```
ls -asl ~/.ssh

   4 drwx------ 2 datamgr datamgr 4096 Jan 23 10:27 . a
   4 drwx------ 4 datamgr datamgr 4096 Jan 12 19:09 ..
   4 -rw-r--r-- 1 datamgr datamgr  397 Jan 23 10:27 authorized_keys
   4 -rw------- 1 datamgr datamgr  1675 Jan 23 10:00 id_rsa b
   4 -rw-r--r-- 1 datamgr datamgr  397 Jan 13 10:00 id_rsa.pub
   4 -rw-r--r-- 1 datamgr datamgr 1705 Jan 23 10:27 known_hosts
```

1 The directory permissions for the **.ssh** directory indicate that access is denied for all users other than the directory owner.

2 The id_rsa file is the private key. Read access and Write access are available to the file owner only.

Note: If the machines in the cluster are not configured to access the home directories for the users, create local home directories for the users. Copy the **.ssh** directory for each user to his or her local home directory. Make sure that the permissions are preserved.

About Passwordless SSH and Windows Clients

If you need to access a distributed SAS LASR Analytic Server from a Windows client, then you need to perform the following steps to copy your SSH keys to the Windows machine:

1 Determine your Windows home directory. Enter the following command in a command window:

```
echo %HOMEDRIVE%%HOMEPATH%
```

The results are typically something like **C:\Users\sasdemo**.

2 You can use Windows Explorer to drag-and-drop the **.ssh** directory from your UNIX home directory, or you can use a command like the following to copy it:

```
xcopy driverLetter:\.ssh\* "%HOMEDRIVE%%HOMEPATH%\.ssh" /s /i
```

These steps are typically necessary for deployments that use SAS Studio on a Windows client or SAS solutions that use Windows machines for the server tier.

Troubleshooting

If access problems occur, use the following steps to help diagnose any SSH configuration errors:

1 Impersonate the user or ask the user to perform the following command that requires passwordless SSH:

```
/opt/TKGrid/bin/simsh hostname
```

If each of the machines in the cluster responds with a host name, then no passwordless SSH configuration error exists.

2 As root, log on to one of the machines in the cluster and monitor the logon access:

```
tail -f /var/log/secure
```

3 Review the messages in the **/var/log/secure** file. The following example shows that the file system access permissions for **/home/sas** are not set correctly:

```
Mar 14 22:12:36 hostname sshd[11235]: pam_unix(sshd:session): session opened
for user root by (uid=0)
Mar 14 22:12:57 hostname sshd[11266]: Authentication refused: bad ownership or
modes for directory /home/sas
```

Recommended Reading

Here is the recommended reading list for this title:

- Configuration Guide for SAS Foundation for Microsoft Windows for x64.
- Configuration Guide for SAS Foundation for UNIX Environments.
- *SAS/ACCESS for Relational Databases: Reference*.
- SAS Deployment Wizard and SAS Deployment Manager: User's Guide.
- *SAS Guide to Software Updates and Product Changes*.
- SAS High-Performance Computing Management Console: User's Guide.
- *SAS and Hadoop Technology: Deployment Scenarios*.
- *SAS and Hadoop Technology: Overview*.
- *SAS and SAS Viya Embedded Process: Deployment Guide*.
- *SAS Intelligence Platform: Installation and Configuration Guide*.
- *SAS Intelligence Platform: Security Administration Guide*.
- *SAS LASR Analytic Server: Reference Guide*.
- SAS 9.4 Support for Hadoop.

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

SAS Books
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-0025
Fax: 1-919-677-4444
Email: sasbook@sas.com
Web address: support.sas.com/en/books.html

Glossary

browser

See web browser.

co-located data provider

a distributed data source, such as SAS Visual Analytics Hadoop or a third-party vendor database, that has SAS High-Performance Analytics software installed on the same machines. The SAS software on each machine processes the data that is local to the machine or that the data source makes available as the result of a query.

data set

See SAS data set.

data warehouse (warehouse)

a collection of pre-categorized data that is extracted from one or more sources for the purpose of query, reporting, and analysis. Data warehouses are generally used for storing large amounts of data that originates in other corporate applications or that is extracted from external data sources.

deployment plan

information about what software should be installed and configured on each machine in a SAS deployment. A deployment plan is stored in a plan.xml file.

encryption

the conversion of data by the use of algorithms or other means into an unintelligible form in order to secure data (for example, passwords) in transmission and in storage.

Extensible Markup Language (XML)

a markup language that structures information by tagging it for content, meaning, or use. Structured information contains both content (for example, words or numbers) and an indication of what role the content plays. For example, content in a section heading has a different meaning from content in a database table.

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 (HDFS)

a portable, scalable framework, written in Java, for managing large files as blocks of equal size. The files are replicated across multiple host machines in a Hadoop cluster in order to provide fault tolerance.

HDFS

See Hadoop Distributed File System.

high-performance root node

See root node.

identity

See metadata identity.

Integrated Windows authentication (IWA)

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 (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.

IPv6

See Internet Protocol Version 6.

IWA

See Integrated Windows authentication.

JAR (Java Archive)

the name of a package file format that is typically used to aggregate many Java class files and associated metadata and resources (text, images, etc.) into one file to distribute application software or libraries on the Java platform.

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 Archive

See JAR.

Java Database Connectivity (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.

Java Development Kit (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.

JDBC

See Java Database Connectivity.

JDK

See Java Development Kit.

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 (MPI)

a standardized and portable message-passing system that was designed to function on a wide variety of parallel computers. SAS Analytics applications implement MPI for use in high-performance computing environments.

metadata identity (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 (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.

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 (high-performance root node)

in a SAS High-Performance Analytics application, 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 (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 (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, 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 (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 server that provides access to SAS Foundation features such as the SAS programming language and SAS libraries.

SASHDAT file format

a SAS proprietary data format that is optimized for high performance and computing efficiency. For distributed servers, SASHDAT files are read in parallel. When used with the Hadoop Distributed File System (HDFS), the file takes advantage of data replication for fault-tolerant data access.

SASHOME directory

the location in a file system 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 that is installed on the same computer.

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 (SAS installation data file)

a control file containing license information that is required in order to install SAS.

single sign-on (SSO)

an authentication model that enables users to access a variety of computing resources without being repeatedly prompted for their user IDs and passwords. For example, single sign-on can enable a user to access SAS servers that run on different platforms without interactively providing the user's ID and password for each platform. Single sign-on can also enable someone who is using one application to launch other applications based on the authentication that was performed when the user initially logged on.

SOE

See software order email.

software order email (SOE)

an email message, sent to a customer site, that announces arrival of the software and describes the order. It explains the initial installation steps and might also contain instructions for using Electronic Software Delivery (ESD), if applicable.

spawner

See object spawner.

SSO

See single sign-on.

trusted user

a privileged service account that can act on behalf of other users on a connection to the metadata server.

unrestricted identity

a user or group that has all capabilities and permissions in the metadata environment due to membership in the META: Unrestricted Users Role (or listing in the adminUsers.txt file with a preceding asterisk).

update mode

an operating state of the SAS Deployment Wizard in which users are required to install software updates before they can perform any other deployment tasks. The SAS Deployment Wizard automatically goes into update mode when it determines that the current SAS order contains new versions or maintenance updates to the deployed products in a given SAS installation directory.

warehouse

See data warehouse.

web application

an application that is accessed via a web browser over a network such as the Internet or an intranet. SAS web applications are Java Enterprise Edition (JEE) applications that are delivered via web application archive (WAR) files. The applications can depend on Java and non-Java web technologies.

web authentication

a configuration in which users of web applications and web services are verified at the web perimeter, and the metadata server trusts that verification.

web browser (browser)

a software application that is used to view web content, and also to download or upload information. The browser submits URL (Uniform Resource Locator) requests to a web server and then translates the HTML code into a visual display.

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.

XML

See Extensible Markup Language.

Index