## Contents

What’s New in SAS Embedded Process: Deployment Guide .................................. ix

### PART 1  Getting Started  1

**Chapter 1 / Introduction to SAS Embedded Process** .................................................. 3
- What Is SAS Embedded Process? ............................................................. 3
- Where Do I Start? ................................................................................. 3
- Intended Audience ............................................................................... 4

### PART 2  SAS Embedded Process Deployment for Aster  5

**Chapter 2 / Deploying SAS Embedded Process for Aster** ........................................ 7
- Aster Prerequisites .............................................................................. 7
- Overview of SAS Embedded Process for Aster ..................................... 8
- Aster Deployment Steps ....................................................................... 8
- Deploying SAS Embedded Process for Aster ....................................... 9
- Validating the Publishing of the SAS_SCORE( ) and the SAS_PUT( ) Functions........... 13
- Aster Permissions .................................................................................. 13
- Where to Go from Here ....................................................................... 14

**Chapter 3 / Upgrading or Uninstalling SAS Embedded Process for Aster** .................... 15
- Upgrade from a Previous Aster Version – SAS 9.4M6 and Later .................... 15
- Upgrade from a Previous Aster Version – Prior to SAS 9.4M6 ...................... 16
- Uninstall SAS Embedded Process for Aster .......................................... 16
- Aster SAS Embedded Process Versions ............................................... 17

### PART 3  SAS Embedded Process Deployment for DB2  19

**Chapter 4 / Deploying SAS Embedded Process for DB2** ........................................... 21
- DB2 Prerequisites ............................................................................... 21
- Overview of SAS Embedded Process for DB2 ....................................... 22
- DB2 Deployment Steps ......................................................................... 22
- Deploying SAS Embedded Process for DB2 ......................................... 23
- DB2SET Command Syntax for In-Database Processing ......................... 27
- Controlling the SAS Embedded Process for DB2 .................................. 29
- DB2 Permissions .................................................................................. 30
- Where to Go from Here ....................................................................... 33
## Changing the Trace Level

106

## Adjusting the SAS Embedded Process Performance

107

## Specifying the Amount of Memory That SAS Embedded Process Uses

109

## Additional Configuration for All Hadoop Distributions – SAS 9.4

109

## Additional Configuration for MapR Version 6.0

110

## Additional Configuration for Amazon Elastic MapReduce (EMR)

110

## Configuring Temporary Folders When Writing Data to a Hive Table

113

## Additional Configuration for Hortonworks 3.1 with Patch HDP 3.1.0.29-2

113

## Additional Configuration for the SAS Embedded Process for Spark Continuous Session

114

### Chapter 13 / Upgrading or Uninstalling SAS Embedded Process for Hadoop

115

- Upgrade the SAS Embedded Process Parcel from a Previous Version
- Upgrade the SAS Embedded Process Stack from a Previous Version
- Upgrade the Manually Installed SAS Embedded Process from a Previous Version
- Hadoop SAS Embedded Process Versions

### Chapter 14 / Where to Go from Here

129

- Additional Configuration for Other Software

---

### PART 6 SAS Embedded Process Deployment for Netezza

131

### Chapter 15 / Deploying SAS Embedded Process for Netezza

133

- Netezza Prerequisites
- Overview of SAS Embedded Process for Netezza
- Netezza Deployment Steps
- Netezza Deployment Process for Netezza
- Netezza Permissions
- Where to Go from Here

### Chapter 16 / Upgrading or Uninstalling SAS Embedded Process for Netezza

143

- Upgrade from a Previous Netezza Version – SAS 9.4M6 and Later
- Upgrade from a Previous Netezza Version – Prior to SAS 9.4M6
- Uninstall SAS Embedded Process for Netezza
- Netezza SAS Embedded Process Versions

---

### PART 7 SAS Embedded Process Deployment for Oracle

147

### Chapter 17 / Deploying SAS Embedded Process for Oracle

149

- Oracle Prerequisites
- Overview of SAS Embedded Process for Oracle
- Oracle Deployment Steps
- Deploying SAS Embedded Process for Oracle
- Creating Users and Objects for the SAS Embedded Process
- Oracle Permissions
- Where to Go from Here
Chapter 18 / Upgrading or Uninstalling SAS Embedded Process for Oracle

Chapter 19 / Deploying SAS Embedded Process for SAP HANA

Chapter 20 / Upgrading or Uninstalling SAS Embedded Process for SAP HANA

Chapter 21 / Deploying SAS Embedded Process for Teradata

Chapter 22 / Upgrading or Uninstalling SAS Embedded Process for Teradata
## PART 10 Appendixes

### Appendix 1 / SAS Mirror Manager for Linux
- Create a Mirror Repository – Linux .......................................................... 205
- Using a Proxy ......................................................................................... 207
- Specify a Distribution-Specific Subset of Files ................................. 208
- Specify a Log Location ....................................................................... 208
- Options for the mirrormgr mirror Command .................................. 208

### Appendix 2 / SAS Mirror Manager for Windows ................................. 211
- Create a Mirror Repository – Windows ............................................... 211
- Using a Proxy ......................................................................................... 213
- Specify a Log Location ....................................................................... 213
- Options for the mirrormgr mirror Command .................................. 214

### Appendix 3 / SAS Embedded Process Environment Variables ............... 217
What’s New in SAS Embedded Process: Deployment Guide

Overview

This document supports the deployment of SAS Embedded Process for SAS Viya and SAS 9.4.

In SAS Viya 3.5, the following new features and enhancements were added to expand the capabilities of SAS Embedded Process:

- New configuration tasks that are specific to the Hadoop environment were added.
- The SAS Embedded Process for Hadoop deployment script, sasep-admin.sh, was updated with new arguments.

SAS Embedded Process for Hadoop

In SAS Viya 3.5, these changes were made to the deployment process for SAS Embedded Process for Hadoop:

- New configuration tasks specific to the Hadoop environment are added. For more information, see Chapter 12, “Additional Configuration for SAS Embedded Process,” on page 103.
- Several new arguments were added to the deployment script for SAS Embedded Process for Hadoop. New arguments are -license, -prep, and -linklatest. The -sshidf option to support passwordless systems was added to the -prep, -linklatest, and -sync arguments. For more information, see “The sasep-admin.sh Script” on page 93.
PART 1

Getting Started

Chapter 1

Introduction to SAS Embedded Process
Introduction to SAS Embedded Process

What Is SAS Embedded Process?

SAS Embedded Process is a SAS server process that is deployed in and runs within your data source to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that is deployed on your data source. SAS Embedded Process is delivered with any SAS 9.4 software or SAS Viya software that requires it.

Only one version of SAS Embedded Process is required for SAS 9.4, SAS Viya, or both. Only one version should be active and you should always use the latest version that is available.

Where Do I Start?

Note: Starting with SAS 9.4M6, the delivery, installation, and deployment process for SAS Embedded Process has changed. The SAS Embedded Process deployment instructions that were originally contained in two separate documents – SAS In-Database Products: Administrator’s Guide for SAS 9.4 users and SAS Viya for Linux: Deployment Guide for SAS Viya customers – have been combined into this single guide.

SAS Embedded Process is now being delivered via a link in your Software Order Email (SOE), and the deployment process is identical for both SAS 9.4 and SAS Viya.

When you license a SAS product that requires SAS Embedded Process, you will receive two SOEs:
One SOE is for your SAS 9.4 or SAS Viya software offering.

Another SOE is for SAS Embedded Process.

---

Note: SAS Embedded Process is delivered from a SAS Viya repository regardless of whether your software order includes SAS 9.4 or SAS Viya. Prior to this release, SAS Embedded Process for SAS 9.4 software was delivered from the SAS 9.4 Software Depot. For SAS Viya, SAS Embedded Process was delivered from a SAS Viya repository.

---

This deployment guide is organized by data source. Start your deployment by going to the section that is appropriate for your data source. If additional configuration is required, you will be directed to other documentation.

---

**Intended Audience**

This deployment guide is intended for those who need to deploy SAS Embedded Process on their data source. This guide serves anyone who licenses SAS 9.4 software, SAS Viya software, or both, that requires SAS Embedded Process.
PART 2

SAS Embedded Process Deployment for Aster

Chapter 2

*Deploying SAS Embedded Process for Aster* .......................... 7

Chapter 3

*Upgrading or Uninstalling SAS Embedded Process for Aster* .......... 15
Deploying SAS Embedded Process for Aster

Aster Prerequisites

The SAS Embedded Process deployment for Aster requires the following:

- A specific version of the Aster client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.
- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.
- Approximately 300 MB of disk space is required in the /home/beehive/partner file system on each Aster node.
- Connection to the internet and, specifically, SAS repositories online are required in order to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
Overview of SAS Embedded Process for Aster

SAS Embedded Process for Aster is available with SAS Scoring Accelerator and must be deployed before you can use the in-database processing features of the software. For information about using this software, see *SAS In-Database Products: User’s Guide*.

SAS Embedded Process is a SAS server process that runs within Aster to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that is installed on your Aster system.

**Note:** To use your software, some additional configuration might be required. For more information, see “Where to Go from Here” on page 14.

**Note:** If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.

Aster Deployment Steps

To deploy SAS Embedded Process for Aster, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or re-installing) a previous release in SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 3, “Upgrading or Uninstalling SAS Embedded Process for Aster,” on page 15</td>
</tr>
<tr>
<td>3</td>
<td>Validate that the SAS_SCORE( ) and the SAS_PUT( ) functions were published successfully.</td>
<td>“Validating the Publishing of the SAS_SCORE( ) and the SAS_PUT( ) Functions” on page 13</td>
</tr>
<tr>
<td>4</td>
<td>Review the permissions that are needed for in-database processing.</td>
<td>“Aster Permissions” on page 13</td>
</tr>
<tr>
<td>5</td>
<td>If you license SAS Scoring Accelerator for Aster, perform additional configuration as required.</td>
<td>“Where to Go from Here” on page 14</td>
</tr>
</tbody>
</table>
Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all the SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - Linux — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

   **TIP** For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you download it from Internet Explorer. For Windows, the file has a .zip extension.

   Here is an example of a Linux command that is used to unpack SAS Mirror Manager:

   ```bash
   tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz
   ```

3. (Optional) Run the following commands to provide platform and repository information that can be helpful if you need to specify the --platform or --repo options in the mirrormgr mirror command in the next step.

   ```bash
   ./mirrormgr list remote platforms --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   
   ./mirrormgr list remote repos --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   ```

4. Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

   ```bash
   ./mirrormgr mirror --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip <options>
   ```
Note: SAS Mirror Manager downloads all entitled repositories to the `/user-home/sas_repos` (Linux) or `C:\Users\user-id\sas_repos` (Windows) directory by default. You can change the location by using the `--path path-to-mirror-repository` option in the `mirrormgr mirror` command. You can also choose to download only one platform by using the `--platform platform-name`.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.

```
./mirrormgr mirror --deployment-data
    path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
    --platform x64-suse-linux-12 --latest
```

Note: SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

---

**Install SAS Embedded Process RPM Package File for Aster**

To install SAS Embedded Process RPM package file, follow these steps:

1. Locate SAS Embedded Process RPM package file, `sas-sepcoreastr-version-date.timestamp<.rpm-os-designation>.x86_64.rpm`.

   The following table describes the file name options:

   **Table 2.1 RPM File Name Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>version</code></td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “Aster SAS Embedded Process Versions” on page 17.</td>
</tr>
<tr>
<td><code>date.timestamp</code></td>
<td>The date and time at which the RPM package was created.</td>
</tr>
<tr>
<td><code>rpm-os-designation</code></td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For Aster, the permissible value is <code>suse</code>.</td>
</tr>
<tr>
<td></td>
<td>There is no <code>rpm-os-designation</code> value for Red Hat Enterprise Linux.</td>
</tr>
</tbody>
</table>

The RPM file is located in one of the following directories depending on your operating system:
The following table describes the path options:

Table 2.2 Path Options for the RPM File

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of &quot;Create a Mirror Repository&quot; on page 9. Here is an example: /mysas/sas_repos/</td>
</tr>
<tr>
<td>repo-os-designation&lt;option-if-specified&gt;</td>
<td>repo-os-designation identifies the operating system. Available values depend on your data source and software order. Possible values are as follows: x64_suse_linux_12-yum x64_redhat_linux_6-yum If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in option-if-specified. For example, if you specified --latest --platform x64-redhat-linux-6 in your mirrormgr mirror command, the directory would be as follows: x64_redhat_linux_6-yum-latest.</td>
</tr>
</tbody>
</table>

2 Log on to the queen node.
   ssh -l root name-or-ip-of-queen-node

3 Navigate to the parent of the partner directory.
   cd /home/beehive/

4 Create a partner directory if it does not already exist.
   mkdir partner

5 Navigate to the partner directory.
   cd partner

6 Copy the sas-sepcoreastr-version-date.timestamp<rpm-os-designation>.x86_64.rpm file to a location on the Aster server. Make sure that you copy the file to the server machine according to the procedures that are used at your site.

   Here is an example of a secure copy command:
   scp sas-sepcoreastr-16.0.0-20180821.18223344532.suse.x86_64.rpm root@astrmach1:/home/beehive/partner
Note: The /home/beehive/partner directory must be in a location where it can be accessed from at least one of the Aster nodes.

7 Install the RPM file.

```
rpm -ivh sas-sepcoreastr-version-date.timestamp
   <.rpm-os-designation>.x86_64.rpm
```

After the RPM file is installed, the /home/beehive/partner/sas/ep/home/ directory is created in which the SAS Embedded Process files are installed.

8 Run the following code to manually copy the files across all nodes on the beehive by using secure copy and SSH:

```
efile=sas-sepcoreastr-version-date.timestamp
   <.rpm-os-designation>.x86_64.rpm
location=/home/beehive/partner
for ip in `cat /home/beehive/cluster-management/hosts | \
   grep node | awk '{print $3}'`; \
   do \
   echo $ip; \
   scp $location/$epfile root@$ip":$location"; \
   ssh $ip "rpm -ivh $location/$epfile"; \
   done
```

9 Navigate to the directory where SAS Embedded Process is installed.

```
cd /home/beehive/partner/sas/ep/home/sasexe
```

10 Install the SAS_SCORE( ), SAS_PUT( ), and other SQL/MR functions.

a Start the ACT tool.

```
/home/beehive/clients/act -U db_superuser -w db_superuser-password -d database-to-install-sas_score-into
```

b Run this command to verify whether any existing SQL/MR functions are installed:

```
dlF *SAS*
```

If existing SAS functions are installed, it is recommended that you remove them before installing the new ones. Enter these commands to remove the functions:

```
\remove sas_score.tk.so
\remove sas_put.tk.so
\remove sas_row.tk.so
\remove sas_partition.tk.so
```

c Enter the following commands to install the new SQL/MR functions. The SQL/MR functions must be installed under the PUBLIC schema.

```
\install sas_score.tk.so
\install sas_put.tk.so
\install sas_row.tk.so
\install sas_partition.tk.so
```

11 Exit the ACT tool.

```
\q
```
12 Verify the existence and current date of the tkast-runInCluster and tkeastmr.so files. These two binary files are needed by the SAS SQL/MR functions.

```bash
for ip in \
  'hostname -i; cat /home/beehive/cluster-management/hosts | grep node | awk '{print $3}'; \
  do \
    echo $ip; \
    ssh $ip "ls -al /home/beehive/partner/sas/ep/home/sasexe/tkeastmr.so"; \
    ssh $ip "ls -al /home/beehive/partner/sas/ep/home/utilities/bin/\n    tkast-runInCluster";
  done
```

Validating the Publishing of the SAS_SCORE( ) and the SAS_PUT( ) Functions

To validate that the SAS.Score( ) and the SAS.Put( ) functions were published with SAS Embedded Process, run the `\dF` command in the Aster Client or use any of the following views:

- `nc_all_sqlmr_funcs`, where **all** returns all functions on the system
- `nc_user_sqlmr_funcs`, where **user** returns all functions that are owned by or granted to the user
- `nc_user_owned_sqlmr_funcs`, where **user-owned** returns all functions that are owned by the user

Aster Permissions

The person who installs the SAS Embedded Process for Aster needs root privileges for the queen node. This permission is most likely, but not necessarily, needed by the Aster system administrator.

For Aster 6.10 or later, the following schema permissions are needed by the person who runs the scoring and format publishing macros, because all functions and files can be published to a specific schema.

- **USAGE** permission
  - `GRANT USAGE ON SCHEMA your-schema-name TO your-user-id;`

- **INSTALL FILE** permission
  - `GRANT INSTALL FILE ON SCHEMA your-schema-name TO your-user-id;`

- **CREATE** permission
  - `GRANT CREATE ON SCHEMA your-schema-name TO your-user-id;`

- **EXECUTE** permission
  - `GRANT EXECUTE ON FUNCTION PUBLIC.SAS_SCORE TO your-user-id;`
GRANT EXECUTE ON FUNCTION PUBLIC.SAS_PUT TO your-user-id;
GRANT EXECUTE ON FUNCTION PUBLIC.SAS_ROW TO your-user-id;
GRANT EXECUTE ON FUNCTION PUBLIC.SAS_PARTITION TO your-user-id;

Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
</table>
Upgrading or Uninstalling SAS Embedded Process for Aster

Upgrade from a Previous Aster Version – SAS 9.4M6 and Later .................. 15
Upgrade from a Previous Aster Version – Prior to SAS 9.4M6 .................. 16
Uninstall SAS Embedded Process for Aster ........................................ 16
Aster SAS Embedded Process Versions ............................................. 17

Upgrade from a Previous Aster Version – SAS 9.4M6 and Later

To find out when your version of SAS Embedded Process was released, see “Aster SAS Embedded Process Versions” on page 17.

To upgrade from a version of SAS Embedded Process SAS 9.4M6 or a later release, follow these steps:

1 Log on to the queen node.
   ssh -l root name-or-ip-of-queen-node

2 Navigate to the partner directory.
   cd /home/beehive/partner

3 If a sas directory exists in the partner directory, enter these commands to remove the existing installation from the queen node and all the workers.
   for ip in `hostname -i; cat /home/beehive/cluster-management/hosts | grep node | awk '{print $3}'`; do 
     echo $ip; 
     ssh $ip "rpm -e sas-sepcoreastr";
   done

4 Continue with the installation steps in “Aster Deployment Steps” on page 8.
Upgrade from a Previous Aster Version – Prior to SAS 9.4M6

To find out when your version of SAS Embedded Process was released, see “Aster SAS Embedded Process Versions” on page 17.

To upgrade from a version of SAS Embedded Process prior to SAS 9.4M6, follow these steps:

1. Log on to the queen node.
   ```bash
   ssh -l root name-or-ip-of-queen-node
   ```

2. Navigate to the partner directory.
   ```bash
   cd /home/beehive/partner
   ```

3. If a SAS or SASEPHome directory exists in the partner directory, enter these commands to remove the existing installation from the queen node and all the workers.
   ```bash
   location=/home/beehive/partner/
   hostname -i;
   rm -rf $location/SAS;
   rm -rf $location/SASEPHome;
   for ip in `cat /home/beehive/cluster-management/hosts | grep node | awk '{print $3}'`; \
   do \
     echo $ip; \
     ssh $ip "rm -r $location/SAS"; \
     ssh $ip "rm -r $location/SASEPHome"; \
   done
   ```

4. Continue with the installation steps in “Aster Deployment Steps” on page 8.

Uninstall SAS Embedded Process for Aster

The method that you use to uninstall SAS Embedded Process for Aster is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the last step, which is about continuing the installation.

- “Upgrade from a Previous Aster Version – SAS 9.4M6 and Later” on page 15
- “Upgrade from a Previous Aster Version – Prior to SAS 9.4M6” on page 16

To find out when your version of SAS Embedded Process was released, see “Aster SAS Embedded Process Versions” on page 17.
Aster SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Starting with SAS 9.4M6, SAS 9.4 customers can get the latest version of SAS Embedded Process at any time.

<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcoreastr-16.0.0-&lt;date&gt;.timestamp.x86_64.rpm</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoreastr-16.0.0-&lt;date&gt;.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcoreastr-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcoreastr-12.00000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
PART 3

SAS Embedded Process Deployment for DB2

Chapter 4
Deploying SAS Embedded Process for DB2 ........................................... 21

Chapter 5
Upgrading or Uninstalling SAS Embedded Process for DB2 .............. 35
Deploying SAS Embedded Process for DB2

DB2 Prerequisites

The SAS Embedded Process deployment for DB2 requires the following:

- A specific version of the DB2 client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.
- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.
- RPM version 4.9.1.3 or later on AIX systems per IBM’s recommendation. You can check which version of RPM you are using with this command:

  lslpp -l rpm.rte

- Approximately 300 MB of disk space is required in the /opt file system on each DB2 node.
- Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
## Overview of SAS Embedded Process for DB2

SAS Embedded Process for DB2 is available with SAS Scoring Accelerator and must be deployed before you can use the in-database processing features of the software.

For more information about using this software, see “SAS Scoring Accelerator for DB2 under UNIX” in SAS In-Database Products: User's Guide.

SAS Embedded Process is a SAS server process that runs within DB2 to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that is installed on your DB2 system.

**Note:** To use your software, additional configuration is required. For more information, see “Where to Go from Here” on page 33.

**Note:** If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.

## DB2 Deployment Steps

To deploy SAS Embedded Process for DB2, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or re-installing) a previous release of SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 5, “Upgrading or Uninstalling SAS Embedded Process for DB2,” on page 35</td>
</tr>
<tr>
<td>3</td>
<td>Review the permissions that are needed for in-database processing</td>
<td>“DB2 Permissions” on page 30</td>
</tr>
<tr>
<td>4</td>
<td>If you license SAS Scoring Accelerator for DB2, perform additional configuration as required.</td>
<td>“Where to Go from Here” on page 33</td>
</tr>
</tbody>
</table>
Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - Linux — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

   **TIP** For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you downloaded it from Internet Explorer. For Windows, the file has a .zip extension.

   Here is an example of a Linux command to unpack SAS Mirror Manager:
   ```
   tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz
   ```

3. (Optional) Run the following commands to provide platform and repository information that can be helpful if you need to specify the `--platform` or `--repo` options in the `mirrormgr mirror` command in the next step.

   ```
   ./mirrormgr list remote platforms --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   
   ./mirrormgr list remote repos --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   ```

4. Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

   ```
   ./mirrormgr mirror --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   <options>
   ```
Note: SAS Mirror Manager downloads all entitled repositories to the /user-home/sas_repos (Linux) or C:\Users\user-id\sas_repos (Windows) directory by default. You can change the location by using the --path path-to-mirror-repository option in the mirrormgr mirror command. You can also choose to download only one platform by using the --platform platform-name.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.

./mirrormgr mirror --deployment-data path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip --platform x64-suse-linux-12 --latest

Note: SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

---

Install the SAS Embedded Process RPM Package File for DB2

To install the SAS Embedded Process RPM package file, follow these steps:

Note: Ubuntu package files have a DEB extension. For the sake of readability, in this document the file extension RPM is used to denote both RPM and DEB files.

1 Locate the SAS Embedded Process RPM package files, sas-sepcoredb2platform-version-date.timestamp<.rpm-os-designation>.x86_64.rpm. There are two RPM files, AIX and Linux. Both files are in the same directory.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>platform</td>
<td>Is either “a” (AIX) or “l” (Linux).</td>
</tr>
<tr>
<td>version</td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “DB2 SAS Embedded Process Versions” on page 38.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time at which the RPM package was created.</td>
</tr>
</tbody>
</table>
Option | Description
---|---
`rpm-os-designation` | The operating system for which the RPM files were created. For DB2, the permissible values are either `suse` (for SUSE) or `amd64` (for Ubuntu). There is no `rpm-os-designation` value for Red Hat Enterprise Linux.

The two RPM files are located in one of the following directories depending on your operating system:

- **Linux** — `/path-to-mirror-repository/repos/shipped/sepdb21/100.0/sepdb21-100.0.0-repo-os-designation<--option-if-specified>/Packages/s/`
- **Windows** — `C:\path-to-mirror-repository\repos\shipped\sepdb21\100.0\sepdb21-100.0.0-repo-os-designation<--option-if-specified>\Packages\s`

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>path-to-mirror-repository</code></td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of &quot;Create a Mirror Repository&quot; on page 135. Here is an example: <code>/mysas/sas_repos/</code></td>
</tr>
</tbody>
</table>
| `repo-os-designation<--option-if-specified>` | `repo-os-designation` identifies the operating system. Available values depend on your data source and software order. Possible values are as follows:  
  - `x64_suse_linux_12-yum`  
  - `x64_redhat_linux_6-yum`  
  - `x64_ubuntu_linux_14-apt`  
  If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in `option-if-specified`. For example, if you specified `--latest --platform x64-redhat-linux-6` in your `mirrormgr` mirror command, the directory would be as follows: `x64_redhat_linux_6-yum/latest`. |

2. Copy the `sas-sepcoredb2platform-version-date.timestamp<.rpm-os-designation>.x86_64.rpm` to a location on the DB2 server.

The RPM file does not have to be downloaded to a specific location.
Make sure that you copy the file to the server machine according to the procedures that are used at your site.

Here is an example of a secure copy command for the AIX RPM file:

```
scp sas-sepcoredb2a-16.0.0-20180821.18223344532.suse.x86_64.rpm
   root@db2mach1:/RPMTempDir
```

Note: The $RPMTempDir$ directory must be in a location where it can be accessed from at least one of the DB2 nodes.

3 Stop SAS Embedded Process if it is running.
   a Log on as the user who owns the DB2 instance from a secured shell such as SSH.
   b Enter the following command to find out whether SAS Embedded Process is running.

```
ps -ef | grep db2sas
```

   c Enter the following command to determine whether there are any users connected to the instance.

```
db2 list applications
```

d Stop SAS Embedded Process using the db2ida command.

```
db2ida -provider sas -stop
```

If SAS Embedded Process is still running, an error occurs. Enter this command to force SAS Embedded Process to stop.

```
db2ida -provider sas -stopforce
```

4 Log on as a user who has root privileges from a secured shell such as SSH.

5 Navigate to the directory where the RPM file was transferred in Step 2.

```
cd /RPMTempDir
```

6 Install the RPM file.
   a Here is the command for DB2 AIX:

```
rpm -ivh --ignoreos --ignorearch sas-sepcoredb2a-version-
date.timestamp<.rpm-os-designation>.x86_64.rpm
```

   b Here is the command for DB2 Linux:

```
rpm -ivh sas-sepcoredb2l-version-
date.timestamp<.rpm-os-designation>.x86_64.rpm
```

After the RPM file is installed, a /opt/sas/ep/home/ directory is created in which the SAS Embedded Process files are installed. The target directories should be similar to the following, depending on your operating system. Part of the directory path is highlighted to emphasize the different target directories that are used.

```
/opt/sas/ep/home/bin
/opt/sas/ep/home/misc
/opt/sas/ep/home/sasexe
/opt/sas/ep/home/utilities
```

he
Use the db2set command to enable SAS Embedded Process in DB2 and to inform SAS Embedded Process about the location of the SAS Embedded Process library files.

\[
\text{db2set DB2_SAS_SETTINGS="ENABLE_SAS_EP:true;}
\text{LIBRARY_PATH:/opt/sas/ep/home/sasexe"}
\]

The DB2 instance owner must run this command in order for it to be successful. Note that this is similar to setting a UNIX system environment variable using the UNIX EXPORT or SETENV commands. The db2set command registers the environment variable within DB2 only for the default database instance.

For more information about all the arguments that can be used with the db2set command for SAS Embedded Process, see "DB2SET Command Syntax for In-Database Processing" on page 27.

To verify that SAS Embedded Process is set appropriately, run the db2set command without any parameters.

\[
\text{db2set}
\]

If the path was set correctly, it should be similar to the following:

\[
\text{DB2_SAS_SETTINGS=ENABLE_SAS_EP:true;}
\text{LIBRARY_PATH:/opt/sas/ep/home/sasexe}
\]

Start SAS Embedded Process:

\[
\text{db2ida -provider sas -start}
\]

Verify that SAS Embedded Process started.

\[
\text{ps -ef | grep db2sas}
\]

For example, if SAS Embedded Process was started on Linux, lines that are similar to the following are displayed:

\[
\text{db2v10 23265382 20840668 0 Oct 06 - 4:03 db2sasep}
\text{db2v10 27983990 16646196 1 08:24:09 pts/10 0:00 grep db2sasep}
\]

If SAS Embedded Process was started on AIX, lines that are similar to the following are displayed:

\[
\text{db2v10 16449656 15597696 0 14:08:59 - 0:00 db2sase}
\text{db2v10 32702602 12583050 0 14:09:00 - 0:00 db2sase}
\]

In the DB2 instance, you can also verify whether the SAS Embedded Process log file was created in the DB2 instance’s diagnostic directory.

\[
\text{cd instance-home/sql1lib/db2dump}
\text{ls -al sasep0.log}
\]
Arguments

**ENABLE_SAS_EP:**
- Specifies whether the SAS Embedded Process is started with the DB2 instance.
  - **Default:** FALSE

**LIBRARY_PATH:**
- Specifies the path from which the SAS Embedded Process library is loaded.
  - **Requirement:** The path must be fully qualified.

**COMM_BUFFER_SZ:**
- Specifies the size in 4K pages of the shared memory buffer that is used for communication sessions between DB2 and SAS.
  - **Default:** ASLHEAPSZ dbm configuration value
  - **Range:** 1–32767
  - **Requirement:** Size must be an integer value.

**COMM_TIMEOUT:**
- Specifies a value in seconds that DB2 uses to determine whether the SAS Embedded Process is non-responsive when DB2 and SAS are exchanging control messages.
  - **Default:** 600 seconds
  - **Note:** If the time-out value is exceeded, DB2 forces the SAS Embedded Process to stop in order for it to be re-spawned.

**RESTART_RETRIES:**
- Specifies the number of times that DB2 attempts to re-spawn the SAS Embedded Process after DB2 has detected that the SAS Embedded Process has terminated abnormally.
  - **Default:** 10
  - **Range:** 1–100
  - **Requirement:** Number-of-tries must be an integer value.
  - **Note:** When DB2 detects that the SAS Embedded Process has terminated abnormally, DB2 immediately attempts to re-spawn it. This argument limits the number of times that DB2 attempts to re-spawn the SAS Embedded Process. Once the retry count is exceeded, DB2 waits 15 minutes before trying to re-spawn it again.
**DIAGPATH:** *path*

specifies the path that indicates where the SAS Embedded Process diagnostic logs are written.

**Default**

DIAGPATH dbm configuration value

**Requirement**

The path must be fully qualified.

**DIAGLEVEL:** *level-number*

specifies the minimum severity level of messages that are captured in the SAS Embedded Process diagnostic logs. The levels are defined as follows.

1. **SEVERE**
2. **ERROR**
3. **WARNING**
4. **INFORMATIONAL**

**Default**

DIAGLEVEL dbm configuration value

**Range**

1–4

---

**Controlling the SAS Embedded Process for DB2**

The SAS Embedded Process starts when a query is submitted. The SAS Embedded Process continues to run until it is manually stopped or the database is shut down.

The DB2IDA command is a utility that is installed with the DB2 server to control the SAS Embedded Process. The DB2IDA command enables you to manually stop and restart the SAS Embedded Process without shutting down the database. You might use the DB2IDA command to upgrade or re-install the SAS Embedded Process library or correct an erroneous library path.

**Note:** DB2IDA requires IBM Fixpack 6 or later.

The DB2IDA command has the following parameters:

- **-provider sas**
  specifies the provider that is targeted by the command. The only provider that is supported is "sas".

- **-start**
  starts the SAS Embedded Process on the DB2 instance if the SAS Embedded Process is not currently running.
  If the SAS Embedded Process is running, this command has no effect.

**Note:** Once the SAS Embedded Process is started, the normal re-spawn logic in DB2 applies if the SAS Embedded Process is abnormally terminated.

- **-stop**
  stops the SAS Embedded Process if it is safe to do so.
If the SAS Embedded Process is stopped, this command has no effect.
If any queries are currently running on the SAS Embedded Process, the db2ida -stop command fails and indicates that the SAS Embedded Process is in use and could not be stopped.

Note: DB2 does not attempt to re-spawn the SAS Embedded Process once it has been stopped with the db2ida -stop command.

- stopforce
forces the SAS Embedded Process to shut down regardless of whether there are any queries currently running on it.
If the SAS Embedded Process is stopped, this command has no effect.
If any queries are currently running on the SAS Embedded Process, those queries receive errors.

Note: DB2 does not attempt to re-spawn the SAS Embedded Process once it has been stopped with the db2ida -stopforce command.

Here are some examples of the DB2IDA command:

db2ida -provider sas -stopforce

db2ida -provider sas -start

---

**DB2 Permissions**

There are two sets of permissions involved with the in-database software.

- The first set of permissions is needed by the person who publishes the SAS_COMPILEUDF and SAS_DELETEUDF functions and creates the SASUDF_COMPILER_PATH and SASUDF_DB2PATH global variables.
  These permissions must be granted before the %INDB2_PUBLISH_COMPILEUDF and %INDB2_PUBLISH_DELETEUDF macros are run. Without these permissions, running these macros fails.
  The following table summarizes the permissions that are needed by the person who publishes the functions and creates the global variables.
Table 4.3 Permissions Needed to Publish the Functions and Create Global Variables

<table>
<thead>
<tr>
<th>Permission Needed</th>
<th>Authority Required to Grant Permission</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATEIN permission for the SASLIB schema in which the SAS_COMPILEUDF and SAS_DELETEUDF functions are published and the SASUDF_COMPILER_PATH and SASUDF_DB2PATH global variables are defined</td>
<td>System Administrator or Database Administrator</td>
<td>GRANT CREATEIN ON SCHEMA SASLIB TO compiledeletepublisheruserid</td>
</tr>
<tr>
<td>CREATE_EXTERNAL_ROUTINE permission to the database in which the SAS_COMPILEUDF and SAS_DELETEUDF functions are published</td>
<td>Note: If you have SYSADM or DBADM authority or are the DB2 instance owner, then you have these permissions. Otherwise, contact your database administrator to obtain these permissions.</td>
<td>GRANT CREATE_EXTERNAL_ROUTINE ON DATABASE TO compile-delete-publisher-userid</td>
</tr>
</tbody>
</table>

- The second set of permissions is needed by the person who publishes the format or scoring model functions. The person who publishes the format or scoring model functions is not necessarily the same person who publishes the SAS_COMPILEUDF and SAS_DELETEUDF functions and creates the SASUDF_COMPILER_PATH and SASUDF_DB2PATH global variables. These permissions are most likely needed by the format publishing or scoring model developer. Without these permissions, the publishing of the format or scoring model functions fails.

  Note: Permissions must be granted for every format or scoring model publisher and for each database that the format or scoring model publishing uses. Therefore, you might need to grant these permissions multiple times.

  Note: If you are using the SAS Embedded Process to run your scoring functions, only the CREATE TABLE permission is needed.

After the DB2 permissions have been set appropriately, the format or scoring publishing macro should be called to register the formats or scoring model functions.

The following table summarizes the permissions that are needed by the person who publishes the format or scoring model functions.
### Table 4.4 Permissions Needed to Publish the Format or Scoring Model Functions

<table>
<thead>
<tr>
<th>Permission Needed</th>
<th>Authority Required to Grant Permission</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXECUTE permission for functions that have been published. This enables the person who publishes the formats or scoring model functions to execute the SAS_COMPILEUDF and SAS_DELETEUDF functions.</td>
<td>System Administrator or Database Administrator Note: If you have SYSADM or DBADM authority, then you have these permissions. Otherwise, contact your database administrator to obtain these permissions.</td>
<td>GRANT EXECUTE ON FUNCTION SASLIB.* TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>CREATE_EXTERNAL_ROUTINE permission to the database to create format or scoring model functions</td>
<td></td>
<td>GRANT CREATE_EXTERNAL_ROUTINE ON DATABASE TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>CREATE_NOT_FENCED_ROUTINE permission to create format or scoring model functions that are not fenced</td>
<td></td>
<td>GRANT CREATE_NOT_FENCED_ROUTINE ON DATABASE TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>CREATEIN permission for the schema in which the format or scoring model functions are published if the default schema (SASLIB) is not used</td>
<td></td>
<td>GRANT CREATEIN ON SCHEMA scoring-schema TO scoring-or-fmt-publisherid</td>
</tr>
<tr>
<td>CREATE TABLE permission to create the model table used in with scoring and the SAS Embedded Process</td>
<td></td>
<td>GRANT CREATETAB TO scoring-publisher-SEP-id</td>
</tr>
<tr>
<td>READ permission to read the SASUDF_COMPILER_PATH and SASUDF_DB2PATH global variables Note: The person who ran the %INDB2_PUBLISH_COMPILEUDF macro has these READ permissions and does not need to grant them to himself or herself again. Person who ran the %INDB2_PUBLISH_COMPILEUDF macro Note: For security reasons, only the user who created these variables has the permission to grant READ permission to other users. This is true even for the user with administrator permissions such as the DB2 instance owner.</td>
<td></td>
<td>GRANT READ ON VARIABLE SASLIB.SASUDF_DB2PATH TO scoring-or-fmt-publisher-id GRANT READ ON VARIABLE SASLIB.SASUDF_COMPILER_PATH TO scoring-or-fmt-publisher-id</td>
</tr>
</tbody>
</table>

Note: If you plan to use SAS Model Manager with the SAS Scoring Accelerator for in-database scoring, additional permissions are required. For more information, see “Configuring SAS Model Manager” in SAS In-Database Products: Administrator’s Guide.
Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Scoring Accelerator for DB2</td>
<td>“DB2 Administrator’s Guide” in SAS In-Database Products: Administrator’s Guide</td>
</tr>
</tbody>
</table>
Chapter 4 / Deploying SAS Embedded Process for DB2
Upgrading or Uninstalling SAS Embedded Process for DB2

Upgrade from a Previous DB2 Version – SAS 9.4M6 and Later

Upgrade from a Previous DB2 Version – Prior to SAS 9.4M6

Uninstall SAS Embedded Process for DB2

DB2 SAS Embedded Process Versions

Upgrade from a Previous DB2 Version – SAS 9.4M6 and Later

To find out when your version of SAS Embedded Process was released, see “DB2 SAS Embedded Process Versions” on page 38.

To upgrade from a version of SAS Embedded Process SAS 9.4M6 or a later release, follow these steps:

1. Log on as a user who owns the DB2 instance from a secured shell, such as SSH.

2. Enter the following command to find out whether SAS Embedded Process is running.
   
   `ps -ef | grep db2sas`

   For example, if SAS Embedded Process is running on Linux, here are typical results:

   `db2v10 20246 28078 0 Oct19 ? 00:00:27`
   `db2sas sep ,1,0,0,1,0,0,0,000,1,0,9958c,14,1e014,2,0,1,100080,0x210000000,0x210000000,160000,44498003,2,7e2b0085`

   If SAS Embedded Process is running on AIX, here are typical results:

   `db2v10 27721884 15597696 0 Oct 19 - 0:00 db2sase`
   `db2v10 29229304 12583050 0 Oct 19 - 0:00 db2sase`

3. Enter the following command to determine whether there are any users that are connected to the DB2 instance.
   
   `db2 list applications`
4 Stop SAS Embedded Process for DB2 using the db2ida command.

Note: If you are upgrading SAS Embedded Process, you do not need to shut down the database. The db2ida command enables you to upgrade only the SAS Embedded Process components without affecting clients that are already connected to the database. For more information about the db2ida command, see “Controlling the SAS Embedded Process for DB2” on page 29.

Use this command to stop SAS Embedded Process.

db2ida -provider sas -stop

If SAS Embedded Process is still running, an error occurs. Enter this command to force SAS Embedded Process to stop.

db2ida -provider sas -stopforce

5 Log on as a user who has root privileges from a secured shell, such as SSH.

6 Query the RPM files in the database.

```
psh "rpm -qa '*sepcore*'"
```

7 Delete the RPM file.

```
psh "rpm -e sas-sepcoredb2"
```

8 Continue with the installation steps in “DB2 Deployment Steps” on page 22.

---

Upgrade from a Previous DB2 Version – Prior to SAS 9.4M6

To find out when your version of SAS Embedded Process was released, see “DB2 SAS Embedded Process Versions” on page 38.

To upgrade from a previous version of SAS Embedded Process prior to SAS 9.4M6, follow these steps:

1 Enter the following command to find out whether SAS Embedded Process is running.

```
ps -ef | grep db2sasep
```

If SAS Embedded Process is running, here are typical results:

```
db2v10 23265382 20840668   0  Oct  6      -  4:03 db2sasep
db2v10 27983990 16646196   1 08:24:09 pts/10  0:00 grep db2sasep
```

2 Enter the following command to determine whether any users are connected to the DB2 instance.

```
db2 list applications
```

3 Stop SAS Embedded Process for DB2 using the db2ida command.
Note: If you are upgrading SAS Embedded Process, you do not need to shut down the database. The db2ida command enables you to upgrade only the SAS Embedded Process components without affecting clients that are already connected to the database. For more information about the db2ida command, see “Controlling the SAS Embedded Process for DB2” on page 29.

Use this command to stop SAS Embedded Process.

db2ida -provider sas -stop

If SAS Embedded Process is still running, an error occurs. Enter this command to force SAS Embedded Process to stop.

db2ida -provider sas -stopforce

4 Remove the SAS directory that contains the SAS Embedded Process binary files from the DB2 instance path.

   If you are upgrading a version of SAS Embedded Process prior to SAS 9.4M4, enter the following commands to change to the db2instancepath directory and remove the SAS directory. The db2instancepath path identifies the location of the SAS Embedded Process binary files in the DB2 instance.

      cd db2instancepath
      rm -fr SAS

   If you are upgrading a version of SAS Embedded Process starting with SAS 9.4M4, enter the following commands to change to the db2instancepath directory and remove the SAS directory. The db2instancepath path identifies the location of the SAS Embedded Process binary files in the DB2 instance.

      cd db2instancepath
      rm -fr SASEPHome

5 Continue with the installation steps in “DB2 Deployment Steps” on page 22.

Uninstall SAS Embedded Process for DB2

The method that you use to uninstall SAS Embedded Process for DB2 is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the last step, which is about continuing the installation.

   “Upgrade from a Previous DB2 Version – SAS 9.4M6 and Later” on page 35
   “Upgrade from a Previous DB2 Version – Prior to SAS 9.4M6” on page 36

To find out when your version of SAS Embedded Process was released, see “Greenplum SAS Embedded Process Versions” on page 54.
DB2 SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Starting with SAS 9.4M6, SAS 9.4 customers can get the latest version of SAS Embedded Process at any time.

Table 5.1 SAS Embedded Process Version by Release Date

<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcoredb2a-16.0.0-date.timestamp.x86_64.rpm</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoredb2a-16.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoredb2a-16.0.0-date.timestamp.amd.x86_64.deb</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoredb2l-16.0.0-date.timestamp.x86_64.rpm (Red Hat Enterprise Linux)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoredb2l-16.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoredb2l-16.0.0-date.timestamp.amd.x86_64.deb</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcoredb2a-13.00000-1.sh</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sepcoredb2l-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcoredb2a-12.00000-1.sh</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sepcoredb2l-12.00000-1.sh</td>
</tr>
<tr>
<td>July 2015</td>
<td>SAS 9.4M3</td>
<td>sepcoredb2a-9.43000-1.sh</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sepcoredb2l-9.43000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td></td>
<td></td>
<td>tkindbsrv-9.4_M2-n_r64.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td></td>
<td></td>
<td>tkindbsrv-9.4_M1-n_r64.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
PART 4

SAS Embedded Process Deployment for Greenplum

Chapter 6
Deploying SAS Embedded Process for Greenplum .................................. 41

Chapter 7
Upgrading or Uninstalling SAS Embedded Process for Greenplum ........ 51
Deploying SAS Embedded Process for Greenplum

Greenplum Prerequisites

- A specific version of the Greenplum client and server environment and the Greenplum Partner Connector (GPPC) API. For more information, see the SAS Foundation system requirements documentation for your operating system.
- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.
- The rsync utility must be installed on all nodes.
- Approximately 300 MB of disk space is required in the /opt file system on each Greenplum node.
- Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
Overview of SAS Embedded Process for Greenplum

SAS Embedded Process for Greenplum is available with the following SAS software products and must be deployed before you can use the in-database processing features of that software:

- SAS Scoring Accelerator for Greenplum
  For information about using this software, see SAS In-Database Products: User’s Guide.

- SAS In-Database Code Accelerator for Greenplum
  For information about using this software, see SAS In-Database Products: User’s Guide.

- SAS High-Performance Analytics
  For information about using this software, see SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide.

SAS Embedded Process is a SAS server process that runs within Greenplum to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that installed on your Greenplum system.

Note: To use your software, additional configuration is required. For more information, see “Where to Go from Here” on page 48.

Note: If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.

Greenplum Deployment Steps

To deploy SAS Embedded Process for Greenplum, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or reinstalling) a previous release in SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 7, “Upgrading or Uninstalling SAS Embedded Process for Greenplum,” on page 51</td>
</tr>
</tbody>
</table>
Deploying SAS Embedded Process for Greenplum

Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all the SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - Linux — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

   TIP For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you download it from Internet Explorer. For Windows, the file has a .zip extension.

Here is an example of a Linux command that is to unpack the SAS Mirror Manager:

```
tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz
```
3 (Optional) Run the following commands to provide platform and repository information that can be helpful if you need to specify the --platform or --repo options in the mirrormgr mirror command in the next step.

```bash
./mirrormgr list remote platforms --deployment-data path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
```

```bash
./mirrormgr list remote repos --deployment-data path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
```

4 Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

```bash
./mirrormgr mirror --deployment-data path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
```

**Note:** SAS Mirror Manager downloads all entitled repositories to the /user-home/sas_repos (Linux) or C:\Users\user-id\sas_repos (Windows) directory by default. You can change the location by using the --path path-to-mirror-repository option in the mirrormgr mirror command. You can also choose to download only one platform by using the --platform platform-name.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.

```bash
./mirrormgr mirror --deployment-data path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip --platform x64-suse-linux-12 --latest
```

**Note:** SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

---

Install the SAS Embedded Process RPM Package File for Greenplum

To install the SAS Embedded Process RPM package file, follow these steps:

1 Locate the SAS Embedded Process RPM package file, sas-sepcoregplm-version-date.timestamp.<rpm-os-designation>.x86_64.rpm.

The following table describes the file name options:
### Table 6.1  RPM File Name Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “Greenplum SAS Embedded Process Versions” on page 54.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time that the RPM package was created.</td>
</tr>
<tr>
<td>rpm-os-designation</td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For Greenplum, the permissible value is suse. There is no rpm-os-designation value for Red Hat Enterprise Linux.</td>
</tr>
</tbody>
</table>

The RPM file is located in one of the following directories depending on your operating system:

- **Linux** – /path-to-mirror-repository/repos/shipped/sepgplm/100.0/sepgplm-100.0.0-repo-os-designation<-option-if-specified>/Packages/s/
- **Windows** – C:\path-to-mirror-repository\repos\shipped\sepgplm\100.0\sepgplm-100.0.0-repo-os-designation<-option-if-specified>\Packages\s\

The following table describes the path options:

### Table 6.2  Path Options for the RPM File

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 43. Here is an example: /mysas/sas_repos/</td>
</tr>
</tbody>
</table>
2 Copy the sas-sepcoregplm-version-date.timestamp<.rpm-os-designation>.x86_64.rpm file to a location on the master node of the Greenplum server.

Make sure that you copy the file to the server machine according to the procedures that are used at your site.

Here is an example of a secure copy command:

```
scp sas-sepcoregplm-16.0.0-20180821.1822334532.suse.x86_64.rpm root@gplmmach1:/RPMTempDir
```

**Note:** `RPMTempDir` must be in a location where it can be accessed from at least one of the Greenplum nodes.

3 Navigate to the directory to which the RPM file was transferred in Step 2.

```
cd /RPMTempDir
```

4 Log on as a user who has root privileges from a secure shell, such as SSH.

5 Install the RPM file on the master node of the Greenplum server.

```
rpm -ivh /RPMTempDir/sas-sepcoregplm-version-date.timestamp<.rpm-os-designation>.x86_64.rpm
```

**Note:** You must have root privileges in order to install the RPM.

After the RPM file is installed, a `/opt/sas/ep/home/` directory is created in which the SAS Embedded Process files are installed. After the files are unpacked, the contents of the target directories should look similar to the following:

```
/opt/sas/ep/home/bin
/opt/sas/ep/home/misc
/opt/sas/ep/home/sasexe
/opt/sas/ep/home/security
/opt/sas/ep/home/utilities
```

The `/opt/sas/ep/home/bin/` directory contains several scripts:
The InstallSASEPFiles.sh file installs SAS Embedded Process. Step 7 explains how to run this file.

The StartupSASEP.sh and ShutdownSASEP.sh files enable you to manually start and stop SAS Embedded Process.

The UninstallSASEPFiles.sh file uninstalls SAS Embedded Process. The ShowSASEPStatus.sh file shows the status of SAS Embedded Process on each node.

For more information about running these scripts, see “Controlling the SAS Embedded Process” in SAS In-Database Products: Administrator’s Guide.

6 Log on as the Greenplum administrative user.

7 Use the following commands to install SAS Embedded Process to the master node and all host nodes.

**CAUTION**
The timing option must be set to off in order for the InstallSASEPFiles.sh script to work. Add `timing off` to your .psqlrc file before you run the script.

The InstallSASEPFiles.sh file must be run from the `/opt/sas/ep/home/bin/` directory.

```bash
cd /opt/sas/ep/home/bin
./InstallSASEPFiles.sh <-quiet | -verbose> <-nohostcheck>
```

- **-quiet** | **-verbose**
  - `verbose` is on by default. It enables you to see all messages that are generated during the installation process. Specify `-quiet` to suppress messages.

- **-nohostcheck**
  - `nohostcheck` specifies that the check for proper configuration of domain authentication for passwordless SSH is disabled and whether the rsync utility is installed.

  The script checks whether the RSA public key of the server is in the `.ssh/known_hosts` file. If the RSA public key is found, the script continues. If it is not found, the script fails and prints a warning that there is an issue with user authentication.

  The script also checks whether the rsync utility is installed on all nodes of the cluster. If the rsync utility is found, the script continues. If it is not found, the script fails and prints an error that the rsync utility has not been found.

  The installation also creates a `full-path-to-pkglibdir/SASEPHome/` directory. This directory is created on the master node and each host node.

  The installation also copies the SAS directories and files (from Step 5) to every node.

  The contents of the `full-path-to-pkglibdir/SASEPHome/` directory should look similar to the following:
Note: You can use the following command to determine the full-path-to-pkglibdir directory:

```
pg_config --pkglibdir
```

If you did not perform the Greenplum installation, you cannot run the `pg_config --pkglibdir` command. The `pg_config --pkglibdir` command must be run by the person who performed the Greenplum installation.

Here is an example of the SAS Embedded Process installation location:

```
usr/local/greenplum-db-4.2.6.1/lib/postgresql/SASEPHome/
```

The SAS Embedded Process will start when it is invoked by a process.

---

Greenplum Permissions

To publish the utility (SAS_COMPILEUDF, SAS_COPYUDF, SAS_DIRECTORYUDF, SAS_DEHEXUDF, SAS_EP), format, and scoring model functions, Greenplum requires that you have superuser permissions to create and execute these functions in the SASLIB (or other specified) schema and in the specified database.

In addition to Greenplum superuser permissions, you must have CREATE TABLE permission to create a model table when using the SAS Embedded Process.

If you plan to use SAS Model Manager with the SAS Scoring Accelerator for in-database scoring, additional permissions are required. For more information, see "Configuring SAS Model Manager" in SAS In-Database Products: Administrator's Guide.

---

Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software</td>
<td>Where to Go for Information</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SAS High-Performance Analytics</td>
<td>SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide</td>
</tr>
</tbody>
</table>
Upgrading or Uninstalling SAS Embedded Process for Greenplum

Upgrade from a Previous Greenplum Version – SAS 9.4M6 and Later .......................... 51
Upgrade an Existing Version of SAS Embedded Process for Greenplum By Performing a Full Deployment ................................................................. 51
Upgrade an Existing Version of SAS Embedded Process for Greenplum without Performing a Full Deployment .......................................................... 52
Upgrade from a Previous Greenplum Version – SAS 9.4M4 to SAS 9.4M5 ................ 52
Upgrade from a Previous Greenplum Version – SAS 9.3 ........................................... 53
Uninstall SAS Embedded Process for Greenplum .......................................................... 54
Greenplum SAS Embedded Process Versions ............................................................. 54

Upgrade from a Previous Greenplum Version – SAS 9.4M6 and Later

To find out when your version of SAS Embedded Process was released, see “Greenplum SAS Embedded Process Versions” on page 54.

To upgrade from a version of SAS Embedded Process SAS 9.4M6 or a later release, follow these steps. The upgrade process does not delete any scoring models or formats that were previously published.

1 Log on to the Greenplum master node as the Greenplum administrative user.

2 Uninstall SAS Embedded Process.
   a Navigate to the directory that contains the uninstall script.
     cd /opt/sas/ep/home/bin/
   b Run the UninstallSASEPFiles.sh file.
UninstallSASEPFiles.sh

This script stops SAS Embedded Process on each database host node. The script deletes the /opt/sas/ep/home/bin directory and all its contents from each database host node.

**CAUTION**
The timing option must be off in order for the .sh scripts to work. Add \timing off to your .psqlrc file before you run this script.

c Log on as a user with root privileges.

d Remove the RPM file.

ps \"rpm -e sas-sepcoregplm\"

You can verify that the RPM file was removed with the following command.

ps \"rpm -qa sas-sepcoregplm\"

3 Continue with the installation steps in “Deploying SAS Embedded Process for Greenplum” on page 43.

---

Upgrade an Existing Version of SAS Embedded Process for Greenplum without Performing a Full Deployment

Beginning in SAS 9.4M6, you can run the InstallSASEPFiles.sh script with a -sync option. The -sync option enables you to resynchronize SAS Embedded Process on all nodes in the cluster without performing a full deployment.

In this scenario, you can update the RPM file on the master node. Then you can run the InstallSASEPFiles.sh script with the -sync option to update all nodes on the cluster. This is the process that you would use when installing a hot fix.

---

**Note:** The -sync option should not be used when deploying a new version of SAS Embedded Process.

---

**CAUTION**
The timing option must be off in order for the .sh scripts to work. Add \timing off to your .psqlrc file before you run this script.

---

Upgrade from a Previous Greenplum Version – SAS 9.4M4 to SAS 9.4M5

To find out when your version of SAS Embedded Process was released, see “Greenplum SAS Embedded Process Versions” on page 54.
To upgrade from a previous version of SAS 9.4 SAS Embedded Process, follow these steps. If you upgrade SAS Embedded Process in this manner, you do not delete any scoring models or formats that were previously published.

1. Log on to the Greenplum master node as a superuser.

2. Run the UninstallSASEPFiles.sh file.
   
   ```
   ./UninstallSASEPFiles.sh
   ```

   This script stops SAS Embedded Process on each database host node. The script deletes the `/SAS/SASTKInDatabaseServerForGreenplum` directory and all its contents from each database host node.

   Prior to SAS 9.4M4 (November 2016), the UninstallSASEPFiles.sh file was in the `path_to_sh_file` directory to which you had copied the tkindbsrv self-extracting archive file.

   Starting with SAS 9.4M4 (November 2016) and later, the UninstallSASEPFiles.sh file is in the `path_to_sh_file/SASEPHome/bin` directory.

   **CAUTION**

   The timing option must be off in order for the UninstallSASEPFiles.sh scripts to work. Add `
timing off` to your `.psqlrc` file before you run this script.

3. Continue with the installation steps in “Deploying SAS Embedded Process for Greenplum” on page 43.

---

**Upgrade from a Previous Greenplum Version – SAS 9.3**

To find out when your version of SAS Embedded Process was released, see “Greenplum SAS Embedded Process Versions” on page 54.

To upgrade from a previous version of SAS 9.3 SAS Embedded Process, follow these steps:

1. Delete the `full-path-to-pkglibdir/SAS` directory that contains the SAS Formats Library and SAS Embedded Process.

   ```
   Note: You can use the following command to find out the `full-path-to-pkglibdir` directory.
   ```

   ```
   pg_config --pkglibdir
   ```

   If you did not perform the Greenplum installation, you cannot run the `pg_config --pkglibdir` command. The `pg_config --pkglibdir` command must be run by the person who performed the Greenplum installation.

   **CAUTION**
If you delete the SAS directory, all the scoring models that you published using scoring functions and all user-defined formats that you published are deleted. If you previously published scoring models using scoring functions or if you previously published user-defined formats, you must republish your scoring models and formats. If you used SAS Embedded Process to publish scoring models, the scoring models are not deleted.

It is a best practice to delete the SAS directory when you upgrade from a previous version. Deletion of the SAS directory ensures that you get the latest version of both the SAS Formats Library and SAS Embedded Process.

2. Continue with the installation steps in “Deploying SAS Embedded Process for Greenplum” on page 43.

Uninstall SAS Embedded Process for Greenplum

The method that you use to uninstall SAS Embedded Process for Greenplum is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the final step, which is about continuing the installation.

- “Upgrade from a Previous Greenplum Version – SAS 9.4M6 and Later” on page 51
- “Upgrade from a Previous Greenplum Version – SAS 9.4M4 to SAS 9.4M5” on page 52
- “Upgrade from a Previous Greenplum Version – SAS 9.3” on page 53

To find out when your version of SAS Embedded Process was released, see “Greenplum SAS Embedded Process Versions” on page 54.

Greenplum SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Starting with SAS 9.4M6, SAS 9.4 customers can get the latest version of SAS Embedded Process at any time.
<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcoregplm-16.0.0-date.timestamp.x86_64.rpm (Red Hat Enterprise Linux)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoregplm-16.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcoregplm-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcoregplm-12.00000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
Overview of SAS Embedded Process Deployment for Hadoop

Hadoop Prerequisites

The SAS Embedded Process deployment for Hadoop requires the following:

- A specific version of the Hadoop client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.

- You must have working knowledge of the Hadoop vendor distribution that you are using (for example, Cloudera or Hortonworks).

You also need working knowledge of the Hadoop Distributed File System (HDFS), MapReduce 2, YARN, and Hive services. For more information, see the Apache website or the vendor’s website.

For MapR, you must install the MapR client. The installed MapR client version must match the version of the MapR cluster to which SAS 9.4 or SAS Viya connects. For more information, see the MapR documentation.

If you want your SAS Embedded Process programs to run as a Spark job, Spark must be installed.

- You understand and can verify your Hadoop user authentication.

- You must have root authority to install the RPM files.

- The rsync utility must be installed.

- You understand and can verify your security setup. Security providers include Kerberos, Knox, Ranger, and Sentry.

  - Verify that you can use your defined security protocol to connect from your client machine to your Hadoop cluster.
- It is highly recommended that you enable Kerberos or another security protocol for data security. If your cluster is secured with Kerberos, you must obtain a Kerberos ticket. You also must have knowledge of any additional security policies.

- For clusters that have Kerberos security enabled, verify that you have a valid ticket on the node on which the Hive2 service is running.

- The installation of SAS Embedded Process involves writing a configuration file to HDFS and deploying files on all data nodes. Enable the HDFS user with Write permission to the root of HDFS. The HDFS user home directory, `/user/user-account`, must exist. If you deploy SAS Embedded Process manually, this user account is used.

- Verify that the Hadoop master node can connect to the Hadoop slave nodes using passwordless SSH. For more information, see the ssh-keygen and ssh-copy-id Linux manual pages.

  The passwordless SSH user must also have Read, Write, and Execute permissions on the installation directory. The directory structure of the nodes must match that of the installation directory.

- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.

- Identify the machine on which the Hive server is running. If the Hive server is not running on the same machine as the NameNode, note the server and port number of the Hive server for future configuration.

- Know the host name of the Hive server and the host name of the NameNode.

- For MapReduce:
  - Know the location of the MapReduce home directory.
  - Request permission to restart the MapReduce service.
  - Verify that you can run a MapReduce job successfully.

- For SAS Viya:
  - The CAS controller and each CAS worker node must have an IP address that can be routed to externally from the SAS Embedded Process nodes.

  - If you are using SAS Embedded Process, you can secure data transfers between your cluster and SAS Cloud Analytic Services (CAS). To use Transport Layer Security (TLS) with SAS Embedded Process, the following software is required on each node in the cluster: OpenSSL, version 1.0.1.g or later and appropriate CA certificates to match the server certificates that are configured.

- Approximately 300 MB of disk space is required in the `/opt` file system on each Hadoop node.

- Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
Overview of SAS Embedded Process for Hadoop

SAS Embedded Process for Hadoop is available with the following SAS software products and must be deployed before you can use the in-database processing features of that software:

- SAS Scoring Accelerator for Hadoop
  For more information about using this software, see “SAS Scoring Accelerator for Hadoop” in SAS In-Database Products: User’s Guide.

- SAS In-Database Code Accelerator for Hadoop
  For more information about using this software, see “Using the SAS In-Database Code Accelerator” in SAS In-Database Products: User’s Guide.

- SAS Data Loader for Hadoop
  For information about using this software, see SAS Data Loader for Hadoop: User’s Guide.

- In-Database Technologies for Hadoop (on SAS Viya) (includes the SAS Data Connect Accelerator for Hadoop and the SAS Data Connect Accelerator for Spark)
  For more information about using this software, see SAS Data Connector to Hadoop and SAS Data Connect Accelerator for Hadoop.

- SAS Contextual Analysis In-Database Scoring in Hadoop
  For more information, see SAS Contextual Analysis In-Database Scoring for Hadoop: User’s Guide.

- SAS High-Performance Analytics
  For information about using this software, see SAS High-Performance Analytics Server: User’s Guide.

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

Note: To use your software, additional configuration is required. For more information, see “Where to Go from Here” on page 48.

Note: If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.
Additional Software Components Included with SAS Embedded Process

The deployment steps that follow refer only to SAS Embedded Process. However, if you have licensed one or more of the products listed below, an additional SAS software component is included in SAS Embedded Process. When you install SAS Embedded Process, this additional component is also deployed.

<table>
<thead>
<tr>
<th>SAS Product Licensed</th>
<th>SAS Software Component Installed with SAS Embedded Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Data Loader for Hadoop</td>
<td>JAR file for SAS Data Quality Accelerator for Hadoop</td>
</tr>
<tr>
<td>SAS Contextual Analysis In-Database Scoring for Hadoop</td>
<td>DS2 and Threaded Kernel extensions</td>
</tr>
</tbody>
</table>

For more information about configuring the software for those products, see SAS Data Loader for Hadoop: Installation and Configuration Guide and SAS Contextual Analysis In-Database Scoring for Hadoop: Administrator’s Guide

Hadoop Permissions

The installation of the SAS Embedded Process for Hadoop involves writing a configuration file to HDFS and deploying files on all data nodes. These tasks require the following permissions:

- Writing the configuration file requires Write permission to HDFS.
  - The HDFS user home directory, `/user/user-account`, must exist. If you deploy the SAS Embedded Process, this user account is used when you manually deploy the SAS Embedded Process section.

- Deploying files across all nodes requires passwordless SSH from the master node to the slave nodes.

Hadoop Deployment Steps

To deploy SAS Embedded Process for Hadoop, follow these steps:
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or reinstalling) a previous release of SAS 9.4 or SAS Viya, follow these instructions. Otherwise, continue with step 2.</td>
<td>Chapter 13, “Upgrading or Uninstalling SAS Embedded Process for Hadoop,” on page 115</td>
</tr>
<tr>
<td>2</td>
<td>Install the Hadoop RPM files.</td>
<td>Chapter 9, “Installing the Hadoop RPM Files,” on page 65</td>
</tr>
</tbody>
</table>
| 3    | Deploy SAS Embedded Process, as appropriate:  
  - automatically using the cluster manager for the Hadoop distribution  
  - manually | Chapter 10, “Deploying SAS Embedded Process Using a Parcel or Stack,” on page 77  
  Chapter 11, “Deploying SAS Embedded Process Manually,” on page 89 |
| 4    | Review any additional configuration that might be needed depending on your Hadoop distribution. | Chapter 12, “Additional Configuration for SAS Embedded Process,” on page 103 |
| 5    | Review the permissions that are needed for in-database processing. | “Hadoop Permissions” on page 62 |
| 6    | If you license the following software, perform additional configuration as required:  
  - SAS Scoring Accelerator for Hadoop  
  - SAS Code Accelerator for Hadoop  
  - SAS Data Loader for Hadoop  
  - In-Database Technologies for Hadoop (on SAS Viya) (includes the SAS Data Connect Accelerator for Hadoop and Spark)  
  - SAS Contextual Analysis In-Database Scoring in Hadoop  
  - SAS High-Performance Analytics | Chapter 14, “Where to Go from Here,” on page 129 |
Overview of Hadoop RPM Files

SAS Embedded Process for Hadoop requires four RPM files:

- **sas-hadoopmrep**
  This file contains the SAS Hadoop MapReduce JAR file.

- **sas-dqacchadp**
  This file contains a JAR file for the SAS Data Quality Accelerator.

- **sas-sepcorehadp**
  This file contains SAS Embedded Process.

- **sas-ep-hadoop-media**
  This file contains the two scripts that enable you to create a parcel or stack.

**Note:** Ubuntu package files have a .DEB extension. In this guide, the file extension .RPM is used to denote both RPM and DEB files.

**CAUTION**

You must install all four RPM files in the order in which they are listed. Otherwise, the installation of the RPM files fails.
Install the Hadoop RPM Files

Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all the SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - Linux — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

   **TIP** For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you download it from Internet Explorer. For Windows, the file has a .zip extension.

   Here is an example of a Linux command that is used to unpack SAS Mirror Manager:

   ```
tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz
   ```

3. (Optional) Run the following commands to provide platform and repository information that can be helpful if you need to specify the --platform or --repo options in the mirrormgr mirror command in the next step.

   ```
   ./mirrormgr list remote platforms --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   
   ./mirrormgr list remote repos --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   ```

4. Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

   ```
   ./mirrormgr mirror --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   <options>
   ```
Note: SAS Mirror Manager downloads all entitled repositories to the `~/sas_repos` (Linux) or `C:\Users\user-id\sas_repos` (Windows) directory by default. You can change the location by using the `--path` option in the `mirrormgr` mirror command. You can also choose to download only one platform by using the `--platform` option.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.

```
./mirrormgr mirror --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/sas_Viya_deployment_data.zip
   --platform x64-suse-linux-12 --latest
```

Note: SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

---

**Install the SAS-HADOOPMREP RPM File**

To install the sas-hadoopmrep RPM file and the JAR files that it contains, follow these steps:

Note: Ubuntu package files have a DEB extension. In this guide, the file extension RPM is used to denote both RPM and DEB files.

1. Locate the sas-hadoopmrep RPM file, `sas-hadoopmrep-version-date.timestamp<.rpm-os-designation>.x86_64.rpm`.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>version</code></td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “Hadoop SAS Embedded Process Versions” on page 125.</td>
</tr>
<tr>
<td><code>date.timestamp</code></td>
<td>The date and time at which the RPM package was created.</td>
</tr>
<tr>
<td><code>rpm-os-designation</code></td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For Hadoop, the permissible values are either <code>suse</code> or <code>amd64</code> (for Ubuntu).</td>
</tr>
</tbody>
</table>
The RPM file is located in one of the following directories depending on your operating system:

- **Linux** — `/path-to-mirror-repository/repos/shipped/sephadoop/100.0/sephadoop-100.0.0-repo-os-designation<--option-if-specified>/Packages/s/`
- **Windows** — `C:\path-to-mirror-repository\repos\shipped\sephadoop\100.0\sephadoop-100.0.0-repo-os-designation<--option-if-specified>\Packages\s/`

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>path-to-mirror-repository</code></td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 66. Here is an example. /mysas/sas_repos/</td>
</tr>
</tbody>
</table>
| `repo-os-designation<--option-if-specified>`| `repo-os-designation` identifies the operating system. Available values depend on your data source and your software order. Possible values are as follows:  
- x64_redhat_linux_6-yum  
- x64_suse_linux_12-yum  
- x64_ubuntu_linux_14-apt  
  If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in `option-if-specified`. For example, if you specified `--latest --platform x64-redhat-linux-6` in your `mirrormgr` mirror command, the directory would be `x64_redhat_linux_6-yum-latest`. |

2. Copy the `sas-hadoopmrep-version-date.timestamp.<rpm-os-designation>.x86_64.rpm` file to the Hadoop master node. Make sure that you copy the file to the server machine according to the procedures that are used at your site.

Here is an example of a secure copy command:

```
scp sas-hadoopmrep-1.7.21-20190918.1568837393406.suse.x86_64.rpm root@hdpmach1:/RPMTempDir
```

Note: The `RPMTempDir` directory must be in a location where it can be accessed from at least one of the Hadoop nodes.

3. Navigate to the directory where the RPM file was transferred in Step 2.
Install the RPM file.

```
rpm -ivh sas-hadooprep-version-date.timestamp
   <.rpm-os-designation>.x86_64.rpm
```

After the RPM file is installed, a `/opt/sas/ep/home/jars/` directory is created in which the SAS Hadoop MapReduce JAR file is installed.

5 Install the sas-dqacchadm RPM file.

Follow the steps in “Install the SAS-DQACCHADP RPM File” on page 69.

---

**Install the SAS-DQACCHADP RPM File**

To install the sas-dqacchadm RPM file and the JAR files contained within, follow these steps:

---

**Note:** Ubuntu package files have a DEB extension. In this guide, the file extension RPM is used to denote both RPM and DEB files.

---

1 Locate the sas-dqacchadm RPM file, `sas-dqacchadm-version-date.timestamp<.rpm-os-designation>.x86_64.rpm`.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>version</code></td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td><code>date.timestamp</code></td>
<td>The date and time at which the RPM package was created.</td>
</tr>
<tr>
<td><code>rpm-os-designation</code></td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For Hadoop, the permissible values are either <code>suse</code> or <code>amd64</code> (for Ubuntu).</td>
</tr>
</tbody>
</table>

This file is located in one of the following directories depending on your operating system:

- Linux — `/path-to-mirror-repository/repos/shipped/sephadoop/100.0/
  sephadoop-100.0.0-repo-os-designation<--option-if-specified>/
  Packages/s/`
Windows — C:\path-to-mirror-repository\repos\shipped\sephadoop
\100.0\sephadoop-100.0.0-repo-os-designation<-option-if-specified>\Packages\s

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of &quot;Create a Mirror Repository&quot; on page 66. An example is /mysas/sas_repos/.</td>
</tr>
<tr>
<td>repo-os-designation</td>
<td>repo-os-designation designates the operating system. Which values are available depends on your data source and your software order. Possible values are as follows:</td>
</tr>
<tr>
<td></td>
<td>x64_redhat_linux_6-yum</td>
</tr>
<tr>
<td></td>
<td>x64_suse_linux_12-yum</td>
</tr>
<tr>
<td></td>
<td>x64_ubuntu_linux_14-apt</td>
</tr>
<tr>
<td></td>
<td>If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in option-if-specified. For example, if you specified --latest --platform x64-redhat-linux-6 in your mirrormgr mirror command, the directory would be x64_redhat_linux_6-yum-latest.</td>
</tr>
</tbody>
</table>

2 Copy the sas-dqacchadp-version-date.timestamp<.rpm-os-designation>.x86_64.rpm file to the Hadoop master node. Make sure that you copy the file to the server machine according to the procedures that are used at your site.

Here is an example of a secure copy command.

```
scp sas-dqacchadp-1.2.0-20190502.155681106790.suse.x86_64.rpm root@hdpmach1:/RPMTempDir
```

Note: The RPMTempDir directory must be in a location where it can be accessed from at least one of the Hadoop nodes.

3 Navigate to the directory where the RPM file was transferred in Step 2.

```
cd /RPMTempDir
```

4 Install the RPM file.

```
rpm -ivh sas-dqacchadp-version-date.timestamp<.rpm-os-designation>.x86_64.rpm
```

After the RPM file is installed, the SAS Data Quality Accelerator JAR file is put in the /opt/sas/ep/home/jars/ directory.
5 Install the sas-sepcorehadp RPM file.
   Follow the steps in "Install the SAS-SEPCOREHADP RPM File" on page 71.

Install the SAS-SEPCOREHADP RPM File

To install the sas-sepcorehadp RPM file and SAS Embedded Process, follow these steps:

---

Note: Ubuntu package files have a DEB extension. In this guide, the file extension RPM is used to denote both RPM and DEB files.

1 Locate the sas-sepcorehadp RPM file, sas-sepcorehadp-version-date.timestamp<.rpm-os-designation>.x86_64.rpm.
   The following table describes the file name options:

   **Table 9.5  RPM File Name Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>version</strong></td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “Hadoop SAS Embedded Process Versions” on page 125</td>
</tr>
<tr>
<td><strong>date.timestamp</strong></td>
<td>The date and time at which the RPM package was created.</td>
</tr>
<tr>
<td><strong>rpm-os-designation</strong></td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For Hadoop, the permissible values are either <strong>suse</strong> or <strong>amd64</strong> (for Ubuntu). There is no rpm-os-designation value for Red Hat Enterprise Linux.</td>
</tr>
</tbody>
</table>

This file is located in one of the following directories depending on your operating system:

- **Linux** — */path-to-mirror-repository/repos/shipped/sephadoop/100.0/sepadoop-100.0.0-repo-os-designation<--option-if-specified>/Packages/s/*

- **Windows** — *C:\path-to-mirror-repository\repos\shipped\sephadoop \100.0\sepadoop-100.0.0-repo-os-designation<--option-if-specified>\Packages\s/*

The following table describes the path options:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 66. An example is /mysas/sas_repos/.</td>
</tr>
</tbody>
</table>
| repo-os-designation<option-if-specified> | repo-os-designation identifies the operating system. Available values depend on your data source and your software order. Possible values are as follows:  
  - x64_redhat_linux_6-yum  
  - x64_suse_linux_12-yum  
  - x64_ubuntu_linux_14-apt  
If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in option-if-specified. For example, if you specified --latest --platform x64-redhat-linux-6 in your mirrormgr mirror command, the directory would be x64_redhat_linux_6-yum/latest. |

2 Copy the sas-sepcorehadp-version-date.timestamp<.rpm-os-designation>.x86_64.rpm file to the Hadoop master node. Make sure that you copy the file to the server machine according to the procedures that are used at your site. Here is an example of a secure copy command.

    scp sas-sepcorehadp-17.0.0-20191005.1822334532.suse.x86_64.rpm root@hdpmach1:/RPMTempDir

Note: The RPMTempDir directory must be in a location where it can be accessed from at least one of the Hadoop nodes.

3 Navigate to the directory where the RPM file was transferred in Step 2.

    cd /RPMTempDir

4 Install the RPM file.

    rpm -ivh sas-sepcorehadp-version-date.timestamp<.rpm-os-designation>.x86_64.rpm

After the RPM file is installed, a /opt/sas/ep/home/ directory is created where the EP files are installed. After the files are unpacked, the contents of the target directories should look similar to these.

    /opt/sas/ep/home/bin  
    /opt/sas/ep/home/conf  
    /opt/sas/ep/home/install  
    /opt/sas/ep/home/jars  
    /opt/sas/ep/home/license
Install the SAS-EP-HADOOP-MEDIA RPM File

To install the sas-ep-hadoop-media RPM file, follow these steps:

Note: Ubuntu package files have a DEB extension. In this guide, the file extension RPM is used to denote both RPM and DEB files.

1. Locate the sas-ep-hadoop-media RPM file, sas-ep-hadoop-media-version-date.timestamp<.rpm-os-designation>.x86_64.rpm.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file. For more information, see “Hadoop SAS Embedded Process Versions” on page 125.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time at which the RPM package was created.</td>
</tr>
<tr>
<td>rpm-os-designation</td>
<td>The operating system for which the RPM file was created. For Hadoop, the permissible values are either suse or amd64 (for Ubuntu). There is no rpm-os-designation value for Red Hat Enterprise Linux.</td>
</tr>
</tbody>
</table>

This file is located in one of the following directories depending on your operating system:

- Linux – /path-to-mirror-repository/repos/shipped/sephadoop/100.0/sephadoop-100.0.0-repo-os-designation<option-if-specified>/Packages/s/
- Windows – C:\path-to-mirror-repository\repos\shipped\sephadoop\100.0\sephadoop-100.0.0-repo-os-designation<option-if-specified>\Packages\s\
The following table describes the path options:

### Table 9.8 Path Options for the RPM File

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 66.</td>
</tr>
<tr>
<td></td>
<td>An example is /mysas/sas_repos/.</td>
</tr>
<tr>
<td>repo-os-designation&lt;—option-if-specified&gt;</td>
<td>repo-os-designation identifies the operating system. Which values are available depends on your data source and your software order.</td>
</tr>
<tr>
<td></td>
<td>Possible values are as follows:</td>
</tr>
<tr>
<td></td>
<td>- x64_redhat_linux_6-yum</td>
</tr>
<tr>
<td></td>
<td>- x64_suse_linux_12-yum</td>
</tr>
<tr>
<td></td>
<td>- x64_ubuntu_linux_14-apt</td>
</tr>
<tr>
<td></td>
<td>If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in option-if-specified. For example, if you specified --latest --platform x64-redhat-linux-6 in your mirrormgr mirror command, the directory would be x64_redhat_linux_6-yum-latest.</td>
</tr>
</tbody>
</table>

2 Copy the sas-ep-hadoop-media.version-date.timestamp<.rpm-os-designation>.x86_64.rpm file to the Hadoop master node. Make sure that you copy the file to the server machine according to the procedures that are used at your site.

Here is an example of a secure copy command.

```
scp sas-ep-hadoop-media-1.0.5-20180821.18223344532.suse.x86_64.rpm root@hdpmach1:/RPMTempDir
```

Note: The RPMTempDir directory must be in a location where it can be accessed from at least one of the Hadoop nodes.

3 Navigate to the directory where the RPM file was transferred in Step 2.

```
cd /RPMTempDir
```

4 Install the RPM file.

```
rpm -ivh sas-ep-hadoop-media-version-date.timestamp<.rpm-os-designation>.x86_64.rpm
```

After the RPM file is installed, a /opt/sas/ep/home/media directory is created where the EP files are installed on all nodes of the Hadoop cluster. After the files are unpacked, the contents of the /media directory should look similar to these.

```
/opt/sas/ep/home/media/bin
```
```
/opt/sas/ep/home/media/bin/create_parcel.sh
```
5 Deploy SAS Embedded Process.

You can either deploy manually or deploy automatically by using the cluster manager for your Hadoop distribution:

- You should deploy SAS Embedded Process manually under these conditions:
  - Your Hadoop distribution is MapR.
  - Your Hadoop distribution is Cloudera and any of the following is true:
    - Cloudera Manager is not installed.
    - You are not using Cloudera 5.8 or a later release.
  - Your Hadoop distribution is Hortonworks and any of the following are true:
    - Ambari is not installed or you are using a version of Ambari prior to 2.4.
    - You are not using Hortonworks 2.5 or a later release.

To deploy manually, see Chapter 11, “Deploying SAS Embedded Process Manually,” on page 89.

- You can use parcels and stacks to deploy SAS Embedded Process if the following conditions are met:
  - For Cloudera:
    - You are using Cloudera 5.8 or a later release. For the latest information, see the SAS Foundation system requirements documentation for your operating system.
    - Cloudera Manager is installed.
    - Your other SAS software, such as Base SAS and SAS/ACCESS Interface to Hadoop, was installed on a UNIX server.
  - For Hortonworks:
    - You are using Hortonworks 2.5 or a later release. For the latest information, see the SAS Foundation system requirements documentation for your operating system.
    - You are using Ambari 2.4 or a later release.
    - Your other SAS software, such as Base SAS and SAS/ACCESS Interface to Hadoop, was installed on a UNIX server.

- To deploy with Cloudera Manager, “Create and Deploy the SAS Embedded Process Parcel” on page 77.
- To deploy with Hortonworks Ambari, see “Create and Deploy the SAS Embedded Process Stack” on page 82.

**CAUTION**

After you have chosen a deployment method, you should continue to use that deployment method when uninstalling, upgrading or redeploying SAS Embedded Process. Otherwise, SAS Embedded Process can become unusable. For example, if you use parcels and stacks to deploy SAS software in the cluster, you should continue to use parcels and stacks to uninstall, upgrade or
redeploy the SAS software. You should not use the manual deployment method to uninstall, upgrade, or redeploy the software. If you do need to change deployment methods, you must first uninstall the SAS software in the cluster using the same method that you used to deploy it. You can then use the other deployment method to install the software.
Deploying SAS Embedded Process Using a Parcel or Stack

Create and Deploy the SAS Embedded Process Parcel

When to Deploy SAS Embedded Process Using Cloudera Manager
Parcel Creation and Deployment Steps
Create the Parcel
The create_parcel.sh Syntax
Linux Suffixes That Can Be Detected by the create_parcel.sh Script
Deploy the SAS Embedded Process Parcel on Cloudera

Create and Deploy the SAS Embedded Process Stack

When to Deploy SAS Embedded Process Using Ambari
Stack Creation and Deployment Steps
Create the Stack
The create_stack.sh Syntax
Deploying the SAS Embedded Process Stack on Hortonworks

Create and Deploy the SAS Embedded Process Parcel

When to Deploy SAS Embedded Process Using Cloudera Manager

You can use parcels to deploy SAS Embedded Process under the following conditions:

- You are using Cloudera 5.8 or a later release. For the latest information, see the SAS Foundation system requirements documentation for your operating system.
- Cloudera Manager is installed.
- Your other SAS software, such as Base SAS and SAS/ACCESS Interface to Hadoop, is installed on a UNIX server.

CAUTION
After you have chosen a deployment method, you should continue to use that deployment method when uninstalling, upgrading or redeploying SAS Embedded Process. Otherwise, SAS Embedded Process can become unusable. For example, if you use parcels and stacks to deploy the SAS software on the cluster, you should continue to use parcels and stacks to uninstall, upgrade, or redeploy. You should not use the manual deployment method to uninstall, upgrade, or redeploy. If you do need to change deployment methods, you must first uninstall the SAS software on the cluster using the same method that you used to deploy it. You can then use the other deployment method to install it.

Parcel Creation and Deployment Steps

To create and deploy the SAS Embedded Process parcel, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create the SAS Embedded Process parcel.</td>
<td>“Create the Parcel” on page 78</td>
</tr>
<tr>
<td>2</td>
<td>Deploy the SAS Embedded Process parcel to the cluster.</td>
<td>“Deploy the SAS Embedded Process Parcel on Cloudera” on page 80</td>
</tr>
</tbody>
</table>

Create the Parcel

To create the SAS Embedded Process parcel, follow these steps:

1. Create a directory that will become the target location for the parcel.
   
   For more information about the target location, see “The create_parcel.sh Syntax” on page 79.

2. Navigate to the directory where the create_parcel.sh self-extracting archive file exists.
   
   cd /opt/sas/ep/home/media/bin/
   
   This directory was created when you installed the sas-ep-hadoop-media RPM file. For more information, see “Install the SAS-EP-HADOOP-MEDIA RPM File” on page 73.

3. Run the create_parcel.sh script.
   
   sudo ./create_parcel.sh -t target-location
   
   The execution of the create_parcel.sh script requires root privileges. Use either the su command to become root or use the sudo command. This example uses the sudo command.
   
   When the script is run, it detects the installed Linux distribution.
   
   After the script is run, the parcel is put in the /target-location/sasep/cdh/ directory.
For more information about the options that are available for the create_parcel.sh syntax, see “The create_parcel.sh Syntax” on page 79.

### The create_parcel.sh Syntax

**create_parcel.sh**

- **t** target-location <-v new-suffix>

**Arguments**

- **-t target-location**
  Specifies the output location where the parcel is located.

  **Note** After you run the create_parcel.sh script, the parcel is copied to the /target-location/sasep/cdh/ directory.

- **-v new-suffix**
  Specifies the suffix of a new Linux distribution.

  **Note** This option should be used when a Linux distribution that is not detectable by the create_parcel.sh script is available.

**See** “Linux Suffixes That Can Be Detected by the create_parcel.sh Script” on page 79

### Linux Suffixes That Can Be Detected by the create_parcel.sh Script

The following table lists the Linux suffixes and distributions that can be automatically detected when you run the create_parcel.sh script.

**Table 10.1 Linux Distribution Suffixes**

| Distribution                                                                 | Suffix  |
|----------------------------------------------------------------------------|--|---|
| Debian 6.x (No CDH 5.x parcel provided)                                     | wheezy |
| Debian 6.x (No CDH 5.x parcel provided)                                     | squeeze |
| Red Hat Enterprise Linux 5 and clones (CentOS, Scientific Linux, and so on) | el5    |
| Red Hat Enterprise Linux 6 and clones (CentOS, Scientific Linux, and so on) | el6    |
| Red Hat Enterprise Linux 7 and clones (CentOS, Scientific Linux, and so on) | el7    |
| SUSE Linux Enterprise Server 11.x                                            | sles11 |
Deploy the SAS Embedded Process Parcel on Cloudera

After you create the SAS Embedded Process parcel, you must distribute and activate the parcel on the cluster. Follow these steps:

Note: More than one SAS Embedded Process parcel can be deployed on your cluster, but only one parcel can be activated at one time. Before activating a new parcel, deactivate the old one.

1. Log on to Cloudera Manager.
2. Distribute the parcel to all nodes.
   a. From the menu bar, select **Hosts ➔ Parcels**.
      The **SASEP** parcel is located under your cluster. An example name for the parcel is p0.1.
      
      Note: If the **SASEP** parcel is missing, run **Check for New Parcels**.
   b. On the row for the **SASEP** parcel, click **Distribute** to copy the parcel to all nodes.

   **CAUTION**

   SAS Embedded Process must be copied to all YARN Node nodes. Hive and HCatalog must be available on all nodes where SAS Embedded Process is distributed. Otherwise, SAS Embedded Process does not function properly.

   You can log on to the node to see the contents in the /opt/cloudera/parcel directory.

3. Click **Activate**. Answer OK to the Activation prompt. You might be prompted to either restart the cluster or to close the window.
CAUTION
Do not restart the cluster.

This step creates a symbolic link to the SAS Hadoop JAR file.
If prompted, click Close.

Note: When you activate the parcel, files that are created are placed in the /opt/sasep/ directory.

4 Generate the SAS EP configuration file. The method that you use to generate the configuration file depends on whether you have sudo access.
- If you have sudo access, run the sasep-admin.sh script on the cluster node.
  ```shell
  sudo /opt/sasep/home/bin/sasep-admin.sh -genconfig
  ```
- If you do not have sudo access, manually copy the ep-config.xml configuration file to HDFS.

Note: The user must have Write permission to the HDFS root folder. If your Hadoop cluster is secured with Kerberos, the user who copies the configuration file to HDFS must have a valid Kerberos ticket.

1 Log on as an HDFS user or as the user that can access HDFS.
2 Create the /sas/ep/config directory for the configuration file.
  ```shell
  hadoop fs -mkdir -p /sas/ep/config
  ```
3 Navigate to the /opt/sasep/home/conf directory.
4 Use the Hadoop copyFromLocal command to copy the ep-config.xml file to HDFS.
  ```shell
  hadoop fs -copyFromLocal ep-config.xml /sas/ep/config/ep-config.xml
  ```

5 Verify that the configuration file, ep-config.xml, was written to HDFS.
  ```shell
  hadoop fs -ls /sas/ep/config/ep-config.xml
  hadoop fs -cat /sas/ep/config/ep-config.xml
  ```

Note: If your Hadoop cluster is secured with Kerberos, you must have a valid Kerberos ticket in order to access HDFS. Otherwise, you can use the WebHDFS browser.

Note: The /sas/ep/config directory is created automatically when you run the installation script with sudo access. If you used the -genconfig option to specify a non-default location, use that location for the ep-config.xml file.

6 If you used a non-default location, you must add a configuration property to the mapred-site.xml configuration file that is used on the client side.

```xml
<property>
  <name>sas.ep.config.filename</name>
  <value>config-file-location-on-hdfs</value>
</property>
```
Note: The config-file-location-on-hdfs is the location of the SAS Embedded Process configuration file on HDFS.

7 Review any additional configuration that might be needed depending on your Hadoop distribution.

For more information, see Chapter 12, “Additional Configuration for SAS Embedded Process,” on page 103.

8 Validate the deployment of SAS Embedded Process by running a program that uses SAS Embedded Process and the MapReduce service. An example is a scoring program.

Create and Deploy the SAS Embedded Process Stack

When to Deploy SAS Embedded Process Using Ambari

You can use stacks to deploy SAS Embedded Process under the following conditions:

- You are using Hortonworks 2.5 or a later release. For the latest information, see the SAS Foundation system requirements documentation for your operating system.
- You are using Ambari 2.4 or a later release.
- Your other SAS software, such as Base SAS and SAS/ACCESS Interface to Hadoop, was installed on a UNIX server.

CAUTION
After you have chosen a deployment method, you should continue to use that deployment method when uninstalling, upgrading or redeploying SAS Embedded Process. Otherwise, SAS Embedded Process can become unusable. For example, if you use parcels and stacks to deploy the SAS software on the cluster, you should continue to use parcels and stacks to uninstall, upgrade or redeploy. You should not use the manual deployment method to uninstall, upgrade, or redeploy. If you do need to change deployment methods, you must first uninstall the SAS software on the cluster using the same method that you used to deploy it. You can then use the other deployment method to install it.
Stack Creation and Deployment Steps

To create and deploy the SAS Embedded Process stack, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create the SAS Embedded Process stack.</td>
<td>“Create the Stack” on page 83</td>
</tr>
<tr>
<td>2</td>
<td>Deploy the SAS Embedded Process stack to the cluster.</td>
<td>“Deploying the SAS Embedded Process Stack on Hortonworks” on page 84</td>
</tr>
</tbody>
</table>

Create the Stack

To create the SAS Embedded Process stack, follow these steps:

1. Create a directory that will become the target location for the stack.
   For more information about the target location, see “The create_stack.sh Syntax” on page 83.

2. Navigate to the directory where the create_stack.sh self-extracting archive file exists.
   cd /opt/sas/ep/home/media/bin/
   This directory was created when you installed the sas-ep-hadoop-media RPM file. For more information, see "Install the SAS-EP-HADOOP-MEDIA RPM File" on page 73.

3. Run the create_stack.sh script.
   sudo ./create_stack.sh -t target-location -a ambari-admin-name -r y
   The execution of the create_stack.sh script requires root privileges. Use either the su command to become root or use the sudo command. This example uses the sudo command.
   When the script is run, it detects the installed Linux distribution.
   After the script is run, the stack is put in the /target-location/sasep/hdp/ directory.
   For more information about the options that are available for the create_stack.sh syntax, see “The create_stack.sh Syntax” on page 83.

The create_stack.sh Syntax

create_stack.sh - t target-location -a ambari-admin-name -r | -p ssl-port

Arguments
-t **target-location**
  specifies the output location where the stack is located.

  **Note**: After you run the create_stack.sh script, the stack is copied to the `/target-location/sasep/hdp/` directory.

-a **ambari-admin-name**
  specifies the user name of the Ambari administrator.

-`y | n`
  specifies whether to restart the Ambari server after the stack is deployed.

-p **ssl-port**
  specifies the SSL port number if the Ambari server is enabled with SSL.

---

**Deploying the SAS Embedded Process Stack on Hortonworks**

**Deploying the SAS Embedded Process Stack for the First Time**

After you create the SAS Embedded Process stack, you must distribute and activate the stack on the cluster. Follow these steps:

**Note**: If the SAS Embedded Process stack already exists on your cluster, follow the instructions in “Deploying a New Version of the SAS Embedded Process Stack” on page 87.

1. Log on to the machine that is hosting Ambari.
2. Start the Ambari server and log on.
3. If the requiretty option was enabled when you deployed SAS Embedded Process, you must restart the Ambari server at this time. Otherwise, skip to step 4.
   a. Log on to the cluster.
      ```bash
      sudo - su
      ```
   b. Restart the Ambari server.
      ```bash
      sudo ambari-server restart
      ```
   c. Start the Ambari server and log on.
4. Click **Actions** and choose **+ Add Service**.
   The **Add Service Wizard** page and the **Choose Services** panel is displayed.
5. In the **Choose Services** panel, select the **SASEP** service. Click **Next**.
   The **Assign Slaves and Clients** panel is displayed.
6. In the **Assign Slaves and Clients** panel, ensure that NameNode, HDFS_CLIENT, and HCAT_CLIENT are selected where you want the stack to be
deployed. By default, the three clients are selected. SAS recommends that you select all clients.

**CAUTION**
SAS Embedded Process must be deployed on all YARN Node nodes. Hive and HCatalog must be available on all nodes where SAS Embedded Process is deployed. Otherwise, SAS Embedded Process does not function properly.

---

Note: On the Assign Slaves and Clients panel, place your pointer over the host name to view the details for NameNode, HDFS_CLIENT, and HCAT_CLIENT.

---

7 Click **Next**. The **Customize Services** panel is displayed. The SASEP service stacks are listed. An example stack name is s0.1.

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

---

Note: By default, Ambari will not retain the credentials that you provide unless you have configured encrypted passwords for storage in Ambari. If you have not configured Ambari for password encryption, you will be prompted to provide credentials whenever cluster changes are made.

---

If your cluster is secured with Kerberos, the **Configure Identities** panel is displayed. Enter your Kerberos credentials in the **admin_principal** and **admin_password** text boxes. Click **Next**.

---

Note: If you have a Hortonworks v3.1 cluster that is Kerberos-enabled, you might get a Create Keytabs error on the Background Operations / Install Services panel. In most cases, the SAS Embedded Process has been deployed successfully. When **Ambari-2351** is resolved, this error will not occur.

---

The **Review** panel is displayed.

9 Review the information in the panel. If the information is correct, click **Deploy**.

---

The **Install, Start, and Test** panel is displayed. After the SAS Embedded Process stack is installed on all nodes, click **Next**.

---

The **Summary** panel is displayed.

10 Click **Complete**. The SAS Embedded Process stack is now installed on all nodes of the cluster.

---

The **SASEP** service is displayed on the Ambari dashboard.

---

Note: When you add the stack to your Hadoop cluster, files that are created are placed in the `/opt/sasep/` directory.

---

11 Generate the SAS Embedded Process configuration file. How you do this depends on whether you have sudo access.
If you have sudo access, run the sasep-admin.sh script on the cluster node to generate the SAS Embedded Process configuration file.

```
sudo /opt/sasep/home/bin/sasep-admin.sh -genconfig
```

If you do not have sudo access, manually copy the ep-config.xml configuration file to HDFS.

---

**Note:** The user must have Write permission to the HDFS root folder. If your Hadoop cluster is secured with Kerberos, the user who copies the configuration file to HDFS must have a valid Kerberos ticket.

1. Log on as an HDFS user or as the user that can access HDFS.
2. Create the `/sas/ep/config` directory for the configuration file.
   
   ```
   hadoop fs -mkdir -p /sas/ep/config
   ```

3. Navigate to the `/opt/sasep/home/conf` directory.
4. Use the Hadoop copyFromLocal command to copy the ep-config.xml file to HDFS.
   
   ```
   hadoop fs -copyFromLocal ep-config.xml /sas/ep/config/ep-config.xml
   ```

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

````
hadoop fs -ls /sas/ep/config/ep-config.xml
hadoop fs -cat /sas/ep/config/ep-config.xml
````

**Note:** If your Hadoop cluster is secured with Kerberos, you must have a valid Kerberos ticket in order to access HDFS. Otherwise, you can use the WebHDFS browser.

**Note:** The `/sas/ep/config` directory is created automatically when you run the installation script with sudo access. If you used the -genconfig option to specify a non-default location, use that location for the ep-config.xml file.

13 If you used a non-default location, you must add a configuration property to the mapred-site.xml configuration file that is used on the client side.

```xml
<property>
  <name>sas.ep.config.filename</name>
  <value>config-file-location-on-hdfs</value>
</property>
```

**Note:** The `config-file-location-on-hdfs` is the location of the SAS Embedded Process configuration file on HDFS.

14 Review any additional configuration that might be needed depending on your Hadoop distribution.

For more information, see Chapter 12, “Additional Configuration for SAS Embedded Process,” on page 103.
15 **Validate the deployment of SAS Embedded Process by running a program that uses SAS Embedded Process and the MapReduce service. An example is a scoring program.**

**Deploying a New Version of the SAS Embedded Process Stack**

More than one SAS Embedded Process stack can be deployed on your cluster, but only one stack can be activated at one time. After you create the SAS Embedded Process stack, follow these steps to deploy an additional SAS Embedded Process stack when one already exists on your cluster.

1 Log on to the machine that is hosting Ambari.
2 Restart the Ambari server and log on to the Ambari manager.
3 Select SASEP.
   In the Services panel, a restart symbol appears next to SASEP. The Configs tab indicates that a restart is required.
4 Click Restart.
5 Click Restart All.

After the service is restarted, the previous version of SAS Embedded Process still appears in the activated_version text box on the Configs tab. All deployed versions of the SAS Embedded Process stack should appear in the sasep_allversions text box.

6 Refresh the browser.
   The new version of SAS Embedded Process should now appear as the activated_version text box on the Configs tab.

---

**Note:** When you add the stack to your Hadoop cluster, files that are created are placed in the /opt/sasep/ directory.

---

If, at any time, you want to activate another version of the SAS Embedded Process stack, follow these steps:

1 Enter the version number in the activated_version text box on the Configs tab.
2 Click Save.
3 Add a note describing your action (for example, “Changed from version s01.1 to s01.2”), and click Next.
4 Click Restart.
5 Click Restart All.
6 Refresh Ambari.
   The new service is activated.
7 Review any additional configuration that might be needed depending on your Hadoop distribution.
For more information, see Chapter 12, “Additional Configuration for SAS Embedded Process,” on page 103.

8 Validate the deployment of SAS Embedded Process by running a program that uses SAS Embedded Process and the MapReduce service. An example is a scoring program.
When to Deploy SAS Embedded Process Manually

You should deploy SAS Embedded Process manually under these conditions:

- Your Hadoop distribution is MapR.
- Your Hadoop distribution is Cloudera and any of the following is true:
  - Cloudera Manager is not installed.
  - You are not using Cloudera 5.8 or a later release.
- Your Hadoop distribution is Hortonworks and any of the following are true:
  - Ambari is not installed or you are using a version of Ambari prior to 2.4.
  - You are not using Hortonworks 2.5 or a later release.

**CAUTION**

After you have chosen a deployment method, you should continue to use that deployment method when uninstalling, upgrading or redeploying SAS Embedded Process. Otherwise, SAS Embedded Process can become unusable. For example, if you use parcels and stacks to deploy the SAS software in the cluster, you should continue to use parcels and stacks to uninstall, upgrade, or redeploy the SAS software. You should not use the manual deployment method to uninstall, upgrade, or redeploy. If you do need to change deployment methods, you must first uninstall the SAS software on the cluster using the same method that you used to deploy it. You can then use the other deployment method to install it.
To deploy SAS Embedded Process manually, follow these steps:

**Note:** Passwordless SSH is required in order to install SAS Embedded Process for Hadoop. Also, Write permission to Hadoop Distributed File System (HDFS) might be required.

1. Navigate to the location of the sasep-admin.sh file on the Hadoop cluster.
   
   ```
   cd /opt/sas/ep/home/bin/
   ```

2. If your Hadoop cluster is secured with Kerberos and you have sudo access, the HDFS user must have a valid Kerberos ticket in order to access HDFS. You can obtain a valid Kerberos ticket with the `kinit` command.
   
   ```
   sudo su - root
   su - hdfs | hdfs-userid
   kinit -kt location-of-keytab-file-user-for-which-you-are-requesting-a-ticket principal-name
   exit
   ```

   **Note:** For all Hadoop distributions except MapR, the default HDFS user is `hdfs`. For MapR distributions, the default HDFS user is `mapr`. You can specify a different user ID with the `-hdfsuser` argument when you run the `sasep-admin.sh -add` script. If you use a different HDFS superuser, ensure that the user has a home directory in HDFS before you run the `sasep-admin.sh -add` script. For example, if the HDFS superuser is `prodhdfs`, ensure that the `/user/prodhdfs` directory exists in HDFS.

3. Run the sasep-admin.sh script to deploy SAS Embedded Process across all nodes where the YARN Node Manager is installed. How you run the script depends on whether you have sudo access.

**TIP** To check the status of your Kerberos ticket on the server, as the HDFS user, run the `klist` command. Here is an example of the command and its output:

   ```
   klist
   Ticket cache: FILE/tmp/krb5cc_493
   Default principal: hdfs@HOST.COMANY.COM
   Valid starting Expires Service principal
   06/20/18 09:51:26 06/27/18 09:51:26
   krbtgt/HOST.COMANY.COM@HOST.COMANY.COM
   renew until 06/18 09:51:26
   ```
Note: It is recommended that you run the sasep-admin.sh script from the /opt/sas/ep/home/bin/ location.

**TIP**  Many options are available for installing SAS Embedded Process. SAS recommends that you review the script syntax before running it. For more information, see "The sasep-admin.sh Syntax" on page 94.

- If you have sudo access:
  
  Run the sasep-admin.sh script.

  ```
  ./sasep-admin.sh -add
  ```

  The sasep-admin.sh script detects the Hadoop cluster topology and installs SAS Embedded Process on all YARN Node Manager nodes. The script also detects whether you have sudo privileges to access the HDFS superuser account. The script installs SAS Embedded Process on the host node from which you run the script. You are not required to install SAS Embedded Process on the YARN Resource Manager node if no Node Manager is running on it. SAS Embedded Process is installed on the host node from which you run the script even if a YARN Resource Manager or a YARN Node Manager is not present. To add SAS Embedded Process to new nodes at a later time, you should run the script with the `-host <hosts>` option.

**CAUTION**

SAS Embedded Process must be installed on all nodes that are capable of running a YARN container (MapReduce 2 or Spark). Hive and HCatalog must be available on all nodes on which SAS Embedded Process is installed. Otherwise, SAS Embedded Process does not function properly.

In addition, a configuration file, ep-config.xml, is automatically created and written to the /opt/sas/ep/home/conf directory. If you have sudo access, the SAS Embedded Process configuration file is automatically copied to the HDFS file system in the /sas/ep/config directory.

- If you do not have sudo access, follow these steps to manually copy the ep-config.xml configuration file to HDFS.

  **Note:** This step must be performed by a user that has Write permission to the HDFS root folder. If your Hadoop cluster is secured with Kerberos, the user who copies the configuration file to HDFS must have a valid Kerberos ticket.

  1. Log on as your HDFS user or as the user that is assigned to access HDFS.

  2. Create the /sas/ep/config directory for the configuration file.

     ```
     hadoop fs -mkdir -p /sas/ep/config
     ```

  3. Navigate to the /opt/sas/ep/home/conf directory.

     ```
     cd /opt/sas/ep/home/conf
     ```
4 Use the Hadoop `copyFromLocal` command to copy the ep-config.xml file to HDFS.

```
hadoop fs -copyFromLocal ep-config.xml /sas/ep/config/ep-config.xml
```

4 Verify that SAS Embedded Process is installed by running the `sasep-admin.sh` script with the `-check` option. By default, this command verifies that SAS Embedded Process was installed where a YARN Node Manager is also installed.

```
cd /opt/sas/ep/home/bin/
./sasep-admin.sh -check
```

This command verifies that SAS Embedded Process was installed on the hosts that you specified.

```
cd /opt/sas/ep/home/bin/
./sasep-admin.sh -check -hostfile host-list-filename | -host "host-list" 
```

**Note:** The `sasep-admin.sh -check` script does not run successfully if SAS Embedded Process is not installed.

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

```
hadoop fs -ls /sas/ep/config/
hadoop fs -cat /sas/ep/config/ep-config.xml
```

**Note:** If your cluster is secured with Kerberos, you must have a valid Kerberos ticket in order to access HDFS. Otherwise, you can use the WebHDFS browser.

**Note:** The `/sas/ep/config` directory is created automatically when you run the installation script with sudo access. If you used the `-genconfig` option to specify a non-default location, use that location to find the ep-config.xml file.

6 When using a non-default location, you must add a configuration property to the sasep-site.xml configuration file that is used on the client side.

```
<property>
  <name>sas.ep.config.file</name>
  <value>config-file-location-on-hdfs</value>
</property>
```

**Note:** The value, `config-file-location-on-hdfs`, specifies the location of the SAS Embedded Process configuration file on HDFS.
The sasep-admin.sh Script

Overview of the sasep-admin.sh Script

The sasep-admin.sh script enables you to perform the following actions.

- Install or uninstall SAS Embedded Process on a single node or a group of nodes.
- Check whether SAS Embedded Process is installed correctly.
- Display the Hadoop configuration environment.
- Generate a SAS Embedded Process configuration file and write the file to an HDFS location.
- Display Hadoop version information for the Hadoop cluster.
- Display the SAS license file to the data nodes or remove them from the data nodes.
- Create a link to the latest SAS Embedded Process JAR file.
- Display all live data nodes on the cluster.
- Change ownership of the SAS Embedded Process root folder.
- (SAS Viya only) Deploy the security settings for SAS Data Connect Accelerator for Hadoop across all nodes in the cluster.
- Synchronize any change to SAS Embedded Process across other nodes on the cluster.
- Display the version of SAS Embedded Process that is installed.
- Display all YARN nodes in the Hadoop cluster.

Note: The installation of SAS Embedded Process for Hadoop involves writing a configuration file to HDFS and deploying files on all YARN Node Manager nodes. These two tasks can occur automatically, depending on your Hadoop and HDFS permissions.

If you run the SAS Embedded Process installation script (sasep-admin.sh) with sudo access, the script detects the Hadoop cluster topology and installs SAS Embedded Process on all YARN Node Manager nodes. The installation script also installs SAS Embedded Process on the host node on which you run the script. In addition, a configuration file, ep-config.xml, is created and written to HDFS.

If you do not have sudo access, you must manually copy the ep-config.xml configuration file to HDFS. For more information, see Step 3 on page 90.
Note: The Hadoop master node must be able to connect to the Hadoop slave nodes using passwordless SSH on the cluster where SAS Embedded Process is installed.

The sasep-admin.sh Syntax

```
sasep-admin.sh -add
    <hostfile host-list-filename | -host <"host-list""> |
    | -yarnm <"resourcemanagerhost:port""> |
    | -maxparallel number-of-connections |
    | -hdfsuser user-id |
    | -nohostcheck |

sasep-admin.sh -check
    <hostfile host-list-filename | -host <"host-list""> |
    | -yarnm <"resourcemanagerhost:port""> |
    | -hdfsuser user-id |

sasep-admin.sh -env

sasep-admin.sh -genconfig
    <HDFS-filename>
    | -force |

sasep-admin.sh -hadoopversion

sasep-admin.sh -license deploy | remove
    <hostfile host-list-filename | -host <"host-list""> |
    | -yarnm <"resourcemanagerhost:port""> |
    | -force |

sasep-admin.sh -linklatest
    <sshidf your-pem-file>

sasep-admin.sh -nodelist

sasep-admin.sh -prep
    <hostfile host-list-filename | -host <"host-list""> > |
    | -sshidf your-pem-file |

sasep-admin.sh -remove
    <hostfile host-list-filename | -host <"host-list""> |
    | -yarnm <"resourcemanagerhost:port""> |
    | -hdfsuser user-id |
    | -nohostcheck |

sasep-admin.sh -security deploy | reset
    <hostfile host-list-filename | -host <"host-list""> |
    | -yarnm <"resourcemanagerhost:port""> |
    | -force |

sasep-admin.sh -sync
    <hostfile host-list-filename | -host <"host-list""> > |
    | -maxparallel number-of-connections |
    | -hdfsuser user-id |
```
The `sasep-admin.sh` Script

<sashdf your-pem-file
<nohostcheck>

### sasep-admin.sh -version

### sasep-admin.sh -yarnnodes

\[ <\text{yarnrm} <\text{resourcemanagerhost:port} > \] >

#### Action and Informational Arguments

**-add**

- **Type**: Action
- **Requirement**: You must use the `-yarnrm` option if the node from which you run the `sasep-admin.sh` script does not have the yarn-site.xml configuration file in the directory path.
- **Tip**: If you add nodes to the cluster, you can specify the hosts on which you want to install SAS Embedded Process by using the `-hostfile` or `-host` option. The `-hostfile` or `-host` options are mutually exclusive.

#### See

- `-hostfile on page 100`
- `-host on page 100`
- `-yarnrm on page 101`

**-check**

- **Type**: Informational
- **Requirements**: You must use the `-yarnrm` option if the node from which you run the `sasep-admin.sh` script does not have the yarn-site.xml configuration file in the directory path.
  - You can specify the hosts for which you want to check SAS Embedded Process by using the `-hostfile` or `-host` option. The `-hostfile` or `-host` options are mutually exclusive.
- **Note**: Starting in **SAS 9.4M6**, the `-check` option validates that SAS Embedded Process is found on all remote nodes. Also, it validates that the number of files in the local installation are identical to the number on files on all remote nodes.

#### See

- `-hostfile on page 100`
- `-host on page 100`
- `-yarnrm on page 101`

**-env**

- **Type**: Informational
  - displays the SAS Embedded Process installation script and the Hadoop configuration environment.
-genconfig <HDFS-filename> <-force>
generates a new SAS Embedded Process configuration file in the /opt/sas/ep/
home/conf directory of the local file system.

<table>
<thead>
<tr>
<th>Type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interactions</td>
<td>When used with sudo access, the script creates the ep-config.xml configuration file and writes it to both the /opt/sas/ep/home/conf directory on the local file system and the /sas/ep/config/ directory on HDFS. You can change the file name and HDFS location by using the HDFS-filename argument. HDFS-filename must be the fully qualified HDFS pathname where the configuration file is located. When used without sudo access, the script does not write the configuration file to HDFS. You must manually copy the file to HDFS. For more information, see Step 3 on page 90.</td>
</tr>
<tr>
<td>Note</td>
<td>The -genconfig argument creates two identical configuration files under /opt/sas/ep/home/conf on the local file system: ep-config.xml and sasep-site.xml. The sasep-site.xml file might be copied to the client side under a folder that is in the classpath path. When the sasep-site.xml file is loaded from the classpath, the configuration file at the HDFS location is not used. However, if sasep-site.xml is not found in the classpath path, a configuration file must exist on HDFS, either on the default HDFS location /sas/ep/config/ep-config.xml or in the location that is set in the sas.ep.config.file property.</td>
</tr>
<tr>
<td>Tips</td>
<td>Use the -genconfig argument to generate a new SAS Embedded Process configuration file if you upgrade your Hadoop installation, you install or upgrade your Hive or HCatalog services, or you upgrade the JDK or JRE that is used by the Hadoop processes. This argument generates an updated ep-config.xml file. Use the -force argument to overwrite the existing configuration file. Use the HDFS-filename argument to specify another location and configuration file name. If you decide to generate the configuration file in a non-default HDFS location, you must set the sas.ep.config.file property in the mapred-site.xml file to the value that you specify in the -genconfig option.</td>
</tr>
</tbody>
</table>
| Examples | The following example generates the configuration files under /opt/sas/ep/home/conf on the local file system and the ep-config.xml configuration file under /sas/ep/config on the HDFS: 
./sasep-admin.sh -genconfig The following example overwrites the configuration files under /opt/sas/ep/home/conf on the local file system and under /sas/ep/config on the HDFS, if it already exists: 
./sasep-admin.sh -genconfig -force The following example generates the configuration files under /opt/sas/ep/home/conf on the local file system and under /home/hadoop/ on the HDFS: 
./sasep-admin.sh -genconfig /home/hadoop/ep-config.xml |
The following example generates the configuration files under `/opt/sas/ep/home/conf` on the local file system only:

```
./sasep-admin.sh -genconfig
```

The following example overwrites the configuration files under `/opt/sas/ep/home/conf` on the local file system only:

```
./sasep-admin.sh -genconfig -force
```

### -hadoopversion

Displays Hadoop version information for the cluster.

<table>
<thead>
<tr>
<th>Type</th>
<th>Informational</th>
</tr>
</thead>
</table>

### -license deploy | remove `<-force>`

Deploys the `seitinitsas` license file to the YARN Node Manager nodes or removes it from the YARN Node Manager nodes.

<table>
<thead>
<tr>
<th>Type</th>
<th>Action</th>
</tr>
</thead>
</table>

**Requirement**

You must use the `-yarnrm` option if the node from which you run the `sasep-admin.sh` script does not have the `yarn-site.xml` configuration file in the directory path.

**Note**

To overwrite license settings without a prompt, use the `-force` argument.

**Tip**

You can specify one or more hosts for which you want to deploy or remove security settings for SAS Embedded Process by using the `-hostfile` or `-host` option. The `-hostfile` option and the `-host` option are mutually exclusive.

**See**

- `-hostfile` on page 100
- `-host` on page 100
- `-yarnrm` on page 101

### -linklatest

Creates a link to the latest SAS Embedded Process JAR file in the `/opt/sas/ep/home/jars` folder.

<table>
<thead>
<tr>
<th>Type</th>
<th>Action</th>
</tr>
</thead>
</table>

**Alias**

`linksasepjaronly`

### -nodelist

Displays all live DataNodes on the Hadoop cluster.

<table>
<thead>
<tr>
<th>Type</th>
<th>Informational</th>
</tr>
</thead>
</table>

**Requirement**

`sudo` access is required.

### -prep

Recursively changes the ownership of the SAS Embedded Process install root folder `/opt/sas/ep` to the current user on the provided list of hosts. If the list of hosts is not provided, the script retrieves the list of YARN Node Manager nodes from the cluster configuration.
### Type Action

### Requirement

- `sudo access is required.`

### Tip

You can specify one or more hosts for which you want to deploy or reset security settings for SAS Embedded Process by using the `-hostfile` or `-host` option. The `-hostfile` option and the `-host` option are mutually exclusive.

### -remove

removes SAS Embedded Process.

---

**CAUTION**

(SAS 9.4 only) If you are using SAS Data Loader for Hadoop, you should remove the Quality Knowledge Base (QKB) and the SAS Data Management Accelerator for Spark from the Hadoop nodes before removing SAS Embedded Process. Removing SAS Embedded Process removes the scripts that are used to remove these products. For more information, see [Remove SAS Software from the Hadoop Cluster](#).

### Type Action

### Requirement

- You must use the `-yarnrm` option if the node from which you run the sasep-admin.sh script does not have the `yarn-site.xml` configuration file in the directory path.

### Interactions

- If you have sudo access, the script automatically retrieves the list of data nodes from the Hadoop configuration. In addition, the script automatically removes the `epconfig.xml` file from HDFS.

### Tip

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

### See

- `-hostfile on page 100`
- `-host on page 100`
- `-yarnrm on page 101`

### -security deploy | reset <force>

deploys or resets security settings across all nodes in the cluster.

### Type Action

### Restriction

- This option is supported only in SAS Viya.

### Requirement

- You must use the `-yarnrm` option if the node from which you run the sasep-admin.sh script does not have the `yarn-site.xml` configuration file in the directory path.
### The sasep-admin.sh Script

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Note</strong></td>
<td>To overwrite security settings without a prompt, use the <code>-force</code> argument.</td>
</tr>
<tr>
<td><strong>Tip</strong></td>
<td>You can specify one or more hosts for which you want to deploy or reset</td>
</tr>
<tr>
<td></td>
<td>security settings for SAS Embedded Process by using the <code>-hostfile</code> or</td>
</tr>
<tr>
<td></td>
<td><code>-host</code> option. The <code>-hostfile</code> option and the <code>-host</code> option are</td>
</tr>
<tr>
<td></td>
<td>mutually exclusive.</td>
</tr>
<tr>
<td><strong>See</strong></td>
<td><code>-hostfile on page 100</code></td>
</tr>
<tr>
<td></td>
<td><code>-host on page 100</code></td>
</tr>
<tr>
<td></td>
<td><code>-yarnrm on page 101</code></td>
</tr>
<tr>
<td></td>
<td>“Encrypt Data Transfer when Using the SAS Data Connect Accelerator” in</td>
</tr>
<tr>
<td></td>
<td><em>Encryption in SAS Viya: Data in Motion</em></td>
</tr>
</tbody>
</table>

#### `-sync`

Synchronizes any change to SAS Embedded Process across other nodes on the cluster.

**Type** Action

**Requirements**

- The `rsync` utility is required.
- Synchronization must be done using the same user ID that was used to perform the initial software installation.
- The `-sync` option synchronizes the local SAS Embedded Process install with the provided list of hosts. If the list of hosts is not provided, the script retrieves the list of YARN Node Manager nodes from the cluster configuration.

**Tip**

- The `-sync` option can be used for installing hot fixes. Hot fixes should be installed using the installation instructions that are provided by SAS Technical Support.

#### `-version`

Displays the version of SAS Embedded Process that is installed.

**Type** Informational

#### `-yarnnodes`

Lists all YARN nodes in the Hadoop cluster.

**Type** Informational

### Parameters for Action and Informational Arguments

#### `-hdfsuser user-id`

Specifies the user ID that has Write access to HDFS root directory.

**Defaults**

- `hdfs` for Cloudera or Hortonworks
- `mapr` for MapR (SAS 9.4 only)

**Requirement**

- The `hdfs` folder `/users/user-id` must exist. Otherwise, the command fails.
This argument has no affect if you do not have sudo access.

Use the -hdfsuser argument with the -add, -check, or -remove argument to change, check, or remove the HDFS user ID.

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

-host <"host-list"> specifies the target host or host list on which SAS Embedded Process is installed or removed.

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

Use the -hostfile argument with the -add, -check, -license, -prep, -remove, -security, or -sync arguments.

"-hdfsuser user-id" on page 99

-hostfile host-list-filename specifies the full path of a file that contains the list of hosts on which SAS Embedded Process is installed or removed.

Use the -hostfile argument with the -add, -check, -license, -prep, -remove, -security, or -sync arguments.

"-hdfsuser user-id" on page 99

-hostfile /opt/sasep/ep.hosts

-maxparallel number-of-connections controls the number of parallel connections for operations in the installation process such as the secure copy of JAR files or the resynchronization of directories across nodes.

Default 10

Use this argument with the -add or -sync argument.

-nohostcheck specifies that the check for proper configuration of domain authentication for passwordless SSH is disabled.

The sasep-admin.sh script checks whether the RSA public key of the server is in the .ssh/known_hosts file. If it is found, the script continues. If it is not found, the script will stop and print a warning that there is an issue with user authentication.

Use the -nohostcheck argument with the -add, -remove, or -sync arguments.
CAUTION There are security risks if you specify -nohostcheck. If the host that you are connecting to has been infected by a man-in-the-middle attack, the attacker could potentially replay the destination server's challenge back to you. The attack misleads you to believe that you are connecting to the remote resource when, in fact, the attacker is using your credentials to connect to that resource.

-sshidf your-pem-file
specifies the file from which the identity (private key) for public key authentication is read. Use this option when the user who is installing SAS Embedded Process does not have passwordless access between the master nodes and the slaves nodes.

Interaction Use the -sshidf argument with the -linklatest, -prep, or -sync arguments.

-yarnrm <"">resourcemanagerhost:portnumber<"">>
specifies the YARN Resource Manager web application host name and port number in the format resourcemanagerhost:portnumber. This option might be used when the yarn-site.xml property file is not present in the Hadoop configuration folder, or when the folder where yarn-site.xml resides is not in the path. If this information is not specified, the script looks for the YARN Resource Manager web application host name and port number in yarn-site.xml.

Requirement If you do not specify the -yarnrm argument, then the host name and port number must be in yarn-site.xml.

Interactions Use the -yarnrm argument when the host name and port number are not in yarn-site.xml.

Use the -yarnrm argument with the -add, -check, -license, -remove, -security, or -yarnnodes argument to specify the YARN Resource Manager web application host name and port number.

Example sasep-admin.sh -check -yarnrm myhostname:8088
Overview of Additional Configuration Tasks

After you have installed SAS Embedded Process either manually or by deploying the parcel or stack, the following additional configuration tasks must be performed:

- “Additional Configuration Needed to Use HCatalog File Formats” on page 104.
- “Adding the YARN Application CLASSPATH for MapR” on page 105.
- “Changing the Trace Level” on page 106.
- “Adjusting the SAS Embedded Process Performance” on page 107.

Additional Configuration Needed to Use HCatalog File Formats

Overview of HCatalog File Types

Prerequisites for HCatalog Support

SAS Client Configuration

SAS Server-Side Configuration

Adding the YARN Application CLASSPATH for MapR

Changing the Trace Level

Adjusting the SAS Embedded Process Performance

Specifying the Amount of Memory That SAS Embedded Process Uses

Additional Configuration for All Hadoop Distributions – SAS 9.4

Additional Configuration for MapR Version 6.0

Additional Configuration for Amazon Elastic MapReduce (EMR)

Additional Configuration to Access Amazon S3 Object Storage

Additional Configuration to Access Amazon Web Services (AWS)

Configuring Temporary Folders When Writing Data to a Hive Table Using SAS Embedded Process

Additional Configuration for Hortonworks 3.1 with Patch HDP 3.1.0.29-2

Additional Configuration for the SAS Embedded Process for Spark Continuous Session
Additional Configuration Needed to Use HCatalog File Formats

Overview of HCatalog File Types

HCatalog is a table management layer that presents a relational view of data in the HDFS to applications within Hadoop. With HCatalog, data structures that are registered in the Hive metastore, including SAS data, can be accessed through Hadoop code. HCatalog is part of Apache Hive.

SAS Embedded Process for Hadoop uses HCatalog to process complex, non-delimited file formats.

Prerequisites for HCatalog Support

If you plan to access complex, non-delimited file types, such as Avro or Parquet, the following conditions must be met:

- The Hive JAR files must be on all the nodes in the clusters.
- Hive and HCatalog must be installed on all the nodes in the cluster.

SAS Client Configuration

Note: If you used the Hadoop tracer script to install the Hadoop JAR files, these configuration tasks are not necessary. The Hadoop tracer script also performed SAS client configuration.
The following additional configuration tasks must be performed:

- The hive-site.xml configuration file must be in the SAS_HADOOP_CONFIG_PATH environment variable (SAS 9.4) or the hadoopConfigDir path (SAS Viya).
- The following Hive or HCatalog JAR files must be in the SAS_HADOOP_JAR_PATH (SAS 9.4) or the hadoopJarPath (SAS Viya)
  
  hive-hcatalog-core-*\.jar
 hive-webhcat-java-client-*\.jar

For more information about the hadoopConfigDir and hadoopJarPath paths, see the CASLib statement in SAS Viya Cloud Analytic Services: Language Reference.

For more information about the SAS_HADOOP_JAR_PATH and SAS_HADOOP_CONFIG_PATH environment variables, see SAS Hadoop Configuration Guide for Base SAS and SAS/ACCESS.

---

SAS Server-Side Configuration

The SAS Embedded Process installation automatically sets the HCatalog CLASSPATH in the ep-config.xml file. Otherwise, you must manually append the HCatalog CLASSPATH to the MapReduce configuration property, mapreduce.application.classpath, in the mapred-site.xml file on the client side.

Here is an example of an HCatalog CLASSPATH for a Cloudera distribution.

```bash
/opt/cloudera/parcels/CDH-version/bin/../lib/hive/lib/*
```

Here is an example of an HCatalog CLASSPATH for a Hortonworks distribution:

```bash
/usr/hdp/2.4.2.0-258/hive/lib/*
```

Adding the YARN Application CLASSPATH for MapR

The two configuration properties that specify the YARN application CLASSPATH must be modified on the client side: yarn.application.classpath and mapreduce.application.classpath. If you do not specify the YARN application CLASSPATH, MapR uses the default CLASSPATH. However, if you specify the MapReduce application CLASSPATH, the YARN application CLASSPATH is ignored. SAS Embedded Process for Hadoop requires both the YARN application CLASSPATH and the MapReduce application CLASSPATH.

To ensure that the YARN application CLASSPATH exists, you must manually add the YARN application CLASSPATH to the yarn-site.xml file. Without the manual definition in the configuration file, the MapReduce application master fails to start a YARN container.

Here is the default YARN application CLASSPATH for Linux:
Here is the default YARN application CLASSPATH for Windows:

%HADOOP_CONF_DIR%;
%HADOOP_COMMON_HOME%/share/hadoop/common/*;
%HADOOP_COMMON_HOME%/share/hadoop/common/lib/*;
%HADOOP_HDFS_HOME%/share/hadoop/hdfs/*;
%HADOOP_HDFS_HOME%/share/hadoop/hdfs/lib/*;
%HADOOP_YARN_HOME%/share/hadoop/yarn/*;
%HADOOP_YARN_HOME%/share/hadoop/yarn/lib/*

Note: On MapR, the YARN application CLASSPATH does not resolve the symbols or variables that are included in pathnames such as $HADOOP_HDFS_HOME.

TIP To apply any change that you make to the cluster, you must restart the node managers.

Changing the Trace Level

You can modify the level of tracing by changing the value of the sas.ep.server.trace.level property in the ep-config.xml file. The default value is 4 (TRACE_NOTE).

<property>
  <name>sas.ep.server.trace.level</name>
  <value>trace-level</value>
</property>

The trace-level represents the level of trace that is produced by SAS Embedded Process. Values for trace-level can be one of the following:

0  TRACE_OFF
1  TRACE_FATAL
2  TRACE_ERROR
3  TRACE_WARN
4  TRACE_NOTE
Adjusting the SAS Embedded Process Performance

You can adjust how SAS Embedded Process runs by changing its properties or adding properties. The SAS Embedded Process configuration properties can be added to the mapred-site.xml configuration on the client side or the sasep-site.xml file. If you change the properties in the ep-config.xml file, which is located in the HDFS, it affects all the SAS Embedded Process jobs. The ep-config.xml file is created automatically when you install SAS Embedded Process.

When using SAS Embedded Process, you can adjust several properties in order to improve performance. Note that performance properties are valid only when SAS Embedded Process runs as a MapReduce job.

- You can specify the number of SAS Embedded Process MapReduce tasks per node by changing the sas.ep.superreader.tasks.per.node property in the ep-config.xml file. The default number of tasks is 6.

  The SAS Embedded Process super reader technology does not use the standard MapReduce split calculation. Instead of assigning one split per task, it assigns many splits per task. The super reader calculates the splits, groups them, and distributes the groups to a configurable number of mapper tasks according to the location of the data.

  <property>
    <name>sas.ep.superreader.tasks.per.node</name>
    <value>number-of-tasks</value>
  </property>

- You can specify the number of concurrent nodes that are allowed to run High-Performance Analytics and Cloud Analytic Services (CAS) Parallel Data Connector output tasks by changing the sas.ep.hpa.output.concurrent.nodes property.

  If this property is set to 0, SAS Embedded Process allocates tasks on all nodes that are capable of running a YARN container. If this property is set to -1, the number of concurrent nodes corresponds to the number of High-Performance Analytics worker nodes. If the number of concurrent nodes exceeds the number of available nodes, the property value is adjusted to the number of available nodes. The default value is 0.

  <property>
    <name>sas.ep.hpa.output.concurrent.nodes</name>
    <value>number-of-nodes</value>
  </property>

---

Note: Tracing can produce a significant volume of output. If tracing is not required for troubleshooting or monitoring, set the trace-level value to 0.
You can specify the number of High-Performance Analytics and CAS Parallel Data Connector output tasks that are allowed to run per node by changing the sas.ep.hpa.output.tasks.per.node property.

The default number of tasks is 1.

You can specify the number of concurrent input reader threads.

Each reader thread takes a file split from the input splits queue, opens the file, positions itself at the beginning of the split, and starts reading the records. Each record is stored in a native buffer that is shared with the DS2 container. When the native buffer is full, its contents are pushed to the DS2 container for processing. After a reader thread finishes reading a file split, it takes another file split from the input splits queue. The default number of input threads is 3.

You can specify the number of output writer threads by changing the sas.ep.output.threads property.

The SAS Embedded Process super writer technology improves performance by writing output data in parallel, which produces multiple parts of the output file per mapper task. Each writer thread is responsible for writing one part of the output file. The default number of output threads is 2.

You can specify the number of compute threads by changing the sas.ep.compute.threads property.

Each compute thread runs one instance of the DS2 program inside SAS Embedded Process. The DS2 code that runs inside the DS2 container processes the records that it receives. At a given point, DS2 flushes output data to native buffers. The super writer threads take the output data from DS2 buffers and writes them to the super writer thread output file at a designated HDFS location. When all file input splits are processed and all output data is flushed and written to the HDFS, the mapper task ends. The default number of compute threads is 1.

You can specify the number of buffers that are used for output data by changing the sas.ep.output.buffers property in the mapred-site.xml file.

The number of output buffers should not be less than sas.ep.compute.threads plus sas.ep.output.threads. The default number of buffers is 3.
You can specify the number of native buffers that are used to cache input data by changing the sas.ep.input.buffers property in the mapred-site.xml file. The default value is 4. The number of input buffers should not be less than sas.ep.compute.threads plus sas.ep.input.threads.

You can specify the optimal size of one input buffer by changing the sas.ep.optimal.input.buffer.size property in the mapred-site.xml file. The optimal row array size is calculated based on the optimal buffer size. The default value is 1 MB.

Specifying the Amount of Memory That SAS Embedded Process Uses

SAS Embedded Process is managed by the Hadoop MapReduce framework. Load balancing and resource allocation are managed by YARN. You can adjust the YARN container limits to change the amount of memory that SAS Embedded Process uses.

For information about how CAS uses memory, see “Memory” in SAS Cloud Analytic Services: Fundamentals.

Additional Configuration for All Hadoop Distributions – SAS 9.4

Depending on the in-database product that you are running on Base SAS with SAS Embedded Process, you might get a java.lang.OutOfMemoryError: PermGen space error. This message indicates that the Java Virtual Machine PermGen space has been exhausted. The PermGen space is not part of the Java Heap memory. It is used to hold metadata that describes the classes and the method that are used by an application.

The PermGen size can be increased to alleviate this issue by adding this line to JREOPTIONS in the SAS configuration file:
Additional Configuration for MapR Version 6.0

In Version 6.0 of MapR, the mapreduce.jobhistory properties have a value of 0.0.0.0 for the node on which your job history server is running.

To ensure that SAS Embedded Process runs correctly, you must substitute the name of the node on which your job history server is running for 0.0.0.0 in all of the mapreduce.jobhistory properties in the mapred-site.xml file. Here are some examples where 0.0.0.0 has been replaced with the node name maprsrv3.unx.comp.com:

```xml
<property>
  <name>mapreduce.jobhistory.address</name>
  <value>maprsrv3.unx.comp.com:10020</value>
</property>
<property>
  <name>mapreduce.jobhistory.webapp.address</name>
  <value>maprsrv3.unx.comp.com:19888</value>
</property>
<property>
  <name>mapreduce.jobhistory.webapp.https.address</name>
  <value>maprsrv3.unx.comp.com:19890</value>
</property>
```

Additional Configuration for Amazon Elastic MapReduce (EMR)

Additional Configuration to Access Amazon S3 Object Storage

Hadoop Amazon Web Services (AWS) and S3 object storage can be configured in different ways. For more information, see Hadoop-AWS module: Integration with Amazon Web Services.

S3A is an Apache Hadoop connector to Amazon S3. S3A supports multiple authentication mechanisms and can be configured to identify the mechanisms to use and the order in which the mechanisms are used.

There are two ways to configure credentials for the Apache Hadoop S3A client using temporary AWS credentials provider:

- Authenticate by setting the following Hadoop configuration properties in the core-site.xml file.
## Hadoop Configuration Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Property setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>fs.s3a.access.key</td>
<td>AWS access key ID</td>
<td><code>&lt;property&gt;</code>&lt;br&gt;<code>&lt;name&gt;fs.s3a.access.key&lt;/name&gt;</code>&lt;br&gt;<code>&lt;value&gt;your-key&lt;/value&gt;</code>&lt;br&gt;<code>&lt;/property&gt;</code></td>
</tr>
<tr>
<td>fs.s3a.secret.key</td>
<td>AWS secret key</td>
<td><code>&lt;property&gt;</code>&lt;br&gt;<code>&lt;name&gt;fs.s3a.secret.key&lt;/name&gt;</code>&lt;br&gt;<code>&lt;value&gt;your-secret-key&lt;/value&gt;</code>&lt;br&gt;<code>&lt;/property&gt;</code></td>
</tr>
<tr>
<td>fs.s3a.session.token</td>
<td>the session token, when using org.apache.hadoop.fs.s3a.TemporaryAWSCredentialsProvider as one of the providers</td>
<td><code>&lt;property&gt;</code>&lt;br&gt;<code>&lt;name&gt;fs.s3a.session.token&lt;/name&gt;</code>&lt;br&gt;<code>&lt;value&gt;your-token&lt;/value&gt;</code>&lt;br&gt;<code>&lt;/property&gt;</code></td>
</tr>
<tr>
<td>fs.s3a.aws.credentials.provider</td>
<td>a comma-separated class names of credential provider classes</td>
<td><code>&lt;property&gt;</code>&lt;br&gt;<code>&lt;name&gt;fs.s3a.aws.credentials.provider&lt;/name&gt;</code>&lt;br&gt;<code>&lt;value&gt;org.apache.hadoop.fs.s3a.TemporaryAWSCredentialsProvider&lt;/value&gt;</code>&lt;br&gt;<code>&lt;/property&gt;</code></td>
</tr>
<tr>
<td>fs.s3a.endpoint</td>
<td>AWS region</td>
<td><code>&lt;property&gt;</code>&lt;br&gt;<code>&lt;name&gt;fs.s3a.endpoint&lt;/name&gt;</code>&lt;br&gt;<code>&lt;value&gt;s3.us-east-1.amazonaws.com&lt;/value&gt;</code>&lt;br&gt;<code>&lt;/property&gt;</code></td>
</tr>
</tbody>
</table>

### Authenticate by using environment variables

S3A supports configuration via the standard AWS environment variables. A full description of AWS environment variables can be found at [Configuring the AWS CLI: Environment Variables](#).

To set the authentication credentials, instead of using properties in the Hadoop configuration, the following environment variables can be used:

## Environment Variables for Setting Authentication Credentials

<table>
<thead>
<tr>
<th>Environment Variable</th>
<th>Description</th>
<th>Example of Setting the Environment Variable in a SAS Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS_ACCESS_KEY_ID</td>
<td>AWS access key</td>
<td>option set=AWS_ACCESS_KEY_ID=&quot;your-key&quot;;</td>
</tr>
<tr>
<td>Environment Variable</td>
<td>Description</td>
<td>Example of Setting the Environment Variable in a SAS Session</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>AWS_SECRET_ACCESS_KEY</td>
<td>AWS secret key Access and secret key variables override credentials that are stored in credential and configuration files.</td>
<td>option set=AWS_SECRET_ACCESS_KEY=&quot;your-secret-key&quot;;</td>
</tr>
<tr>
<td>AWS_SESSION_TOKEN</td>
<td>a session token if you are using temporary security credentials</td>
<td>option set=AWS_SESSION_TOKEN=&quot;your-token&quot;;</td>
</tr>
<tr>
<td>AWS_DEFAULT_REGION</td>
<td>AWS region. This variable overrides the default region of the in-use profile, if set.</td>
<td>option set=AWS_DEFAULT_REGION=&quot;region-name&quot;;</td>
</tr>
</tbody>
</table>

Additional Configuration to Access Amazon Web Services (AWS)

Amazon EMR provides a script that monitors the health of the network. To ensure that this background script does not shut down SAS Embedded Process for Hadoop or SAS Embedded Process for Spark, you can specify -Xrs options to disable signal handling in the Java Virtual Machine (JVM).

- For SAS Embedded Process for Hadoop, specify these -Xrs JVM start-up options in the mapred-site.xml file:

```
<property>
  <name>mapreduce.map.java.opt</name>
  <value>-Xrs -Xmx6144m</value>
</property>
<property>
  <name>mapreduce.reduce.java.opts</name>
  <value>-Xrs -Xmx12288m</value>
</property>
<property>
  <name>yarn.app.mapreduce.am.command-opts</name>
  <value>-Xrs -Xmx12288m</value>
</property>
```

- For SAS Embedded Process for Spark, specify these -Xrs JVM start-up options in the spark-default.conf file:

```
spark.driver.extraJavaOptions -Xrs -XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=70 -XX:MaxHeapFreeRatio=70
```
Configuring Temporary Folders When Writing Data to a Hive Table Using SAS Embedded Process

When SAS Embedded Process writes data to a Hive table, it requires a temporary folder for storing output data before it loads the data into Hive.

Add the sas.ep.tempdir property to the mapred-site.xml file to specify this location. The default location is /tmp in the HDFS.

```
<property>
  <name>sas.ep.tempdir</name>
  <value>/tmp</value>
</property>
```

**Note:** If you are using SAS Embedded Process to write data from Windows to a Hive table in an Amazon S3 cluster, the temporary folder for storing output data must be in the same S3 storage space. You must perform the following actions:

- Either remove the sas.ep.tempdir property from the mapred-site.xml file or ensure that it is set to a location that is not in the S3 storage space.
- Add the sas.ep.s3.job.output.tempdir property to the mapred-site.xml file.

```
<property>
  <name>sas.ep.s3.job.output.tempdir</name>
  <value>s3a://comp_clust/node7/temp</value>
</property>
```

If the sas.ep.s3.job.output.tempdir property is not specified, it defaults to the location that is specified in sas.ep.tempdir.

---

Additional Configuration for Hortonworks 3.1 with Patch HDP 3.1.0.29-2

Perform this task only if you have applied Hortonworks patch HDP 3.1.0.29-2 to your Hortonworks 3.1 cluster.

Perform the following actions, as appropriate:
If you are using SAS Embedded Process for Spark, you must modify the spark-defaults.conf file that is stored in the SAS_HADOOP CONFIG PATH environment variable.

In the spark-defaults.conf file, locate the spark.driver.extraClassPath and spark.executor.extraClassPath properties. Then, prepend the location of the Hadoop configuration folder that is located in the cluster nodes to the existing path of these properties. In this example, the location of the Hadoop configuration folder is /etc/hadoop/conf.

```
spark.driver.extraClassPath /etc/hadoop/conf:
spark.executor.extraClassPath /etc/hadoop/conf:
```

If you are using SAS Embedded Process and running model publishing actions in SAS Viya 3.5, the Hortonworks 3.1.0.29-2 hive-warehouse-connector-assembly*.jar file contains a conflicting core-site.xml. You must prepend the location of the Hadoop configuration files that is specified in SAS_HADOOP_CONFIG_PATH to the classpath option.

---

### Additional Configuration for the SAS Embedded Process for Spark Continuous Session

SAS Embedded Process for Spark continuous session is an instantiation of a long running SAS Embedded Process session in a cluster that is capable of serving one CAS session. You can use the SAS Embedded Process for Spark continuous session and SAS Data Connector to Spark to perform multiple actions within the same CAS session, such as loading and saving a table or running a scoring model.

Additional Spark configuration properties can be set using the PROPERTIES= option in the sparkEmbeddedProcess.startSparkEP action. For more information, see “SAS Embedded Process for Spark Action Set” in SAS Visual Analytics: Programming Guide.
Upgrade the SAS Embedded Process Parcel from a Previous Version

Overview

Upgrade the SAS Embedded Process Parcel

Remove the SAS Embedded Process Parcel

Remove the SAS Embedded Process RPM Package Files from the Cloudera Cluster

Upgrade the SAS Embedded Process Stack from a Previous Version

Overview

Upgrade the SAS Embedded Process Stack

Remove the SAS Embedded Process Stack

Remove the SAS Embedded Process RPM Package Files from the Hortonworks Cluster

Upgrade the Manually Installed SAS Embedded Process from a Previous Version

Upgrade a Previous Hadoop Version – SAS 9.4M6 Release or Later

Upgrade a Previous Hadoop Version – July 2015 Release of SAS 9.4 or SAS Viya 3.4 Prior to November 2018

Upgrade a Previous Hadoop Version – SAS 9.4 Prior to the July 2015 Release of SAS 9.4

Upgrade a Previous Hadoop Version – SAS 9.3

Uninstall SAS Embedded Process for Hadoop (Manual)

Hadoop SAS Embedded Process Versions

Upgrade the SAS Embedded Process Parcel from a Previous Version

Overview

The version number of the parcel is calculated by the sas-ep-hadoop-media.RPM file with the actual version of the installed product that you selected to deploy. You cannot deploy a parcel that has the same version number as a parcel that was previously deployed.
You can either deactivate the existing parcel or remove it before upgrading or re-installing.

- If you want to deactivate the existing parcel, use Cloudera Manager to deactivate the parcel and continue with the installation instructions at Chapter 10, “Deploying SAS Embedded Process Using a Parcel or Stack,” on page 77.
- If you want to remove the existing parcel, follow the instructions in “Remove the SAS Embedded Process Parcel” on page 116.

After you remove or deactivate the parcel, continue with the removal of the RPM files. For more information, see “Remove the SAS Embedded Process RPM Package Files from the Cloudera Cluster” on page 117.

Remove the SAS Embedded Process Parcel

**CAUTION**
(SAS 9.4 only) If you are using SAS Data Loader for Hadoop, you should remove the Quality Knowledge Base (QKB) and the SAS Data Management Accelerator for Spark from the Hadoop nodes before you remove SAS Embedded Process. Removal of SAS Embedded Process implicitly removes the scripts that are needed in order to remove these products. For more information, see Remove SAS Software from the Hadoop Cluster.

To remove SAS Embedded Process Parcel using Cloudera Manager, follow these steps:

1. Start Cloudera Manager.
2. Deactivate the SASEP parcel:
   a. Navigate to the Hosts ➔ Parcels tab.
   b. Select Actions ➔ Deactivate.
      You are asked to restart the cluster.
   c. Click Cancel.
      Note: Restarting the cluster is not required. If you want to restart and a rolling restart is available on your cluster, you can choose to perform a rolling restart instead of a full restart. For instructions about performing a rolling restart, see the Cloudera Manager documentation.
   d. Click OK to continue the deactivation.
3. Remove the SASEP parcel:
   a. Select Activate ➔ Remove from Hosts.
   b. Click OK to confirm.
4. Delete the SASEP parcel.
   5. Click OK to confirm.
      This step deletes the parcel files from the /opt/sasep/ directory.
You must remove the ep-config.xml file. To remove this file:

**CAUTION**

If you do not remove the ep-config.xml file, SAS Embedded Process still appears to be available for use. However, any software that uses SAS Embedded Process fails.

a Log on to HDFS.

```
sudo su - root
su - hdfs | hdfs-userid
```

Note: If your cluster is secured with Kerberos, the HDFS user must have a valid Kerberos ticket to access HDFS. You can obtain a Kerberos ticket with kinit.

b Navigate to the /opt/sasep/home/bin/ directory.

c Remove the ep-config.xml file.

```
./sasep-admin.sh -remove
```

d Remove the RPM files from the cluster.

For more information, see “Remove the SAS Embedded Process RPM Package Files from the Cloudera Cluster” on page 117.

---

Remove the SAS Embedded Process RPM Package Files from the Cloudera Cluster

After the parcel has been removed using Cloudera Manager, you can remove the RPM files.

**Note:** Remove the SAS Embedded Process RPM package files if you no longer need them. If you plan to re-install the same SAS Embedded Process RPM package files, you do not have to remove them.

1 Log on to the Hadoop cluster as a user with root privileges.

2 Navigate to the directory where the Hadoop RPM files are located. The RPMTempDir directory was created when you installed SAS Embedded Process.

```
cd /RPMTempDir/
```

For more information, see "Install the Hadoop RPM Files" on page 66.

3 Enter this command to find out what is currently installed on the cluster.

```
rpm -qa 'sas-*'
```

4 Enter these commands to remove all the RPM files.

```
rpm -e sas-hadoop-media
```
rpm -e sas-sepcorehdp
rpm -e sas-dqacchadp
rpm -e sas-hadoopmrep

**CAUTION**

You must remove all four RPM files in the specified order. Otherwise, the installation of the new RPM files and SAS Embedded Process fails.

---

5 Continue with the installation steps in Chapter 10, “Deploying SAS Embedded Process Using a Parcel or Stack,” on page 77.

---

## Upgrade the SAS Embedded Process Stack from a Previous Version

### Overview

The version number of the stack is calculated by the sas-ep-hadoop-media.RPM file with the actual version of the installed product that you selected to deploy. You cannot deploy a stack that has the same version number as a stack that was previously deployed.

You can either deactivate the existing stack or remove it before upgrading or re-installing.

- If you want to deactivate the existing stack, use Ambari to deactivate the stack.
- If you want to remove the existing stack, follow the instructions in “Remove the SAS Embedded Process Stack” on page 118.

After you remove or deactivate the stack, continue with the removal of the RPM files. For more information, see “Remove the SAS Embedded Process RPM Package Files from the Hortonworks Cluster” on page 120.

---

## Remove the SAS Embedded Process Stack

**CAUTION**

(SAS 9.4 only) If you are using SAS Data Loader for Hadoop, you should remove the Quality Knowledge Base (QKB) and the SAS Data Management Accelerator for Spark from the Hadoop nodes before you remove SAS Embedded Process. Removal of SAS Embedded Process implicitly removes the scripts that are used in order to remove these products. For more information, see Remove SAS Software from the Hadoop Cluster.
Note: You must have root or passwordless sudo access to remove the stack.

To remove the SAS Embedded Process stack using Ambari, follow these steps:

1. Navigate to the /opt/sas/ep/home/media/bin/ directory on the client where the SAS software is downloaded and installed.
   ```
   cd /opt/sas/ep/home/media/bin/
   ```
   The delete_stack.sh file should be in this directory.

2. If the /opt/sas/ep/home/media/bin directory does not exist on the node where Ambari is running, then copy the delete_stack.sh file to a temporary directory where the cluster manager server is located. Here is an example of using a secure copy:
   ```
   scp delete_stack.sh user@cluster-manager-host:/mydir
   ```

3. Use this command to run the delete script.
   ```
   ./delete_stack.sh <Ambari-Admin-User-Name>
   ```

4. Enter the Ambari administrator password at the prompt.

5. You are prompted to restart the Ambari server in order to complete the removal of the SASEP service. To remove SAS Embedded Process, you must restart the Ambari server.

6. Enter y to restart the Ambari server. The SASEP service is no longer listed on the Ambari dashboard user interface.

7. You must remove the ep-config.xml file. To remove this file:
   
   **CAUTION**
   If you do not remove the ep-config.xml file, SAS Embedded Process still appears to be available for use. However, any software that uses SAS Embedded Process fails.

   a. Log on to HDFS.
      ```
      sudo su - root
      su - hdfs | hdfs-userid
      ```
      Note: If your cluster is secured with Kerberos, the HDFS user must have a valid Kerberos ticket to access HDFS. You can obtain a Kerberos ticket with kinit.

   b. Navigate to the /opt/sasep/home/bin/ directory.

   c. Remove the ep-config.xml file.
      ```
      ./sasep-admin.sh -remove
      ```

8. Remove the RPM files from the cluster.
   For more information, see "Remove the SAS Embedded Process RPM Package Files from the Hortonworks Cluster" on page 120.
Remove the SAS Embedded Process RPM Package Files from the Hortonworks Cluster

After the stack has been removed using Ambari, you can remove the installed RPM files.

**Note:** Remove the SAS Embedded Process RPM package files if you no longer need them. If you plan to re-install the same SAS Embedded Process RPM package files, you do not have to remove them.

---

1. Log on to the Hadoop cluster as a user with root privileges.

2. Navigate to the directory where the Hadoop RPM files are located. The `RPMTempDir` directory was created when you installed SAS Embedded Process.

   ```
   cd /RPMTempDir/
   ```

   For more information, see "Install the Hadoop RPM Files" on page 66.

3. Enter this command to find out what is currently installed on the cluster.

   ```
   rpm -qa 'sas-*'
   ```

4. Enter these commands to remove all the installed RPM files.

   ```
   rpm -e sas-hadoop-media
   rpm -e sas-sepcorehadp
   rpm -e sas-dqacchadp
   rpm -e sas-hadoopmrep
   ```

**CAUTION**
You must remove all four RPM files in the specified order. Otherwise, the installation of the new RPM files and SAS Embedded Process fails.

---

5. Continue with the installation steps in Chapter 10, "Deploying SAS Embedded Process Using a Parcel or Stack," on page 77.
Upgrade the Manually Installed SAS Embedded Process from a Previous Version

Upgrade a Previous Hadoop Version – SAS 9.4M6 Release or Later

To find out when your version of SAS Embedded Process was released, see “Hadoop SAS Embedded Process Versions” on page 125.

To upgrade or re-install from SAS 9.4M6 or a later release, follow these steps:

1. Log on to the Hadoop cluster as a user with root privileges.

2. Navigate to the directory where the Hadoop RPM files are located. The RPMTempDir directory was created when you installed SAS Embedded Process.

   ```
   cd /RPMTempDir/
   ```

   For more information, see “Install the Hadoop RPM Files” on page 66.

3. Enter this command to find out what is currently installed on the cluster.

   ```
   rpm -qa '*sas-**'
   ```

4. Enter these commands to remove all the RPM files.

   ```
   rpm -e sas-hadoop-media-
   rpm -e sas-sepcorehadp
   rpm -e sas-dqacchadp
   rpm -e sas-hadoopmrep
   ```

   Note: Remove the SAS Embedded Process RPM package files if you no longer need them. If you plan to re-install the same SAS Embedded Process RPM package files, you do not have to remove them.

   CAUTION

   You must remove all four RPM files in the specified order. Otherwise, the installation of the new RPM files and SAS Embedded Process fails.

Upgrade a Previous Hadoop Version – July 2015 Release of SAS 9.4 or SAS Viya 3.4 Prior to November 2018

**CAUTION**

(SAS 9.4 only) If you are using SAS Data Loader for Hadoop, you should remove the Quality Knowledge Base (QKB) and the SAS Data Management Accelerator for Spark from the Hadoop nodes before you remove SAS Embedded Process. Removal of SAS Embedded Process implicitly removes the scripts that are used in order to remove these products. For more information, see Remove SAS Software from the Hadoop Cluster.

To find out when your version of SAS Embedded Process was released, see “Hadoop SAS Embedded Process Versions” on page 125.

To upgrade or re-install from the July 2015 release of SAS 9.4 or SAS Viya 3.4 prior to November 2018, follow these steps:

1. Locate the sasep-admin.sh file.
   
   This file is in the `EPInstallDir/SASEPHome/bin` directory. The `EPInstallDir` directory is where you installed SAS Embedded Process.
   
   One way to find out the location of the `EPInstallDir` directory is to look at the `sas.ep.classpath` property in the `ep-config.xml` file. The `ep-config.xml` file is located on the HDFS in the `/sas/ep/config/` directory.
   
   a. Enter this Hadoop command to read the `ep-config.xml` file on HDFS.
      
      ```bash
      hadoop fs -cat /sas/ep/config/ep-config.xml
      ```
   
   b. Search for the `sas.ep.classpath` property.
   
   c. Take note of the directory path. You will need it for the next step.
      
      The path should be `EPInstallDir/SASEPHome/` where `EPInstallDir` is where you installed SAS Embedded Process.
   
   d. Navigate to the `EPInstallDir/SASEPHome/bin` directory.

2. Run the sasep-admin.sh -remove script.
   
   This script removes SAS Embedded Process from the data nodes.

3. Run this command to remove the SASEPHome directories from the master node.
   
   ```bash
   rm -rf SASEPHome
   ```

Upgrade a Previous Hadoop Version – SAS 9.4 Prior to the July 2015 Release of SAS 9.4

CAUTION
(SAS 9.4 only) If you are using SAS Data Loader for Hadoop, you should remove the Quality Knowledge Base (QKB) and the SAS Data Management Accelerator for Spark from the Hadoop nodes before you remove SAS Embedded Process. Removal of SAS Embedded Process implicitly removes the scripts that are used in order to remove these products. For more information, see Remove SAS Software from the Hadoop Cluster.

To find out when your version of SAS Embedded Process was released, see “Hadoop SAS Embedded Process Versions” on page 125.

To upgrade or re-install from a version of SAS 9.4 before the July 2015 release of SAS 9.4, follow these steps:


   ```bash
   EPInstallDir/SAS/SASTKInDatabaseServerForHadoop/9.*/bin/sasep-servers.sh
   -stop -hostfile host-list-filename | -host ""host-list""
   ``

   The `EPInstallDir` directory is on the master node where you installed SAS Embedded Process.

   For more information, see the SASEP-SERVERS.SH syntax section of the SAS In-Database Products: Administrator’s Guide that came with your release.

2. Remove SAS Embedded Process from all the nodes.

   ```bash
   EPInstallDir/SAS/SASTKInDatabaseForServerHadoop/9.*/bin/sasep-servers.sh
   -remove -hostfile host-list-filename | -host ""host-list"
   -mrhome dir
   ``

   Note: This step ensures that all old SAS Hadoop MapReduce JAR files are removed.

   For more information, see the SASEP-SERVERS.SH syntax section of the SAS In-Database Products: Administrator’s Guide that came with your release.

3. Restart the MapReduce service to clear the SAS Hadoop MapReduce JAR files from the cache.

4. Verify that all files that are associated with SAS Embedded Process have been removed.

   Note: If all the files have not been deleted, then you must manually delete them. Open-source utilities are available that can be used to delete these files across multiple nodes.

   a. Verify that the `sas.hadoop.ep.apache*.jar` files have been deleted.

   The JAR files are located at `HadoopHome/lib`. 
For Cloudera, the JAR files are typically located here:

/opt/cloudera/parcels/CDH/lib/hadoop/lib

For Hortonworks, the JAR files are typically located here:

/usr/lib/hadoop/lib

b Verify that all the SAS Embedded Process directories and files have been deleted on all nodes except the node from which you ran the sasep-servers.sh -remove script. The sasep-servers.sh -remove script removes the file from all nodes except the node from which you ran the script.

c Manually remove the SAS Embedded Process directories and files on the master node (EPInstallDir) from which you ran the script.

The sasep-servers.sh -remove script removes the file from all nodes except the node from which you ran the script. The sasep-servers.sh -remove script displays instructions that are similar to the following:

localhost WARN: Apparently, you are trying to uninstall SAS Embedded Process for Hadoop from the local node.
The binary files located at

local_node/SAS/SASTKInDatabaseServerForHadoop/local_node/SAS/SASACCESSstoHadoopMapReduceJARFiles will not be removed.

localhost WARN: The init script will be removed from /etc/init.d and the SAS Map Reduce JAR files will be removed from /usr/lib/hadoop-mapreduce/lib.

localhost WARN: The binary files located at local_node/SAS should be removed manually.

TIP You can use this command to find the location of any instance of SAS Embedded Process:

ps -ef | grep depserver


Upgrade a Previous Hadoop Version – SAS 9.3

To find out when your version of SAS Embedded Process was released, see “Hadoop SAS Embedded Process Versions” on page 125.

To upgrade or re-install from SAS 9.3, follow these steps:

1 Stop SAS Embedded Process using the SAS 9.3 sasep-stop.all.sh script.

EPInstallDir/SAS/SASTKInDatabaseServerForHadoop/9.35/bin/sasep-stop.all.sh

The EPInstallDir directory is on the master node where you installed SAS Embedded Process.

2 Delete SAS Embedded Process from all nodes.

EPInstallDir/SAS/SASTKInDatabaseServerForHadoop/9.35/bin/sasep-delete.all.sh

3 Verify that the sas.hadoop.ep.distribution-name.jar files have been deleted.
The JAR files are located at $HadoopHome/lib$.

For Cloudera, the JAR files are typically located here:

$/opt/cloudera/parcels/CDH/lib/hadoop/lib$

For Hortonworks, the JAR files are typically located here:

$/usr/lib/hadoop/lib$

4 Restart the MapReduce service to clear the SAS Hadoop MapReduce JAR files from the cache.


Uninstall SAS Embedded Process for Hadoop (Manual)

The method that you use to uninstall SAS Embedded Process for Hadoop is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the last step, which is about continuing the installation.

- “Upgrade a Previous Hadoop Version – SAS 9.4M6 Release or Later” on page 121
- “Upgrade a Previous Hadoop Version – July 2015 Release of SAS 9.4 or SAS Viya 3.4 Prior to November 2018” on page 122
- “Upgrade a Previous Hadoop Version – SAS 9.3” on page 124

To find out when your version of SAS Embedded Process was released, see “Aster SAS Embedded Process Versions” on page 17.

Hadoop SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Prior to SAS 9.4M6, SAS Viya customers could get only the latest version that was shipped with SAS Viya 3.x. Starting with SAS 9.4M6, both SAS 9.4 and SAS Viya customers can get the latest version of SAS Embedded Process at any time.
### Table 13.1 SAS Embedded Process Version by Release Date

<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2019</td>
<td>SAS Viya 3.5</td>
<td>sas-hadoopmrep-1.7.21-&lt;timestamp&gt;.x86_64.rpm, sas-hadoopmrep-1.7.21-&lt;timestamp&gt;.suse.x86_64.rpm,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-hadoopmrep-1.7.21-&lt;timestamp&gt;.amd.x86_64.deb, sas-dqacchadp-1.2.0-&lt;timestamp&gt;.x86_64.rpm,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-dqacchadp-1.2.0-&lt;timestamp&gt;.suse.x86_64.rpm, sas-dqacchadp-1.2.0-&lt;timestamp&gt;.amd.x86_64.deb,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcorehadp-17.0.0-&lt;timestamp&gt;.x86_64.rpm (Red Hat Enterprise Linux), sas-sepcorehadp-17.0.0-&lt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>timestamp.suse.x86_64.rpm, sas-sepcorehadp-17.0.0-&lt;timestamp&gt;.amd.x86_64.deb (Ubuntu)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-ep-hadoop-media-1.0.5-&lt;timestamp&gt;.x86_64.rpm, sas-ep-hadoop-media-1.0.5-&lt;timestamp&gt;.suse.x86_6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>64.rpm, sas-ep-hadoop-media-1.0.5-&lt;timestamp&gt;.amd.x86_64.deb</td>
</tr>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-hadoopmrep-1.7.0-&lt;timestamp&gt;.x86_64.rpm, sas-hadoopmrep-1.7.0-&lt;timestamp&gt;.suse.x86_64.rpm,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-hadoopmrep-1.7.0-&lt;timestamp&gt;.amd.x86_64.deb, sas-dqacchadp-1.1.1-&lt;timestamp&gt;.x86_64.rpm,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-dqacchadp-1.1.1-&lt;timestamp&gt;.suse.x86_64.rpm, sas-dqacchadp-1.1.1-&lt;timestamp&gt;.amd.x86_64.deb,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcorehadp-16.0.0-&lt;timestamp&gt;.x86_64.rpm (Red Hat Enterprise Linux), sas-sepcorehadp-16.0.0-&lt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>timestamp.suse.x86_64.rpm, sas-sepcorehadp-16.0.0-&lt;timestamp&gt;.amd.x86_64.deb (Ubuntu)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-ep-hadoop-media-1.0.2-&lt;timestamp&gt;.x86_64.rpm, sas-ep-hadoop-media-1.0.2-&lt;timestamp&gt;.suse.x86_6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>64.rpm, sas-ep-hadoop-media-1.0.2-&lt;timestamp&gt;.amd.x86_64.deb</td>
</tr>
<tr>
<td>July 2018</td>
<td>SAS Viya 3.4</td>
<td>sepcorehadp-15.00000-1.sh</td>
</tr>
<tr>
<td>December 2017</td>
<td>SAS Viya 3.3</td>
<td>sepcorehadp-14.00000-1.sh</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcorehadp-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcorehadp-12.00000-1.sh</td>
</tr>
<tr>
<td>Date</td>
<td>SAS Release</td>
<td>SAS Embedded Process File Name</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
Where to Go from Here

Additional Configuration for Other Software

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Code Accelerator for Hadoop</td>
<td>“Hadoop Administrator’s Guide” in SAS In-Database Products: Administrator’s Guide</td>
</tr>
<tr>
<td>SAS Scoring Accelerator for Hadoop</td>
<td>“Hadoop Administrator’s Guide” in SAS In-Database Products: Administrator’s Guide</td>
</tr>
<tr>
<td>SAS Data Loader for Hadoop</td>
<td>SAS Data Loader for Hadoop: Installation and Configuration Guide</td>
</tr>
<tr>
<td>In-Database Technologies for Hadoop (on SAS Viya) (includes the SAS Data Connect Accelerator for Hadoop and Spark)</td>
<td>SAS Viya for Linux: Deployment Guide</td>
</tr>
<tr>
<td>SAS Contextual Analysis In-Database Scoring in Hadoop</td>
<td>SAS Contextual Analysis In-Database Scoring in Hadoop: Administrator’s Guide</td>
</tr>
<tr>
<td>SAS High-Performance Analytics</td>
<td>SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide</td>
</tr>
</tbody>
</table>
PART 6

SAS Embedded Process Deployment for Netezza

Chapter 15

Deploying SAS Embedded Process for Netezza ....................... 133

Chapter 16

Upgrading or Uninstalling SAS Embedded Process for Netezza ........ 143
Deploying SAS Embedded Process for Netezza

**Netezza Prerequisites**

The SAS Embedded Process deployment for Netezza requires the following:

- A specific version of the Netezza client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.
- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.
- Obtain the sas_ep cartridge from Netezza. The sas_ep cartridge creates the NZRC database. The NZRC database contains remote controller functions that are required by SAS Embedded Process. The sas_ep cartridge is available on the Netezza website. For access to the sas_ep cartridge, contact your local Netezza representative.
- Approximately 300 MB of disk space is required in the /opt file system on each Netezza node.
- Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
Overview of SAS Embedded Process for Netezza

SAS Embedded Process for Netezza is available with SAS Scoring Accelerator and must be deployed before you can use the in-database processing features of the software.

For more information about using this software, see “SAS Scoring Accelerator for Netezza” in SAS In-Database Products: User’s Guide.

SAS Embedded Process is a SAS server process that runs within Netezza to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that installed on your Netezza system.

Note: To use your software, additional configuration is required. For more information, see “Where to Go from Here” on page 141.

Note: If you are adding additional nodes, the version of the SAS Embedded Process must be identical for existing and new nodes.

Netezza Deployment Steps

To deploy SAS Embedded Process for Netezza, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or re-installing) a previous release in SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 22, “Upgrading or Uninstalling SAS Embedded Process for Teradata,” on page 197</td>
</tr>
<tr>
<td>3</td>
<td>Review the permissions that are needed for in-database processing</td>
<td>“Netezza Permissions” on page 138</td>
</tr>
<tr>
<td>4</td>
<td>If you license SAS Scoring Accelerator for Netezza, perform additional configuration as required.</td>
<td>“Where to Go from Here” on page 141</td>
</tr>
</tbody>
</table>
Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - Linux — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

3. Run the following commands to provide platform and repository information that can be helpful if you need to specify the --platform or --repo options in the mirrormgr mirror command in the next step.

   ```bash
   ./mirrormgr list remote platforms --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   
   ./mirrormgr list remote repos --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   ```

4. Create a mirror repository for the SOE deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

   ```bash
   ./mirrormgr mirror --deployment-data
   path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
   <options>
   ```
Note: SAS Mirror Manager downloads all entitled repositories to the /user-home/sas_repos (Linux) or C:\Users\user-id\sas_repos (Windows) directory by default. You can change the location by using the --path path-to-mirror-repository option in the mirrormgr mirror command. You can also choose to download only one platform by using the --platform platform-name.

Here is an example for a Red Hat Enterprise Linux 6 system. In this example, only the latest version of the Red Hat files are downloaded.

```
./mirrormgr mirror --deployment-data
    path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
      --platform x64-redhat-linux-6 --latest
```

Note: SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

Install the SAS Embedded Process RPM Package File for Netezza

To install the SAS Embedded Process RPM package file, follow these steps:

1. If you have a database named SAS_EP, you should rename it.
   When you install SAS Embedded Process, a database named SAS_EP is created. The creation of the sas_ep database overwrites any existing database that is named sas_ep.

2. Log on to the Netezza database as the root authority.

3. Check to see whether the sas_ep cartridge is already registered.
   ```
   nzcm --registered
   ```

4. Use the following nzcm commands to install and register the sas_ep cartridge:
   ```
   nzcm -i sas_ep
   nzcm -r sas_ep
   ```

   Note: The sas_ep cartridge creates a database named NZRC. The NZRC database contains remote controller functions that are required by SAS Embedded Process. The sas_ep database is available on the Netezza website. For access to the sas_ep cartridge, contact your local Netezza representative.

5. Decide whether you want to install and register the SAS Embedded Process cartridge file at deployment time.
   By default, when you install SAS Embedded Process, the SAS Embedded Process cartridge file is created, installed, and registered. If you do not want to install and register the SAS Embedded Process cartridge file, set the SAS_INSTALL_EP environment variable to NO. For more information, see “SAS_INSTALL_EP Environment Variable” on page 217.
6 Locate the SAS Embedded Process file, `sas-sepcorenetz-version-date.timestamp.x86_64.rpm`.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file. For more information, see “Netezza SAS Embedded Process Versions” on page 145.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time at which the RPM package was created.</td>
</tr>
</tbody>
</table>

This file is located in one of the following directories depending on your operating system:

- **Linux** — `/path-to-mirror-repository/repos/shipped/sepnetz/100.0/sepnetz-100.0.0<-option-if-specified>/Packages/s/`
- **Windows** — `C:\path-to-mirror-repository\repos\shipped\sepnetz\100.0\sepnetz-100.0.0<-option-if-specified>\Packages\s`

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 135. Here is an example:</td>
</tr>
<tr>
<td></td>
<td>/mysas/sas_repos/</td>
</tr>
<tr>
<td>&lt;-option-if-specified&gt;</td>
<td>If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in &lt;-option-if-specified&gt;. For example, if you specified --latest in your mirrormgr mirror command, the directory would be -latest.</td>
</tr>
</tbody>
</table>

7 Copy the `sas-sepcorenetz-version-date.timestamp.x86_64.rpm` to a location on the Netezza server. Make sure that you copy the file to the server machine according to the procedures that are used at your site. Here is an example of a secure copy command:

```bash
scp root@netzmach1:/u/nzid/tmp/sas-sepcorenetz-16.0.0-20180821.18223344532.x86_64.rpm /RPMTempDir
```
8 Navigate to the directory where the RPM file was transferred in Step 3.
   cd /RPMTempDir

9 Install the RPM file.
   rpm -ivh sas-sepcorenetz-version-date.timestamp.x86_64.rpm

   Note: When you install the RPM, a Netezza cartridge, SASTKInDatabaseServerForNetezza-version.nzc is created, installed, and registered in the /opt/sas/ep/home directory. The version number version is the same as that used by the RPM file.

10 If you chose not to install and register the SAS Embedded Process cartridge file in Step 3, use the following nzcm commands to install and register SAS Embedded Process Step 5 on page 136:
   nzcm -i /opt/sas/ep/home/SASTKInDatabaseServerForNetezza-version.nzc
   nzcm -r /opt/sas/ep/home/SASTKInDatabaseServerForNetezza
   Run this command to verify that the SAS Embedded Process cartridge file is installed.
   nzcm --registered

   For more NZCM commands, see “NZCM Commands for the SAS Embedded Process” in SAS In-Database Products: Administrator’s Guide.

---

Netezza Permissions

There are three sets of permissions involved with the in-database software.

- The first set of permissions is needed by the person who publishes the SAS formats library and the SAS_COMPILEUDF, SAS_DIRECTORYUDF, and SAS_HEXTOTEXTUDF functions. These permissions must be granted before the %INDNZ_PUBLISH_JAZLIB and %INDNZ_PUBLISH_COMPILEUDF macros are run. Without these permissions, running these macros fails.

   The following table summarizes the permissions that are needed by the person who publishes the formats library and the functions.
Table 15.3 Permissions Needed to Publish the SAS Formats Library and Functions

<table>
<thead>
<tr>
<th>Permission Needed</th>
<th>Authority Required to Grant Permission</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE LIBRARY permission to run the %INDNZ_PUBLISH_JAZLIB macro that publishes the SAS formats library (sas_jazlib object)</td>
<td>System Administrator or Database Administrator</td>
<td>GRANT CREATE LIBRARY TO fmtlib-publisher-id</td>
</tr>
<tr>
<td>CREATE FUNCTION permission to run the %INDNZ_PUBLISH_COMPILEUDF macro that publishes the SAS_COMPILEUDF, SAS_DIRECTORYUDF, and the SAS_HEXTOTEXTUDF functions</td>
<td></td>
<td>GRANT CREATE FUNCTION TO compileudf-publisher-id</td>
</tr>
</tbody>
</table>

The second set of permissions is needed by the person who runs the format publishing macro, %INDNZ_PUBLISH_FORMATS, or the scoring publishing macro, %INDNZ_PUBLISH_MODEL. The person who runs these macros is not necessarily the same person who runs the %INDNZ_PUBLISH_JAZLIB and %INDNZ_PUBLISH_COMPILEUDF macros. These permissions are most likely needed by the format publishing or scoring model developer. Without these permissions, the publishing of the scoring model functions and the SAS_PUT( ) function and formats fails.

Note: Permissions must be granted for every format and scoring model publisher and for each database that the format and scoring model publishing uses. Therefore, you might need to grant these permissions multiple times. After the Netezza permissions are set appropriately, the format and scoring publishing macros can be run.

Note: When permissions are granted to specific functions, the correct signature, including the sizes for numeric and string data types, must be specified.

The following table summarizes the permissions that are needed by the person who runs the format or scoring publishing macro.
Table 15.4  Permissions Needed to Run the Format or Scoring Publishing Macro

<table>
<thead>
<tr>
<th>Permission Needed</th>
<th>Authority Required to Grant Permission</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXECUTE permission for the SAS Formats Library</td>
<td>System Administrator or Database Administrator</td>
<td>GRANT EXECUTE ON SAS_JAZLIB TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>EXECUTE permission for the SAS_COMPILEUDF function</td>
<td>Note: If you have SYSADM or DBADM authority, then you have these permissions. Otherwise, contact your database administrator to obtain these permissions.</td>
<td>GRANT EXECUTE ON SAS_COMPILEUDF TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>EXECUTE permission for the SAS_DIRECTORYUDF function</td>
<td></td>
<td>GRANT EXECUTE ON SAS_DIRECTORYUDF TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>EXECUTE permission for the SAS_HEXTOTEXTUDF function</td>
<td></td>
<td>GRANT EXECUTE ON SAS_HEXTOTEXTUDF TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td>CREATE FUNCTION, CREATE TABLE, CREATE TEMP TABLE, and CREATE EXTERNAL TABLE permissions to run the format and scoring publishing macros</td>
<td></td>
<td>GRANT CREATE FUNCTION TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GRANT CREATE TABLE TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GRANT CREATE TEMP TABLE TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GRANT CREATE EXTERNAL TABLE TO scoring-or-fmt-publisher-id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GRANT UNFENCED TO scoring-or-fmt-publisher-id</td>
</tr>
</tbody>
</table>

- The third set of permissions is needed by the person who runs the SAS Embedded Process to create scoring files.

  The SAS Embedded Process has a dependency on the IBM Netezza Analytics (INZA) utility. You must grant the user and database permissions using these commands.

  /nz/export/ae/utlities/bin/create_inza_db_user.sh user-name database-name
  /nz/export/ae/utilities/bin/create_inza_db.sh database-name

  Note: If you plan to use SAS Model Manager with the SAS Scoring Accelerator for in-database scoring, additional permissions are required. For more information, see “Configuring SAS Model Manager” in SAS In-Database Products: Administrator’s Guide.
Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
</table>
Upgrading or Uninstalling SAS Embedded Process for Netezza

Upgrade from a Previous Netezza Version – SAS 9.4M6 and Later .................. 143
Upgrade from a Previous Netezza Version – Prior to SAS 9.4M6 .................. 144
Uninstall SAS Embedded Process for Netezza ........................................ 144
Netezza SAS Embedded Process Versions ............................................. 145

Upgrade from a Previous Netezza Version – SAS 9.4M6 and Later

To upgrade from a previous version of SAS Embedded Process, follow these steps:

1. Log on to the Netezza database as the user who owns the Netezza appliance (usually the nz ID).

2. Check the currently installed version of SAS Embedded Process.
   nzcm --installed

3. Enter these commands to unregister and uninstall SAS Embedded Process.
   nzcm -u SASTKInDatabaseServerForNetezza
   nzcm -e SASTKInDatabaseServerForNetezza

4. Navigate to the /nz/extensions/SASTKInDatabaseServerForNetezza directory and verify that the directory is empty.

Note: Under the SAS directory, the installer for the SAS Formats Library and binary files and the SAS Embedded Process installer both create a directory under the SAS directory. These directories are named SASFormatsLibraryForNetezza and SASTKInDatabaseServerForNetezza, respectively. If you delete everything under the SAS directory, SAS Embedded Process, the SAS Formats Library, and the binary files will be removed automatically. If you want to remove only one directory, then you must explicitly delete only that directory. For example, if you want to remove only the formats library directory, delete the /SAS/SASFormatsLibraryForNetezza directory. The /SAS/SASTKInDatabaseServerForNetezza directory remains intact.
5 Log on to the Netezza database as a user with root privileges.
6 Delete the RPM file.
   `rpm -e sas-sepcorenetz`
7 Continue with the installation steps in “Deploying SAS Embedded Process for Netezza” on page 135.

---

**Upgrade from a Previous Netezza Version – Prior to SAS 9.4M6**

To upgrade from a previous version of SAS Embedded Process, follow these steps:

1 Check the currently installed version of SAS Embedded Process.
   `nzcm --installed`
2 Enter these commands to unregister and uninstall SAS Embedded Process.
   `nzcm -u SASTKInDatabaseServerForNetezza`
   `nzcm -e SASTKInDatabaseServerForNetezza`
3 Navigate to the `/nz/extensions/SASTKInDatabaseServerForNetezza` directory and verify that the directory is empty.

**Note:** Under the SAS directory, the installer for the SAS Formats Library and binary files and the SAS Embedded Process installer both create a directory under the SAS directory. These directories are named `SASFormatsLibraryForNetezza` and `SASTKInDatabaseServerForNetezza`, respectively. If you delete everything under the SAS directory, SAS Embedded Process, the SAS Formats Library, and the binary files are removed. If you want to remove only one, then you must explicitly delete only that directory. For example, if you want to remove only the formats library directory, delete the `/SAS/SASFormatsLibraryForNetezza` directory. The `/SAS/SASTKInDatabaseServerForNetezza` directory remains intact.

4 Continue with the installation steps in “Deploying SAS Embedded Process for Netezza” on page 135.

---

**Uninstall SAS Embedded Process for Netezza**

The method that you use to uninstall SAS Embedded Process for Netezza is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the final step, which is about continuing the installation.
Netezza SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Starting with SAS 9.4M6, SAS 9.4 customers can get the latest version of SAS Embedded Process at any time.

Table 16.1 SAS Embedded Process Version by Release Date

<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcorenetz-16.0.0-&lt;timestamp&gt;.x86_64.rpm</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcorenetz-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcorenetz-12.00000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
PART 7

SAS Embedded Process Deployment for Oracle

Chapter 17
Deploying SAS Embedded Process for Oracle ........................................ 149

Chapter 18
Upgrading or Uninstalling SAS Embedded Process for Oracle ............ 159
Oracle Prerequisites

The SAS Embedded Process deployment for Oracle requires the following:

- A specific version of the Oracle client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.

- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.

- Approximately 300 MB of disk space is required in the /opt file system on each Oracle node.

- Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
Overview of SAS Embedded Process for Oracle

SAS Embedded Process for Oracle is available with the following SAS software products and must be deployed before you can use the in-database processing features of that software:

- **SAS Scoring Accelerator for Oracle**
  For information about using this software, see *SAS In-Database Products: User’s Guide*.

- **SAS High-Performance Analytics**
  For information about using this software, see *SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide*.

SAS Embedded Process is a SAS server process that runs within Oracle to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that is installed on your Oracle system.

**Note:** To use your software, additional configuration is required. For more information, see "Where to Go from Here" on page 156.

**Note:** If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.

Oracle Deployment Steps

To deploy SAS Embedded Process for Oracle, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or re-installing) a previous release in SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 18, “Upgrading or Uninstalling SAS Embedded Process for Oracle,” on page 159</td>
</tr>
<tr>
<td>3</td>
<td>Create Users and Objects for SAS Embedded Process.</td>
<td>“Creating Users and Objects for the SAS Embedded Process” on page 155</td>
</tr>
</tbody>
</table>
## Deploying SAS Embedded Process for Oracle

### Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - **Linux** — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.
   - **Windows** — See Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

   **TIP** For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you download it from Internet Explorer. For Windows, the file has a .zip extension.

   Here is an example of a Linux command to unpack the SAS Mirror Manager:
   ```bash
   tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz
   ```

3. Run the following commands to provide platform and repository information that can be helpful if you need to specify the --platform or --repo options in the mirrormgr mirror command in the next step.
Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

```
./mirrormgr mirror --deployment-data
path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
```

**Note:** SAS Mirror Manager downloads all entitled repositories to the /user-home/sas_repos (Linux) or C:\Users\user-id\sas_repos (Windows) directory by default. You can change the location by using the --path path-to-mirror-repository option in the mirrormgr mirror command. You can also choose to download only one platform by using the --platform platform-name.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.

```
./mirrormgr mirror --deployment-data
path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
--platform x64-suse-linux-12 --latest
```

**Note:** SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

---

**Install the SAS Embedded Process RPM Package File for Oracle**

To install the SAS Embedded Process RPM package file, follow these steps:

1. Log on as a user with root privileges.
2. Locate the SAS Embedded Process RPM package file, sas-sepcoreorcl-version-date.timestamp<.rpm-os-designation>.x86_64.rpm.

**Table 17.1 File Name Options for the RPM File**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “Oracle SAS Embedded Process Versions” on page 160.</td>
</tr>
</tbody>
</table>
### Table 17.2 Path Options for the RPM File

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>path-to-mirror-repository</code></td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 135. Here is an example: /mysas/sas_repos/</td>
</tr>
</tbody>
</table>
| `repo-os-designation<option-if-specified>` | `repo-os-designation` identifies the operating system. Available values depend on your data source and your software order. Possible values are as follows:  
  - x64_redhat_linux_6-yum  
  - x64_suse_linux_12-yum  

If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in `option-if-specified`. For example, if you specified `--latest` in your `mirrormgr mirror` command, the directory would be `x64_suse_linux_12-yum-latest`. |

3. Copy the `sas-sepcoreorcl-version-date.timestamp<.rpm-os-designation>.x86_64.rpm` to a location on the Oracle server.
Make sure that you copy the file to the server machine according to the procedures that are used at your site. Here is an example of a secure copy command:

```bash
scp sas-sepcoreorcl-16.0.0-20180821.18223344532.suse.x86_64.rpm
   root@orclmach1:/RPMTempDir
```

**Note:** The `RPMTempDir` directory must be at a location where it can be accessed from at least one of the Oracle nodes.

4. Navigate to the directory where the RPM file was transferred in Step 2.

```bash
cd /RPMTempDir
```

5. Install the RPM file on every node.

```bash
rpm -ivh sas-sepcoreorcl-version-date.timestamp
   <.rpm-os-designation>.x86_64.rpm
```

After the RPM file is installed, an `/opt/sas/ep/home/` directory is created where the SAS Embedded Process files are installed. The content of the target directories should be similar to the following. Part of the directory path is highlighted to emphasize the different target directories that are used.

```
/opt/sas/ep/home/admin
/opt/sas/ep/home/bin
/opt/sas/ep/home/logs
/opt/sas/ep/home/misc
/opt/sas/ep/home/sasexe
/opt/sas/ep/home/utilities
```

6. On non-shared Oracle home systems, update the contents of the `$ORACLE_HOME/hs/admin/extproc.ora` file on each node. On shared Oracle home systems, you can update the file in one location that is accessible by all nodes.

**Note:** Ask your database administrator whether the ORACLE_HOME environment variable is set.

   a. Log on as the Oracle user.
   b. Make a backup of the current extproc.ora file.
   c. Add the following settings to the extproc.ora file. If these variables have been previously set, make sure that their values are overridden with the following values.

```bash
SET EXTPROC_DLLS=ANY
SET BPPATH=/opt/sas/ep/home
SET TKPATH=/opt/sas/ep/home/sasexe
```

**Note:** This step must be performed only for the first installation of SAS Embedded Process.

7. On non-shared Oracle home systems, update the contents of the `$ORACLE_HOME/network/admin/sqlnet.ora` file on each node.
a Make a backup of the current sqlnet.ora file. If the file does not exist, create one.

b Add the following setting to the sqlnet.ora file.

DIAG_ADR_ENABLED=OFF

Creating Users and Objects for the SAS Embedded Process

After the SAS Embedded Process for Oracle is installed, the DBA must create the users and grant user privileges. The DBA needs to perform these tasks before the SAS administrator can create the objects for the Oracle server. The users and objects are required for the SAS Embedded Process to work.

Note: SQLPLUS or an equivalent SQL tool can be used to submit the SQL statements in this topic.

1 Create a SASADMIN user.

To create the user accounts for Oracle, the DBA must perform the following steps:

a Change the directory to /opt/sas/ep/home/admin.

b Connect as SYS, using the following command:

sqlplus sys/<password> as sysdba

c Create and grant user privileges for the SASADMIN user.

Here is an example of how to create a SASADMIN user.

CREATE USER SASADMIN IDENTIFIED BY <password>
    DEFAULT TABLESPACE <tablespace-name>
    TEMPORARY TABLESPACE <tablespace-name>;
    GRANT UNLIMITED TABLESPACE TO SASADMIN;

d Submit the following SQL script to grant the required privileges to the SASADMIN user.

SQL>@sasadmin_grant_privs.sql

e If the script does not automatically log you off after it is complete, log off from the SQLPLUS session using "Quit" or close your SQL tool.

2 Create the necessary database objects.

To create the objects and the SASEPFUNC table function that are needed to run the scoring model, the SAS administrator (SASADMIN) must perform the following steps:

a Change the current directory to /opt/sas/ep/home/admin (if you are not already there).
b  Connect as SASADMIN, using the following command:

   sqlplus sasadmin/<password>

c  Submit the following SQL statement:

   @create_sasepfunc.sql;

Note: You can ignore the following errors:

ORA-00942: table or view does not exist
ORA-01432: public synonym to be dropped does not exist

Oracle Permissions

The person who runs the %INDOR_CREATE_MODELTABLE needs CREATE permission to create the model table. Here is an example.

GRANT CREATE TABLE TO userid

The person who runs the %INDOR_PUBLISH_MODEL macro needs INSERT permission to load data into the model table. This permission must be granted after the model table is created. Here is an example.

GRANT INSERT ON model-table-name TO user-id

Note: The RESOURCE user privilege that was granted in the previous topic includes the permissions for CREATE, DELETE, DROP, and INSERT.

If you plan to use SAS Model Manager with the SAS Scoring Accelerator for in-database scoring, additional permissions are required. For more information, see “Configuring SAS Model Manager” in SAS In-Database Products: Administrator’s Guide.

Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software</td>
<td>Where to Go for Information</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>SAS High-Performance Analytics</td>
<td>SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide</td>
</tr>
</tbody>
</table>
Upgrade from a Previous Oracle Version – SAS 9.4M6 and Later

To find out when your version of SAS Embedded Process was released, see “Oracle SAS Embedded Process Versions” on page 160.

You can upgrade from a previous version of SAS Embedded Process. Before installing SAS Embedded Process, consult with the database administrator (DBA) about notifying the user community that SAS Embedded Process will be upgraded. The DBA should then alter the availability of the database by restricting access or by bringing the database down.

To upgrade from a version of SAS Embedded Process SAS 9.4M6 or a later release, follow these steps:

1. Log on to the Oracle system as root.
   
   You can use `su` or `sudo` to become the root authority.

2. Enter this command to remove the RPM file.

   ```
   rpm -e sas-sepcoreorcl
   ```

   You can verify that the RPM file was removed with the following command.

   ```
   rpm -qa sas-sepcoreorcl
   ```

3. Continue with the installation steps in “Oracle Deployment Steps” on page 150.
Upgrade from a Previous Oracle Version – Prior to SAS 9.4M6

To find out when your version of SAS Embedded Process was released, see “Oracle SAS Embedded Process Versions” on page 160.

You can upgrade from a previous version of SAS Embedded Process. Before installing SAS Embedded Process for Oracle, consult with the database administrator about notifying the user community that SAS Embedded Process will be upgraded. The DBA should then alter the availability of the database by restricting access or by bringing the database down. Continue with the installation steps in “Oracle Deployment Steps” on page 150.

Uninstall SAS Embedded Process for Oracle

The method that you use to uninstall SAS Embedded Process for Oracle is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the final step, which is about continuing the installation.

- “Upgrade from a Previous Oracle Version – SAS 9.4M6 and Later” on page 159
- “Upgrade from a Previous Oracle Version – Prior to SAS 9.4M6” on page 160

To find out when your version of SAS Embedded Process was released, see “Oracle SAS Embedded Process Versions” on page 160.

Oracle SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Starting with SAS 9.4M6, SAS 9.4 customers can get the latest version of SAS Embedded Process at any time.
<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcoreorcl-16.0.0-date.timestamp.x86_64.rpm (Red Hat Enterprise Linux)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcoreorcl-16.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcoreorcl-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcoreorcl-12.00000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
Deploying SAS Embedded Process for SAP HANA

SAP HANA Prerequisites

The SAS Embedded Process deployment for SAP HANA requires the following:

- A specific version of the SAP HANA client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.
- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.
- Approximately 300 MB of disk space is required in the /opt file system on each SAP HANA node.
Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.

Overview of SAS Embedded Process for SAP HANA

SAS Embedded Process for SAP HANA is available with the following SAS software products and must be deployed before you can use the in-database processing features of that software:

- SAS Scoring Accelerator for SAP HANA
  For more information about using this software, see “SAP HANA Administrator’s Guide” in SAS In-Database Products: Administrator’s Guide.

- SAS High-Performance Analytics
  For information about using this software, see SAS High-Performance Analytics Server: User’s Guide.

SAS Embedded Process is a SAS server process that runs within SAP HANA to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that is installed on your SAP HANA system.

Note: To use your software, some additional configuration is required. For more information, see “Where to Go from Here” on page 179.

Note: If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.

SAP HANA Deployment Steps

To deploy SAS Embedded Process for SAP HANA, follow these steps:

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or re-installing) a previous release of SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 20, “Upgrading or Uninstalling SAS Embedded Process for SAP HANA,” on page 181</td>
</tr>
</tbody>
</table>
### Deploying SAS Embedded Process for SAP HANA

#### Create a Mirror Repository

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.

2. Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:
   - Linux — See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

---

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Install the SASLINK AFL Plugins and make sure that the Auxiliary Wrapper Procedures are available.</td>
<td>“SASLINK AFL Plug-ins and Auxiliary Wrapper Procedures” on page 174</td>
</tr>
<tr>
<td>5</td>
<td>Review the semaphore requirements.</td>
<td>“Semaphore Requirements When Using the SAS Embedded Process for SAP HANA” on page 177</td>
</tr>
<tr>
<td>6</td>
<td>Review the permissions that are needed for in-database processing.</td>
<td>“Oracle Permissions” on page 156</td>
</tr>
<tr>
<td>7</td>
<td>If you license the following software, perform additional configuration as required.</td>
<td>“Where to Go from Here” on page 195</td>
</tr>
<tr>
<td></td>
<td>- SAS Scoring Accelerator for SAP HANA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- SAS High-Performance Analytics</td>
<td></td>
</tr>
</tbody>
</table>
Windows — See Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

**TIP** For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you downloaded it from Internet Explorer. For Windows, the file has a .zip extension.

Here is an example of a Linux command to unpack SAS Mirror Manager:
```
tar -xvzf /path-to-mirror-manager-download/mIRRormgr-linux.tgz
```

3 (Optional) Run the following commands to provide platform and repository information that can be helpful if you need to specify the --platform or --repo options in the mIRRormgr mirror command in the next step.
```
./mIRRormgr list remote platforms --deployment-data
  path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip

./mIRRormgr list remote repos --deployment-data
  path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
```

4 Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.
```
./mIRRormgr mirror --deployment-data
  path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip <options>
```

Note: SAS Mirror Manager downloads all entitled repositories to the /user-home/sas_repos (Linux) or C:\Users\user-id\sas_repos (Windows) directory by default. You can change the location by using the --path path-to-mirror-repository option in the mIRRormgr mirror command. You can also choose to download only one platform by using the --platform platform-name.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.
```
./mIRRormgr mirror --deployment-data
  path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
  --platform x64-suse-linux-12 --latest
```

Note: SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.

---

**Install SAS Embedded Process for SAP HANA – Single Node Deployment**

To install the RPM file and SAS Embedded Process, follow these steps:
1. Locate the SAS Embedded Process file, sas-sepcorehana-version-date.timestamp-rpm-os-designation.x86_64.rpm.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “SAP HANA SAS Embedded Process Versions” on page 183.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time at which the repository was created.</td>
</tr>
<tr>
<td>rpm-os-designation</td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For SAP HANA, the permissible value is suse.</td>
</tr>
<tr>
<td></td>
<td>There is no rpm-os-designation value for Red Hat Enterprise Linux.</td>
</tr>
</tbody>
</table>

The RPM file is located in one of the following directories depending on your operating system:

- **Linux** — `/path-to-mirror-repository/repos/shipped/sephana/100.0/sephana-100.0.0-repo-os-designation-<option-if-specified>/Packages/s/`
- **Windows** — `C:\path-to-mirror-repository\repos\shipped\sephana\100.0\sephana-100.0.0-repo-os-designation-<option-if-specified>\Packages\s/`

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 135. Here is an example: <code>/mysas/sas_repos/</code></td>
</tr>
</tbody>
</table>
Copy the sas-sepcorehana-version-date.timestamp-.x86_64.rpm to a location on the SAP HANA server.

Make sure that you copy the file to the server machine according to the procedures that are used at your site.

Here is an example of a secure copy command:

```
scp sas-sepcorehana-16.0.0.20180821.1822334532.suse.x86_64.rpm
root@hanamach1:/RPMTempDir
```

Note: The RPMTempDir directory must be in a location where it can be accessed from at least one of the SAP HANA nodes.

Navigate to the directory where the RPM file was transferred in Step 2.

```
cd /RPMTempDir
```

Install the RPM file.

```
rpm -ivh sas-sepcorehana-version-date.timestamp-.x86_64.rpm
```

After the RPM file is installed, an /opt/sas/ep/home/ directory is created in which the EP files are installed. The contents of the /opt/sas/ep/home/ directory should be similar to the following listing. Directories and files of interest are highlighted.

```
./opt/sas/ep/home/bin
./opt/sas/ep/home/bin/ShowSASEPStatus.sh
./opt/sas/ep/home/bin/ShutdownSASEP.sh
./opt/sas/ep/home/bin/StartupSASEP.sh
./opt/sas/ep/home/misc
./opt/sas/ep/home/misc/SAS_EP_sas.com.tgz
./opt/sas/ep/home/misc/sas_saslink_installer.tgz
./opt/sas/ep/home/sasexe
./opt/sas/ep/home/utilities
```
The ShowSASEPStatus.sh file shows the status of SAS Embedded Process on each instance. The StartupSASEP.sh and ShutdownSASEP.sh files enable you to manually start and stop SAS Embedded Process. For more information about running these three files, see “Controlling the SAS Embedded Process” on page 177.

Install the SAS Embedded Process RPM Package File for SAP HANA – Multi-Node Deployment

To install the SAS Embedded Process RPM package file, follow these steps:

1. Create a directory on the master node of the multi-node SAP HANA appliance. This directory is referred to as RPMTempDir throughout this section.

   Note: For multi-node deployments, SAS Embedded Process is deployed on the master node first and then on the worker nodes. The deployment is easier if the master node is on a shared file system that is accessible to all worker nodes. If the file system is not shared, you must create an install directory, copy the RPM file to that directory, and install the RPM file on each worker node. For more information about installing the RPM file on each worker node, see Step 7 on page 173.

2. Locate the SAS Embedded Process RPM package file, sas-sepcorehana-<version>-<date.timestamp>-<rpm-os-designation>.x86_64.rpm.

   The following table describes the file name options:

   Table 19.3  RPM File Name Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “SAP HANA SAS Embedded Process Versions” on page 183.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time at which the RPM package was created.</td>
</tr>
<tr>
<td>rpm-os-designation</td>
<td>The operating system for which the RPM file was created.</td>
</tr>
<tr>
<td></td>
<td>For SAP HANA, the permissible value is suse. There is no rpm-os-designation value for Red Hat Enterprise Linux.</td>
</tr>
</tbody>
</table>

   This file is located in one of the following directories depending on your operating system:

   - Linux — /path-to-mirror-repository/repos/shipped/sephana/100.0/sephana-100.0.0-repo-os-designation-<option-if-specified>/Packages/s/
Windows — C:\path-to-mirror-repository\repos\shipped\sephana\100.0\sephana-100.0.0-repo-os-designation\option-if-specified\Packages\s\<-

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of &quot;Create a Mirror Repository&quot; on page 135. Here is an example: &lt;br&gt; /mySas/sas_repos/</td>
</tr>
<tr>
<td>repo-os-designation\option-if-specified&gt;</td>
<td>repo-os-designation identifies the operating system. Available values depend on your data source and software order. Possible values are as follows: &lt;br&gt; x64_suse_linux_12-yum &lt;br&gt; x64_redhat_linux_6-yum &lt;br&gt; If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in option-if-specified. For example, if you specified --latest --platform x64-redhat-linux-6 in your mirrormgr mirror command, the directory would be x64_redhat_linux_6-yum-latest.</td>
</tr>
</tbody>
</table>

3 Copy the sas-sepcorehana-version-date.timestamp.<rpm-os-designation>.x86_64.rpm to the RPMTempDir directory on the master node. Make sure that you copy the file to the server machine according to the procedures that are used at your site. Here is an example of a secure copy command:

scp sas-sepcorehana-16.0.0.20180821.1822334532.suse.x86_64.rpm root@hanamach1:RPMTempDir

Note: The sas-sepcorehana-version-date.timestamp.<rpm-os-designation>.x86_64.rpm file and the RPMTempDir directory require root privileges for the database administrator.

Note: The RPMTempDir directory must be in a location where it can be accessed from at least one of the SAP HANA nodes.

4 After the sas-sepcorehana-version-date.timestamp.<rpm-os-designation>.x86_64.rpm has been transferred, log on to the SAP HANA server as the owner of the SAS Embedded Process installation directory:
5 Navigate to the directory where the RPM file was transferred in step 2.

   cd /RPMTempDir

6 Install the RPM file.

   rpm -ivh sas-sepcorehana-version-date.timestamp.<.rpm-os-designation>.x86_64.rpm

   After the RPM file is installed, a /opt/sas/ep/home/ directory is created in which the SAS Embedded Process files are installed. The contents of the /opt/sas/ep/home/ directory should look similar to the following listing. Directories and files of interest are highlighted.

   /opt/sas/ep/home/bin
   /opt/sas/ep/home/bin/ShowSASEPStatus.sh
   /opt/sas/ep/home/bin/ShutdownSASEP.sh
   /opt/sas/ep/home/bin/StartupSASEP.sh
   /opt/sas/ep/home/misc
   /opt/sas/ep/home/misc/sas_saslink_installer.tgz
   /opt/sas/ep/home/sasexe
   /opt/sas/ep/home/utilities

   The ShowSASEPStatus.sh file shows the status of SAS Embedded Process on each instance. The StartupSASEP.sh and ShutdownSASEP.sh files enable you to manually start and stop SAS Embedded Process. For more information about running these three files, see “Controlling the SAS Embedded Process” on page 177.

7 Install SAS Embedded Process on the worker nodes.

   If the /opt/sas/ep/home/ directory is on a shared file system that is accessible to all worker nodes, follow these steps for each worker node:

   1 Log on to the worker node.

   2 Create a directory to which SAS logs are written. Here is an example:

      mkdir /var/tmp/SAS
      chmod 777 /var/temp/SAS

   3 Create symbolic links to SAS Embedded Process.

      ln -s /opt/sas/ep/home/sasexe/hana_saslink_ep.so /usr/local/lib/hana_saslink_ep.so
      ln -s /opt/sas/ep/home/misc/tkhnmain.sh /usr/local/bin/tkhnmain

      Note: If you installed the RPM to a non-default location, use both of these commands:

      ln -s /RPMInstallDir/sasexe/hana_saslink_ep.so /usr/local/lib/hana_saslink_ep.so
      ln -s /RPMInstallDir/misc/tkhnmain.sh /usr/local/bin/tkhnmain

      TIP If you need to remove symbolic links, use these commands:

      unlink /usr/local/lib/hana_saslink_ep.so
If the /opt/sas/ep/home/ directory is not on a shared file system that is accessible to all worker nodes, follow these steps for each worker node:

1. Create a directory on the SAP HANA appliance, which is referred to as RPMTempDir.

2. Copy the sas-sepcorehana-version-date.timestamp<.rpm-os-designation>.x86_64.rpm file from the RPMTempDir on the master node to the worker node.


SASLINK AFL Plug-ins and Auxiliary Wrapper Procedures

Installing the SASLINK AFL Plug-ins on the Appliance

The SASLINK Application Function Library (AFL) files are included in an installer that is packaged as a tarball (TAR file) and that is provided when the SAS Embedded Process self-extracting archive file is unpacked.

Note: The SID that is referenced in these instructions is the SAP HANA system identifier (for example, HDB).

To install the SASLINK AFL plugins on the appliance, follow these steps:

1. Log on to the SAP HANA server as the database administrator or change the user to the database administrator.
   
   You can use either of these commands:
   
   su - SIDadm
   
   ssh SIDadm@server-name

2. If SAS Embedded Process is running, run the ShutdownSASEP.sh script to stop the process.

   cd /opt/sas/ep/home/bin
   ./ShutdownSASEP.sh

   Alternatively, you can shut down SAS Embedded Process by removing its PID file.

   rm /var/tmp/tkhnmain.pid
To check the status of SAS Embedded Process, run these commands:

```bash
cd /opt/sas/ep/home/bin
./ShowSASEPStatus.sh
```

3. Stop the SAP HANA database if it is running.
```bash
HDB stop
```

4. Use either of these commands or type `exit` to change the user to the root authority.
```bash
su - root
sudo su -
```

5. Copy the TAR file to the `/tmp` directory.
```bash
cp /opt/sas/ep/home/misc/sas_saslink_installer.tgz /tmp
```

6. Unpack the TAR file.
```bash
cd /tmp
tar -xvzf sas_saslink_installer.tgz
```

7. Run the HANA installation utility from the directory to which the TAR file was unpacked. Specify the SID of the HANA instance when prompted by the installation utility.
```bash
cd /sas_saslink_installer/installer
./hdbinst
```

8. Use either of these commands to change the user back to the database administrator or to change the user to the database administrator.
```bash
su - SIDadm
exit
```

9. Restart the SAP HANA database.
```bash
HDB start
```

10. Start SAS Embedded Process on all nodes.
```bash
cd /opt/sas/ep/home/bin/
./StartupSASEP.sh
```

Auxiliary Wrapper Procedures

Operation of the SASLINK AFL and SAS Embedded Process requires wrapper procedures that are already installed in the SAP HANA catalog on the server.

**Note:** SAS Embedded Process for SAP HANA is invoked by an AFL wrapper that is generated by the `SAP HANA AFLLANG_WRAPPER_PROCEDURE_CREATE` procedure. This procedure takes as input the table type definitions that identify the
input and output table columns. The column names cannot contain special characters. Otherwise, the procedure fails.

However, an additional permission, AFLPM_CREATOR_ERASER_EXECUTE, is required. For more information, see “SAP HANA Permissions” on page 178.

Importing the SAS_EP Stored Procedure

The SAS_EP Stored Procedure is used by the %INDHN_RUN_MODEL macro to run the scoring model.

The SAS_EP stored procedure is contained in a delivery unit named SAS_EP_sas.com.tgz. The SAS_EP_sas.com.tgz package was installed in the /opt/sas/ep/home/bin/misc directory when SAS Embedded Process was deployed.

To import the delivery unit into SAP HANA, follow these steps:

Note: Permissions and roles are required in order to import the procedure package. For more information, see “SAP HANA Permissions” on page 178.

1. Navigate to the /opt/sas/ep/home/bin/misc directory.
2. Copy the SAS_EP_sas.com.tgz package to a client machine on which the SAP HANA Studio client is installed.
3. Import the delivery unit.

   There are several methods of importing the .tgz file. Examples are SAP HANA Studio or the Lifecycle Manager. To import the delivery unit using SAP HANA Studio, follow these steps:

   a. Ensure that you have a connection to the target SAP HANA back end from your local SAP HANA Studio.
   b. Ensure that at least the SAP HANA Studio Database Development module is installed.
   c. Select File \ Import.
   d. Select SAP HANA Content \ Delivery Unit and click Next.
   e. Select the target system and click Next.
   g. Select the Overwrite inactive versions and Activate object check boxes. The list of objects is displayed under Object import simulation.
   h. Click Finish to import the delivery unit.
Controlling the SAS Embedded Process

The SAS Embedded Process starts when you run the StartupSASEP.sh script. It continues to run until it is manually stopped or the database is shut down.

**Note:** Starting and stopping the SAS Embedded Process has implications for all scoring model publishers.

**Note:** Manually starting and stopping the SAS Embedded Process requires HANA database administrator user permissions.

When the SAS Embedded Process is installed, the ShutdownSASEP.sh and StartupSASEP.sh scripts are installed in the `/opt/sas/ep/home/bin` directory. For more information about these files, see “Deploying SAS Embedded Process for SAP HANA” on page 167.

Use the following command to start the SAS Embedded Process:

```
./StartupSASEP.sh
```

**Note:** The `-verbose` option is on by default and provides a status of the start-up operations as they occur. You can specify the `-quiet` option to suppress messages.

ShutdownSASEP.sh shuts down the SAS Embedded Process. It is designed to be used to shut down the SAS Embedded Process prior to a database upgrade or re-install. This script should not be used as part of the normal operation.

Use the following command to shut down the SAS Embedded Process:

```
./ShutdownSASEP.sh
```

**Note:** The `-verbose` option is on by default and provides a status of the shutdown operations as they occur. You can specify the `-quiet` option to suppress messages.

Semaphore Requirements When Using the SAS Embedded Process for SAP HANA

Each time a query using the SAS_EP stored procedure is invoked to execute a score, it requests a set of semaphore arrays (sometimes referred to as semaphore "sets") from the operating system. The SAS Embedded Process releases the semaphore arrays back to the operating system after scoring is complete.
The SAP HANA server that runs the SAS Embedded Process should be configured with a minimum of 1024 to 2048 semaphore arrays.

**Note:** The semaphore limit on the “maximum number of arrays” is distinct from the semaphore limit on the “maximum number of semaphores system wide”. The Linux `ipcs -sl` command shows the typical default semaphore-related limits set on SAP HANA:

```
------- Semaphore Limits -------
max number of arrays = 2048
max semaphores per array = 250
max semaphores system wide = 512000
max ops per semop call = 100
semaphore max value = 32767
```

---

**SAP HANA Permissions**

The following permissions are needed by the person who installs the in-database deployment package:

**Note:** Some of the permissions listed below cannot be granted until the Auxiliary Wrapper Generator and Eraser Procedures are installed. For more information, see “SASLINK AFL Plug-ins and Auxiliary Wrapper Procedures” on page 174.

---

**Table 19.5  Permissions Needed to Install the SAP HANA In-Database Deployment Package**

<table>
<thead>
<tr>
<th>Task</th>
<th>Permission Needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unpack the self-extracting archive file</td>
<td>owner of the SAS Embedded Process install directory. The SAS Embedded Process install directory must have permissions that allow Read and Execute permission by the database administrator user.</td>
</tr>
<tr>
<td>Install or uninstall the SAS Embedded Process (run InstallSASEPFiles.sh or UninstallSASEPFiles.sh script)</td>
<td>root authority</td>
</tr>
<tr>
<td>Import the SAS_EP procedure package</td>
<td>a user on the SAP HANA server that has at least the CONTENT_ADMIN role or its equivalent</td>
</tr>
<tr>
<td>Install AFL plugins (requires starting and stopping the database)</td>
<td>root authority and database administrator</td>
</tr>
</tbody>
</table>
The following permissions are needed by the person who runs the scoring models. Without these permissions, the publishing of the scoring models fails:

SAP HANA SPS11:
- AFLPM_CREATOR_ERASER_EXECUTE to userid | role;
- EXECUTE, SELECT, INSERT, UPDATE, and DELETE on the schema that is used for scoring

In addition, the roles of sas.ep::User and AFL__SYSP__AFL__SASLINK__AREA_EXECUTE must be assigned to any user who wants to perform in-database processing. The sas.ep::User role is created when you import the SAS_EP stored procedure. The AFL__SYSP__AFL__SASLINK__AREA_EXECUTE role is created when the AFL wrapper generator is created.

Note: If you plan to use SAS Model Manager with the SAS Scoring Accelerator for in-database scoring, additional permissions are required. For more information, see “Configuring SAS Model Manager” in SAS In-Database Products: Administrator’s Guide.

Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Scoring Accelerator for SAP HANA</td>
<td>“SAP HANA Administrator’s Guide” in SAS In-Database Products: Administrator’s Guide</td>
</tr>
<tr>
<td>SAS High-Performance Analytics</td>
<td>SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide</td>
</tr>
</tbody>
</table>
Upgrading or Uninstalling SAS Embedded Process for SAP HANA

Upgrade from a Previous SAP HANA Version – SAS 9.4M6 and Later

Upgrade from a Previous SAP HANA Version – Prior to SAS 9.4M6

Uninstall SAS Embedded Process for SAP HANA

SAP HANA SAS Embedded Process Versions

Upgrade from a Previous SAP HANA Version – SAS 9.4M6 and Later

To find out when your version of SAS Embedded Process was released, see “Upgrade from a Previous SAP HANA Version – Prior to SAS 9.4M6” on page 182.

To upgrade from a previous version, follow these steps.

1. Log on to the SAP HANA system as root.
   You can use `su` or `sudo` to become the root authority.

2. Remove the RPM file.

   - If the `/opt/sas/ep/home/bin/` directory is on a shared file system that is accessible to all worker nodes, follow these steps for each worker node:
     1. If SAS Embedded Process is running, run the `ShutdownSASEP.sh` script to stop the process.
        ```bash
        cd /opt/sas/ep/home/bin
        ./ShutdownSASEP.sh
        ```
     2. Remove the symbolic links.
        ```bash
        unlink /usr/local/lib/hana_saslink_ep.so
        unlink /usr/local/bin/tkhnmain
        ```
     3. Enter this command to remove the RPM file.
        ```bash
        rpm -e sas-sepcorehana
        ```
        You can verify the removal with the following command.
        ```bash
        rpm -qa sas-sepcorehana
        ```
If the `/opt/sas/ep/home/` directory is not on a shared file system that is accessible to all worker nodes, follow these steps for each worker node:

1. If SAS Embedded Process is running, run the `ShutdownSASEP.sh` script to stop the process.
   ```bash
cd /opt/sas/ep/home/bin/ShutdownSASEP.sh
   ```

2. Navigate to the directory that contains the RPM file.
   ```bash
   cd /opt/sas/ep/home/bin
   ```
   
   Note: You can find out the location of the RPM file by using the following command:
   ```bash
   ls -l /usr/local/bin/tkhnmain
   ```

3. Remove the RPM file.
   ```bash
   rpm -e sas-sepcorehana
   ```
   You can verify the removal with the following command.
   ```bash
   rpm -qa sas-sepcorehana
   ```


---

Upgrade from a Previous SAP HANA Version – Prior to SAS 9.4M6

To find out when your version of SAS Embedded Process was released, see “Upgrade from a Previous SAP HANA Version – Prior to SAS 9.4M6” on page 182.

To upgrade from a previous version, follow these steps.

1. Log on to the SAP HANA system as root.
   You can use `su` or `sudo` to become the root authority.

2. Navigate to the directory that contains the `UninstallSASEPFiles.sh` script.
   ```bash
   cd /EPInstallDir/SASEPHome/bin
   ```
   
   Note: You can find out the location of `EPInstallDir` by using the following command:
   ```bash
   ls -l /usr/local/bin/tkhnmain
   ```

3. Run the `UninstallSASEPFiles.sh` script.
   ```bash
   ./UninstallSASEPFiles.sh
   ```
   This script stops SAS Embedded Process on the server.
For versions prior to SAS 9.4M4 (November 2016), the script deletes the `/SAS/SASTKInDatabaseServerForSAPHANA` directory and all its contents. For versions starting with SAS 9.4M4 (November 2016), the script deletes the `/SASEPHome/bin` directory and all its contents.


Uninstall SAS Embedded Process for SAP HANA

The method that you use to uninstall SAS Embedded Process for SAP HANA is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the final step, which is about continuing the installation.

- “Upgrade from a Previous SAP HANA Version – SAS 9.4M6 and Later” on page 181
- “Upgrade from a Previous SAP HANA Version – Prior to SAS 9.4M6” on page 182

To find out when your version of SAS Embedded Process was released, see “Upgrade from a Previous SAP HANA Version – Prior to SAS 9.4M6” on page 182.

SAP HANA SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since August 2014 (SAS 9.4M2).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Starting with SAS 9.4M6, SAS 9.4 customers can get the latest version of SAS Embedded Process at any time.

<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcorehana-16.0.0-date.timestamp.x86_64.rpm (Red Hat Enterprise Linux)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>sas-sepcorehana-16.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td>Date</td>
<td>SAS Release</td>
<td>SAS Embedded Process File Name</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcorehana-13.00000-1.sh</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcorehana-12.00000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.42-n_lax.sh</td>
</tr>
</tbody>
</table>
PART 9

SAS Embedded Process Deployment for Teradata

Chapter 21
Deploying SAS Embedded Process for Teradata ......................... 187

Chapter 22
Upgrading or Uninstalling SAS Embedded Process for Teradata .... 197
# Deploying SAS Embedded Process for Teradata

## Teradata Prerequisites

The SAS Embedded Process deployment for Teradata requires the following:

- A specific version of the Teradata client and server environment. For more information, see the SAS Foundation system requirements documentation for your operating system.
- For SAS Viya, the CAS controller and each CAS worker node must have an IP address that can be routed to externally from the SAS Embedded Process nodes.
- You must have root authority to deploy SAS Embedded Process and run the install scripts. To accomplish this, run as the root user or as a user with sudo privileges.
- Approximately 300 MB of disk space is required in the /opt file system on each Teradata Trusted Parallel Appliance (TPA) node.
- Connection to the internet and, specifically, SAS repositories online are required to download the package and create a mirror repository. If you receive errors about connecting to the internet, you should work with your system administrator to correct them.
Overview of SAS Embedded Process for Teradata

SAS Embedded Process for Teradata is available with the following SAS software products and must be deployed before you can use the in-database processing features of that software:

- SAS Scoring Accelerator for Teradata
  For more information about using this software, see “SAS Scoring Accelerator for Teradata” in SAS In-Database Products: User’s Guide.

- SAS In-Database Code Accelerator for Teradata
  For more information about using this software, see “Using the SAS In-Database Code Accelerator” in SAS In-Database Products: User’s Guide.

- SAS Data Quality Accelerator for Teradata
  For information about using this software, see SAS Data Quality Accelerator for Teradata: User’s Guide.

- In-Database Technologies for Teradata (on SAS Viya) (includes the SAS Data Connect Accelerator for Teradata)
  For more information about using this software, see “Teradata Data Connector” in SAS Cloud Analytic Services: User’s Guide.

- SAS High-Performance Analytics
  For information about using this software, see SAS High-Performance Analytics Server: User’s Guide.

SAS Embedded Process is a SAS server process that runs within Teradata to read and write data. SAS Embedded Process contains macros, run-time libraries, and other software that is installed on your Teradata system.

Note: To use your software, some additional configuration might be required. For more information, see “Where to Go from Here” on page 195.

Note: If you are adding additional nodes, the version of SAS Embedded Process must be identical for existing and new nodes.

Teradata Deployment Steps

To deploy SAS Embedded Process for Teradata, follow these steps:
<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(Optional) If you are upgrading from (or reinstalling) a previous release in SAS 9.4, follow these instructions. Otherwise, continue with Step 2.</td>
<td>Chapter 22, “Upgrading or Uninstalling SAS Embedded Process for Teradata,” on page 197</td>
</tr>
<tr>
<td>3</td>
<td>Install the SAS Embedded Process support functions.</td>
<td>“Install the SAS Embedded Process Support Functions” on page 193</td>
</tr>
<tr>
<td>4</td>
<td>Review how to control SAS Embedded Process.</td>
<td>“Controlling the SAS Embedded Process” on page 193</td>
</tr>
<tr>
<td>5</td>
<td>Review the permissions that are needed for in-database processing.</td>
<td>“Teradata Permissions for Publishing Formats and Scoring Models” on page 194</td>
</tr>
<tr>
<td>6</td>
<td>If you license the following software, perform additional configuration as required.</td>
<td>“Where to Go from Here” on page 195</td>
</tr>
<tr>
<td></td>
<td>- SAS Scoring Accelerator for Teradata</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- SAS Code Accelerator for Teradata</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- SAS Data Quality Accelerator for Teradata</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- In-Database Technologies for Teradata (on SAS Viya) (includes the SAS Data Connect Accelerator for Teradata)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- SAS High-Performance Analytics</td>
<td></td>
</tr>
</tbody>
</table>

**Deploying SAS Embedded Process for Teradata**

**Create a Mirror Repository**

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all SAS Embedded Process deployments.

To create a mirror repository, follow these steps:

1. The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine where you plan to create your mirror repository.
2 Download and unpack SAS Mirror Manager to a directory on the machine where you want to put your mirror repository by following the instructions for your operating system:

- Linux – See Appendix 1, “SAS Mirror Manager for Linux,” on page 205.

**TIP** For Linux and Macintosh, the file has a .gz extension. Save the file with a .tgz extension if you download it from Internet Explorer. For Windows, the file has a .zip extension.

Here is an example of a Linux command to unpack the SAS Mirror Manager:

```
tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz
```

3 Run the following commands to provide platform and repository information that can be helpful if you need to specify the `--platform` or `--repo` options in the `mirrormgr mirror` command in the next step.

```
./mirrormgr list remote platforms --deployment-data
/path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip

./mirrormgr list remote repos --deployment-data
/path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
```

4 Create a mirror repository for the deployment data file, SAS_Viya_deployment_data.zip. This step determines which files to download from the repository.

```
./mirrormgr mirror --deployment-data
/path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
<options>
```

*Note:* SAS Mirror Manager downloads all entitled repositories to the `/user-home/sas_repos` (Linux) or `C:\Users\user-id\sas_repos` (Windows) directory by default. You can change the location by using the `--path path-to-mirror-repository` option in the `mirrormgr mirror` command. You can also choose to download only one platform by using the `--platform platform-name` option.

Here is an example for a Linux SUSE system. In this example, only the latest version of the SUSE Linux files are downloaded.

```
./mirrormgr mirror --deployment-data
/path-to-location-of-deployment-zip-file-from-SOE/SAS_Viya_deployment_data.zip
--platform x64-suse-linux-12 --latest
```

*Note:* SAS Mirror Manager option choices can be found in Appendix 1, “SAS Mirror Manager for Linux,” on page 205 and Appendix 2, “SAS Mirror Manager for Windows,” on page 211.
Move the SAS Embedded Process RPM Package File to Teradata

To move the SAS Embedded Process RPM package file to the Teradata appliance, follow these steps:

1. Locate the SAS Embedded Process RPM package file, sas-sepcoretera-version-date.timestamp.suse.x86_64.rpm.

The following table describes the file name options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>The version of the RPM file.</td>
</tr>
<tr>
<td></td>
<td>For more information, see “Teradata SAS Embedded Process Versions” on page 201.</td>
</tr>
<tr>
<td>date.timestamp</td>
<td>The date and time at which the RPM package was created.</td>
</tr>
</tbody>
</table>

This file is located in one of the following directories depending on your operating system:

- **Linux** — /path-to-mirror-repository/repos/shipped/septera/100.0/septera-100.0.0<–option-if-specified>/Packages/s/
- **Windows** — c:/path-to-mirror-repository/repos/shipped/septera \100.0\septera-100.0.0<–option-if-specified>/Packages/s/

The following table describes the path options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path-to-mirror-repository</td>
<td>The full path to where SAS Mirror Manager downloaded the repository in step 4 of “Create a Mirror Repository” on page 135. An example is /mysas/sas_repos/.</td>
</tr>
<tr>
<td>&lt;–option-if-specified&gt;</td>
<td>If you specified options when you created your mirror repository with SAS Mirror Manager, the value is used in option-if-specified. For example, if you specified --latest in your mirrormgr mirror command, the directory would be -latest.</td>
</tr>
</tbody>
</table>
Copy the sas-sepcoretera-version-date.timestamp.suse.x86_64.rpm to a location on the Teradata server. Make sure that you copy the file to the server machine according to the procedures that are used at your site. Here is an example of a secure copy command.

```
scp sas-sepcoretera-17.0.0-20191005.1822334532.suse.x86_64.rpm
root@teramach1:/RPMTEMPDir
```

Note: The RPMTEMPDir directory must be in a location where it can be accessed from at least one of the Teradata nodes.

This RPM file is readable by the Teradata Parallel Upgrade Tool.

### Deploy SAS Embedded Process with the Teradata Parallel Upgrade Tool

This installation should be performed by a Teradata system administrator in consultation with Teradata Customer Services. A Teradata Change Control is required when an RPM file is added to the Teradata server. Teradata Customer Services has developed change control procedures for installing SAS Embedded Process.

These deployment steps assume knowledge about the Teradata Parallel Upgrade Tool and your environment. For more information about using the Teradata Parallel Upgrade Tool, see the *Parallel Upgrade Tool (PUT) Reference* which is at the Teradata Online Publications site, located at https://docs.teradata.com/home. On this page, search for “Parallel Upgrade Tool” and download the appropriate document for your system.

The following steps explain the basic steps to install SAS Embedded Process by using the Teradata Parallel Upgrade Tool.

Note: The Teradata Parallel Upgrade Tool is a third-party product. Therefore, its prompts are subject to change.

---

**CAUTION**

You must remove previous versions of the SAS Embedded Process RPM package before you deploy the new version. For more information, see “Uninstall SAS Embedded Process for Teradata” on page 201.

---

1. Locate the sas-sepcoretera-version-date.timestamp.suse.x86_64.rpm file on your machine.

   *date.timestamp* is the date and time at which the repository was created.

   This file is located in the directory where you moved it in *Step 2 on page 192*.

   Note: It must be in a location where it can be accessed from at least one of the Teradata nodes.

2. Start the Teradata Parallel Upgrade Tool.
3 Be sure to select all Teradata TPA nodes for installation, including Hot Stand-By nodes.

4 If Teradata Version Migration and Fallback (VM&F) is installed, you might be prompted whether to use VM&F. If you are prompted, choose Non-VM&F installation.

5 If the installation is successful, `sas-sepcoretera-version-date.timestamp.suse.x86_64.rpm` is displayed.

Alternatively, you can manually verify that the installation is successful by running this command:

```
psh "rpm -qa \\"*sepcore*""
```

---

### Install the SAS Embedded Process Support Functions

The SAS Embedded Process support function package (sasepfunc) includes stored procedures that generate SQL that is used as an interface to SAS Embedded Process. The support function package also includes functions that load the SAS program and other run-time control information into shared memory. The setup script for the support function package creates the `SAS_SYSFNLIB` database and the SAS Embedded Process interface fast path functions in `TD_SYSFNLIB`.

In addition, the support function package installs `grant.sh`, which can be used by an administrator to grant permissions for SAS Embedded Process. An administrator must run the script from a machine with the Teradata BTEQ utility for each SAS Embedded Process user.

**Note:** If you are using the SAS Embedded Process version 17 or later, you must use the Unicode version of the SAS Embedded Process support functions (sasepfunc_unicode-16.20-3 or later). If you are using the SAS Embedded Process version 16 or earlier, you must use the Latin version of the SAS Embedded Process support functions (sasepfunc_latin-16.20-2.1 or later).

The SAS Embedded Process support function package is available from the Teradata Software Server. For access to the package that includes the installation instructions, contact your local Teradata account representative or the Teradata consultant who supports your SAS and Teradata integration activities.

---

### Controlling the SAS Embedded Process

The SAS Embedded Process starts when a query is submitted. The SAS Embedded Process continues to run until it is manually stopped or the database is shutdown. You might want to disable or shutdown the SAS Embedded Process without shutting down the database.

The following commands control the SAS Embedded Process.
<table>
<thead>
<tr>
<th>Action performed</th>
<th>Command (by Teradata version)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provides the status of the SAS Embedded Process.</td>
<td>CALL SQLJ.SERVERCONTROL ('SAS', 'status', :A);</td>
</tr>
<tr>
<td>Shuts down the SAS Embedded Process.</td>
<td>CALL SQLJ.SERVERCONTROL ('SAS', 'shutdown', :A);</td>
</tr>
<tr>
<td>Note: You cannot shut down until all queries are complete.</td>
<td></td>
</tr>
<tr>
<td>Stops new queries from being started.</td>
<td>CALL SQLJ.SERVERCONTROL ('SAS', 'disable', :A);</td>
</tr>
<tr>
<td>Queries that are currently running continue to run until they are complete.</td>
<td></td>
</tr>
<tr>
<td>Enables new queries to start running.</td>
<td>CALL SQLJ.SERVERCONTROL ('SAS', 'enable', :A);</td>
</tr>
</tbody>
</table>

1 Note that the Languagename parameter, 'SAS', is required and must be uppercase. The Cmd parameter (for example, 'status'), must be lowercase.

Here is the sequence of operations to stop and then restart the SAS Embedded Process:

1 Disable the SAS Embedded Process to stop new queries from being started.
   CALL SQLJ.SERVERCONTROL ('SAS', 'disable', :A);

2 Query the status of the SAS Embedded Process until the status returns this message: Hybrid Server is disabled with no UDFs running.
   CALL SQLJ.SERVERCONTROL ('SAS', 'status', :A);

3 Shutdown the SAS Embedded Process.
   CALL SQLJ.SERVERCONTROL ('SAS', 'shutdown', :A);

4 Perform maintenance on the SAS Embedded Process, for example, install a hot fix or upgrade to a new version.

5 Enable the SAS Embedded Process.
   CALL SQLJ.SERVERCONTROL ('SAS', 'enable', :A);

6 Test the SAS Embedded Process. The SAS Embedded Process will start when the next SAS query that uses the SAS Embedded Process is sent to the database.

---

Teradata Permissions for Publishing Formats and Scoring Models

Because functions are associated with a database, the functions inherit the access rights of that database. It might be useful to create a separate shared database for the SAS scoring functions or the SAS_PUT( ) function so that access rights can be customized as needed.
You must grant the following permissions to any user who runs the scoring or format
publishing macros:

CREATE FUNCTION ON database TO userid
DROP FUNCTION ON database TO userid
EXECUTE FUNCTION ON database TO userid
ALTER FUNCTION ON database TO userid

If you use the SAS Embedded Process to run your scoring model, you must grant
the following permissions:

SELECT, CREATE TABLE, INSERT ON database TO userid
EXECUTE PROCEDURE ON SAS_SYSFNLIB TO userid
EXECUTE FUNCTION ON SAS_SYSFNLIB TO userid
EXECUTE FUNCTION ON SYSLIB.MonitorVirtualConfig TO userid

Note: If you plan to use SAS Model Manager with the SAS Scoring Accelerator for
in-database scoring, additional permissions are required. For more information, see
“Configuring SAS Model Manager” in SAS In-Database Products: Administrator’s
Guide.

### Where to Go from Here

If you licensed the following software, you must perform additional configuration.

<table>
<thead>
<tr>
<th>Software</th>
<th>Where to Go for Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAS Data Quality Accelerator for Teradata</td>
<td>“SAS Data Quality Accelerator for Teradata” in SAS In-Database Products: Administrator’s Guide</td>
</tr>
<tr>
<td>Software</td>
<td>Where to Go for Information</td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>In-Database Technologies for Teradata (on SAS Viya) (includes the SAS Data Connect Accelerator for Teradata)</td>
<td>Data that is transferred between the data provider and the CAS server is not encrypted by default. To enable encryption on Teradata, you must perform additional steps after you install SAS Embedded Process. For more information, see Encrypt Data Transfer When Using the SAS Data Connect Accelerator in SAS Viya for Linux: Deployment Guide.</td>
</tr>
<tr>
<td>SAS High-Performance Analytics</td>
<td>SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide</td>
</tr>
</tbody>
</table>
Upgrade from a Previous Teradata Version – SAS 9.4M6 and Later

To find out when your version of SAS Embedded Process was released, see “Teradata SAS Embedded Process Versions” on page 201.

To upgrade from a previous version of SAS Embedded Process on SAS Viya, follow these steps:

1. Run this command to check the currently installed version of SAS Embedded Process and any hot-fix packages.
   
   ```bash
   psh "rpm -qa \"*sepcore*\""
   ```

   If a previous version is installed, a result that is similar to the following is displayed. The version number might be different.

   ```bash
   sas-sepcoretera-16.0.0-20180821.1822334532.suse.x86_64.rpm
   ```

   If SAS Embedded Process is not installed on the Teradata nodes, no output is displayed. You can continue with the installation steps in “Deploying SAS Embedded Process for Teradata” on page 189.

2. If the SAS Data Quality Accelerator for Teradata is installed, you must uninstall it before you uninstall SAS Embedded Process. For more information, see “Removing SAS Data Quality Accelerator from the Teradata Database” in SAS In-Database Products: Administrator’s Guide.

3. Remove the RPM package.
Shut down SAS Embedded Process.

CALL SQLJ.SERVERCONTROL ('SAS', 'shutdown', :A);

For more information about how to shut down SAS Embedded Process, see "Controlling the SAS Embedded Process" on page 193.

b Remove the previous version before you upgrade to the latest version of SAS Embedded Process.

- To remove the packages from all nodes concurrently, run this command:
  
  psh "rpm -e sas-sepcoretera"

  For example, to remove sas-sepcoretera-16.0.0-20180821.182233344532.suse.x86_64.rpm, run this command:
  
  psh "rpm -e sas-sepcoretera"

- To remove the package from each node, run this command on each node:
  
  rpm -e sas-sepcoretera

4 (Optional) To confirm that the sepcoretera package was removed before you install the latest package, run this command:

  psh "rpm -qa sas-sepcoretera"

  SAS Embedded Process should not appear on any node.

5 Continue with the installation steps in "Deploying SAS Embedded Process for Teradata" on page 189.

---

Upgrade a Previous Teradata Version – SAS 9.4M3 (July 2015) Prior to SAS 9.4M6

To find out when your version of SAS Embedded Process was released, see “Teradata SAS Embedded Process Versions” on page 201.

To upgrade from a previous version of SAS Embedded Process, follow these steps:

1 Run this command to check the currently installed version of SAS Embedded Process and any hot-fix packages.

  psh "rpm -qa \"*sepcore*\""

  If a previous version is installed, a result similar to the following is displayed. The version number might be different.

  sepcoretera-13.00000-1
  sepcoreterahf-13.00000-1

  If SAS Embedded Process is not installed on the Teradata nodes, no output is displayed. You can continue with the installation steps in "Deploying SAS Embedded Process for Teradata" on page 189.

2 If the SAS Data Quality Accelerator for Teradata is installed, you must uninstall it before you uninstall SAS Embedded Process. For more information, see
“Removing SAS Data Quality Accelerator from the Teradata Database” in SAS In-Database Products: Administrator’s Guide.

3 If a version of SAS Embedded Process is being installed, and it has a name that is different from the library that was previously installed, then follow these steps. An example is sepcoretera-17.0.0 that will replace tkindbsrv-9.42_M1-2.

Note: Starting with SAS 9.4M5, if the package-name is sepcoretera, the previous sepcoretera package does not have to be removed before the most recent sepcoretera package is installed. However, a sepcoreterahf package must always be removed before you install a new SAS Embedded Process package.

Note: You must remove both the SAS Embedded Process package (if it has a name other than sepcoretera) and any SAS Embedded Process hot-fix package that exists before you upgrade to a new version of SAS Embedded Process. If a hot fix is installed, it blocks the upgrade to a new SAS Embedded Process package. If you force the upgrade to the new package and then later remove the old hot fix, there is a risk that the removal of the previous hot fix also removes files that were delivered in the most recent SAS Embedded Process installation.

a Shut down SAS Embedded Process.

CALL SQLJ.SERVERCONTROL ('SAS', 'shutdown', :A);

For more information about how to shut down SAS Embedded Process, see “Controlling the SAS Embedded Process” on page 193.

b Remove the previous version before you upgrade SAS Embedded Process.

- To remove the packages from all nodes concurrently, run this command:

  psh "rpm -e package-name"

  The value for package name is either tkindbsrv or sepcoreterahf.

  For example, to remove tkindbsrv, run the command psh "rpm -e tkindbsrv".

- To remove the package from each node, run this command on each node:

  rpm -e package-name

  The value for package-name is either tkindbsrv or sepcoreterahf.

4 (Optional) To confirm the removal of the package before you upgrade to the new package, run this command:

  psh "rpm -qa package-name"

  The value for package-name is either tkindbsrv or sepcoreterahf.

  SAS Embedded Process should not appear on any node.

5 Continue with the installation steps in “Deploying SAS Embedded Process for Teradata” on page 189.
Upgrade from a Previous Teradata Version – Prior to SAS 9.4M3 (July 2015)

To find out when your version of SAS Embedded Process was released, see “Teradata SAS Embedded Process Versions” on page 201.

To upgrade from a previous version of SAS Embedded Process, follow these steps:

1. Run this command to check the currently installed version of SAS Embedded Process.
   
   \texttt{psh "rpm -qa '*tkindbsrv*'"}

   If a previous version is installed, a result that is similar to the following is displayed. The version number might be different.

   \texttt{tkindbsrv-9.42_M1-2.x86_64}

   If SAS Embedded Process is not installed on the Teradata nodes, no output is displayed. You can continue with the installation steps in “Deploying SAS Embedded Process for Teradata” on page 189.

2. If a version of SAS Embedded Process is being deployed that has a name that is different from the library that was previously deployed, then follow these steps. An example is sepcoretera-17.0.0 that replaces tkindbsrv-9.42_M1-2.
   
      
      \texttt{CALL SQLJ.SERVERCONTROL ('SAS', 'shutdown', :A)};
      
      For more information about how to shut down SAS Embedded Process, see “Controlling the SAS Embedded Process” on page 193.

   b. If the SAS Data Quality Accelerator for Teradata is installed, you must uninstall it before you uninstall SAS Embedded Process. For more information, see “Removing SAS Data Quality Accelerator from the Teradata Database” in \textit{SAS In-Database Products: Administrator’s Guide}.

   c. Remove the previous version of SAS Embedded Process before you upgrade to the most recent version.
      
      - To remove the packages from all nodes concurrently, run this command:
        
        \texttt{psh "rpm -e tkindbsrv"}
      
      - To remove the package from each node, run this command on each node:
        
        \texttt{rpm -e tkindbsrv}

3. (Optional) To confirm that the package was removed, before you deploy the new package, run this command:

   \texttt{psh "rpm -qa tkindbsrv"}

   SAS Embedded Process should not appear on any node.

4. Continue with the installation steps in “Deploying SAS Embedded Process for Teradata” on page 189.
Uninstall SAS Embedded Process for Teradata

The method that you use to uninstall SAS Embedded Process for Teradata is based on your currently installed version of SAS Embedded Process. To uninstall, follow the upgrade instructions except for the final step, which is about continuing the installation.

- “Upgrade from a Previous Teradata Version – SAS 9.4M6 and Later” on page 197
- “Upgrade a Previous Teradata Version – SAS 9.4M3 (July 2015) Prior to SAS 9.4M6” on page 198
- “Upgrade from a Previous Teradata Version – Prior to SAS 9.4M3 (July 2015)” on page 200

To find out when your version of SAS Embedded Process was released, see “Teradata SAS Embedded Process Versions” on page 201.

Teradata SAS Embedded Process Versions

The following table contains the versions of SAS Embedded Process that have been released since September 2013 (SAS 9.3).

Note: You should always use the latest version of SAS Embedded Process. Prior to SAS 9.4M6, SAS 9.4 customers could get only the latest version that was shipped with SAS 9.4. Prior to SAS 9.4M6, SAS Viya customers could get only the latest version that was shipped with SAS Viya 3.x. Starting with SAS 9.4M6, both SAS 9.4 and SAS Viya customers can get the latest version of SAS Embedded Process at any time.

<table>
<thead>
<tr>
<th>Date</th>
<th>SAS Release</th>
<th>SAS Embedded Process File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 2019</td>
<td>SAS Viya 3.5</td>
<td>sas-sepcoretera-17.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td>November 2018</td>
<td>SAS 9.4M6</td>
<td>sas-sepcoretera-16.0.0-date.timestamp.suse.x86_64.rpm</td>
</tr>
<tr>
<td>July 2018</td>
<td>SAS Viya 3.4</td>
<td>sepcoretera-15.00000-1.sh</td>
</tr>
<tr>
<td>December 2017</td>
<td>SAS Viya 3.3</td>
<td>sepcoretera-14.00000-1.sh</td>
</tr>
<tr>
<td>September 2017</td>
<td>SAS 9.4M5</td>
<td>sepcoretera-13.00000-1.sh</td>
</tr>
<tr>
<td>Date</td>
<td>SAS Release</td>
<td>SAS Embedded Process File Name</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>November 2016</td>
<td>SAS 9.4M4</td>
<td>sepcoretera-12.00000-1.sh</td>
</tr>
<tr>
<td>August 2014</td>
<td>SAS 9.4M2</td>
<td>tkindbsrv-9.4_M2-n_lax.sh</td>
</tr>
<tr>
<td>December 2013</td>
<td>SAS 9.4M1</td>
<td>tkindbsrv-9.4_M1-n_lax.sh</td>
</tr>
<tr>
<td>September 2013</td>
<td>SAS 9.3</td>
<td>tkindbsrv-9.33-n_lax.sh</td>
</tr>
</tbody>
</table>
Appendixes

Appendix 1
SAS Mirror Manager for Linux .......................... 205

Appendix 2
SAS Mirror Manager for Windows ....................... 211

Appendix 3
SAS Embedded Process Environment Variables .......... 217
SAS Mirror Manager and the Mirror Repository – Overview and Requirements for Linux

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all the SAS Embedded Process deployments.

Consider the requirements for your mirror repository:

- SAS Mirror Manager can be used to put the files in several locations, such as on a web server that serves the files up by HTTP or on a shared NFS mount.
- The default location for the files that SAS Mirror Manager downloads is the $HOME/sas_repos directory of the installation user. Ensure that the default location or the location that you select has adequate space. Also ensure that the machine where the mirror repository will be located has adequate space. SAS Embedded Process requires approximately 300 MB of space.

Creating a Mirror Repository for Linux

To create a mirror repository with SAS Mirror Manager:
The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine on which you plan to launch your deployment.

Download SAS Mirror Manager from the SAS Mirror Manager download site to the machine on which you want to put your mirror repository.

This step requires internet connectivity. If you receive warnings or errors regarding connectivity, you should work with your system administrator to correct them.

Note: If you use Internet Explorer to download the Linux or Macintosh version, save the file as a.tgz file instead of a.gz file.

Uncompress the downloaded file using your preferred method.

Here is an example of a Linux command to unpack SAS Mirror Manager:

tar -xvzf /path-to-mirror-manager-download/mirrormgr-linux.tgz

Run the following command to create a mirror repository:

mirrormgr mirror --deployment-data path-to-deployment-zip-file-from-SOE <options>

For more information about the options that are available, see "Options for the mirrormgr mirror Command" on page 208.

By default, the repositories are placed in the sas_repos directory in the installation user’s home directory. If you want to put them in another location, use the --path option followed by the full directory location of the mirror destination. This guide refers to that location as sas_repos. However, if you choose a different location, replace instances of sas_repos in this guide with the actual location that you select. Use the --latest option to download the most recent files. Here is an example.

mirrormgr mirror --deployment-data path-to-deployment-zip-file-from-SOE
  --path location-of-mirror-repository --latest

The sas_repos directories are organized as follows:

- The entitlements.json file is a list of the repositories to which you are entitled.
- The location_group_declarations.json file and the sasmd directory contain data that is used by the SAS Orchestration CLI to create the order-specific tools for your deployment.
- Any remaining directories are the software repositories that are organized by native deployment tools:
  - The repository repos contains yum files for Linux.
  - The repository deb contains APT files for Debian.
  - The repository bosh contains BOSH releases for BOSH.

(Optional) After the initial download is complete, copy the file structure to a web server or shared NFS mount. The destination machine does not have to be connected to the internet.

You can use tools like rsync and scp to copy the files. Here is a typical rsync command:
Using a Proxy

If your environment requires a proxy, and a proxy has already been set up, SAS Mirror Manager commands can be used (as described in the preceding section). However, if a proxy has not been set up in your environment, you can add an environment variable to the SAS Mirror Manager command line to specify the location of the proxy.

The environment variable that you use should be appropriate for the target of the query that goes through the proxy: HTTP or HTTPS.

Here is the syntax for the setup of the environment variable for an HTTP site:

```bash
http_proxy=http://user-name:password@internet-proxy-server-FQDN:proxy-port
```

Here is the syntax for the setup of the environment variable for an HTTPS site:

```bash
https_proxy=https://user-name:password@internet-proxy-server-FQDN:proxy-port
```

Here is an example of the setup of the environment variable for HTTP and the run command that are used together. For readability, the command occupies several lines, but it should be entered as a single command on one line.

```bash
http_proxy=http://proxyid:password@proxy.company.com:443
/opt/sas/viya/home/bin/mirrormgr mirror
   --deployment-data SAS_Viya_deployment_data.zip
   --platform x64-redhat-linux-6 --path sas_repos --latest
```

If you use HTTPS, the run command for SAS Mirror Manager might also require the `--cacert` option, which specifies the location of the certificate that the proxy must use. The proxy certificate will be one that your company manages. Here is an example of the setup of the environment variable for HTTPS and the run command that are used together. For readability, the command occupies several lines, but it should be entered as a single command on one line.

```bash
https_proxy=https://proxyid:password@proxy.company.com:3129
/opt/sas/viya/home/bin/mirrormgr mirror
   --deployment-data SAS_Viya_deployment_data.zip
   --platform x64-redhat-linux-6 --path sas_repos
   --cacert ../proxycert.crt --latest
```

Note: In most cases, if you are trying to reach a SAS repository, you should use the HTTPS environment variable since the SAS repository is on an HTTPS site.

For more information about the command options, see “Options for the mirrormgr mirror Command” on page 208.
Specify a Distribution-Specific Subset of Files

To retrieve only the files for the Linux distribution that is being used:

```
mirrormgr mirror --deployment-data path-to-deployment-zip-file-from-SOE --path location-of-mirror-repository --platform Linux-distribution-value --latest
```

Here are the values that can be used for the `--platform` option for Linux:

- Use `x64-redhat-linux-6` for all supported versions of Red Hat Enterprise Linux and its equivalent such as Oracle Linux.

  Note: You can install an RPM file from a Red Hat Enterprise Linux 6 repository on a server that is running Red Hat Enterprise Linux 7.

- Use `x64-suse-linux-12` for all supported versions of SUSE Linux.

Specify a Log Location

The default location for the logs for SAS Mirror Manager is `user-home-directory/.local/share/mirrormgr/mirrormgr.log`.

To specify an alternative log location:

```
mirrormgr mirror --deployment-data path-to-deployment-zip-file-from-SOE --path location-of-mirror-repository --log-file location-of-mirror-repository/mirrormgr.log --latest
```

Note: SAS recommends that you store the logs in the same location as the mirror repository.

Options for the mirrormgr mirror Command

The following table lists the options for the mirrormgr mirror command.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--cacert &quot;location&quot;</td>
<td>The CA certificate for communicating with the upstream repository server. The default is &quot;${HOME}/.config/mirrormgr/ca.pem&quot;.</td>
</tr>
<tr>
<td>--cert &quot;location&quot;</td>
<td>The location of the certificate for authenticating with the upstream repository server. The default is &quot;${HOME}/.config/mirrormgr/cert.pem&quot;.</td>
</tr>
<tr>
<td>--deployment-data &quot;location&quot;</td>
<td>The deployment data archive location.</td>
</tr>
<tr>
<td>-h, --help</td>
<td>Help for the mirror.</td>
</tr>
<tr>
<td>--httpdebug</td>
<td>Enables debugging of HTTP requests to the log file.</td>
</tr>
<tr>
<td>--latest</td>
<td>Mirrors only the latest versions of content.</td>
</tr>
<tr>
<td>--no-verify</td>
<td>Skips the verification steps during the download.</td>
</tr>
<tr>
<td>-p, --path &quot;location&quot;</td>
<td>The path to which repositories are downloaded. The default is &quot;${HOME}/sas_repos&quot;.</td>
</tr>
<tr>
<td>--platform string</td>
<td>Filters repositories to mirror by platform.</td>
</tr>
<tr>
<td></td>
<td>Here are the string values that can be used for the --platform option for Linux:</td>
</tr>
<tr>
<td></td>
<td>* x64-redhat-linux-6 for all supported versions of Red Hat Enterprise Linux and its equivalent such as Oracle Linux.</td>
</tr>
<tr>
<td></td>
<td>* x64-suse-linux-12 for all supported versions of SUSE Linux.</td>
</tr>
<tr>
<td></td>
<td>* x64-ubuntu-linux-14 for all supported versions of Ubuntu Linux.</td>
</tr>
<tr>
<td>--remove-old</td>
<td>If set with --latest, it removes any older packages from the local mirror.</td>
</tr>
<tr>
<td>-r, --repo strings</td>
<td>The list of repositories to mirror. If not specified, all repositories are mirrored. Note: --type is required if --repo is used.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>-t, --type string</td>
<td>Mirrors repositories of this type (yum or APT)</td>
</tr>
<tr>
<td></td>
<td>Note: --type is required if --repo is used.</td>
</tr>
<tr>
<td>--url &quot;url&quot;</td>
<td>The base URL of upstream repositories. The default is &quot;<a href="https://ses.sas.download/ses">https://ses.sas.download/ses</a>&quot;.</td>
</tr>
<tr>
<td>--workers int</td>
<td>The number of download threads to use. The default value is 8.</td>
</tr>
</tbody>
</table>

The following options are inherited from the parent command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--config &quot;location&quot;</td>
<td>The configuration file location. The default is &quot;$HOME/.config/mirrormgr/mirrormgr.yaml&quot;.</td>
</tr>
<tr>
<td>-d, --debug</td>
<td>Enables debug level logging.</td>
</tr>
<tr>
<td>--log-file &quot;location&quot;</td>
<td>The log file location. The default is &quot;$HOME/.local/share/mirrormgr/mirrormgr.log&quot;.</td>
</tr>
<tr>
<td></td>
<td>Note: SAS recommends that you store the logs in the same location as the mirror repository.</td>
</tr>
</tbody>
</table>
Create a Mirror Repository – Windows

SAS Mirror Manager and the Mirror Repository – Overview and Requirements for Windows

SAS Mirror Manager is a command-line utility for synchronizing a collection of SAS software repositories. Its primary use is to create and manage mirror repositories for software deployment. A mirror repository is required for all the SAS Embedded Process deployments.

Consider the requirements for your mirror repository:

- SAS Mirror Manager can be used to put the files in several locations, such as on a web server that serves the files via HTTP or on a shared NFS mount point.

- The default location for the files that SAS Mirror Manager downloads is the C:\Users\user-ID\sas_repos directory. Ensure that the default location or the location that you select has adequate space. Also ensure that the machine where the mirror repository is located has adequate space. SAS Embedded Process requires approximately 300 MB of space.

Creating a Mirror Repository for Windows

To create a mirror repository with SAS Mirror Manager:
The Software Order Email (SOE) indicated that you should save the SAS_Viya_deployment_data.zip file attachment. If you have not already done so, save that file now to a directory on the machine on which you plan to launch your deployment.

Download SAS Mirror Manager from the SAS Mirror Manager download site to the machine on which you want to put your mirror repository. This step requires internet connectivity. If you receive warnings or errors regarding connectivity, you should work with your system administrator to correct them.

Note: The file is saved as a ZIP file.

Unzip the downloaded file using your preferred method.

Run the following command to create a mirror repository:

```
mirrormgr.exe mirror --deployment-data path-to-deployment-zip-file-from-SOE <options>
```

For more information about the options that are available, see "Options for the mirrormgr mirror Command" on page 208.

Note: If you have an HTTPS proxy, you might also need the --cacert option, which indicates the location of the certificate the proxy will have to use. The proxy certificate will be one that your company manages.

By default, the repositories are placed in C:\Users\user-ID\sas_repos. If you want to put them in another location, use the --path option followed by the full directory location of the mirror destination. This guide will refer to that location as \sas_repos. However, if you choose to use a different location, replace instances of \sas_repos used in this guide with the actual location that you select.

The sas_repos directories are organized as follows:

- The entitlements.json file contains a list of the repositories to which you are entitled.
- The location_group_declarations.json file and the sasmd directory contain data that is used by the SAS Orchestration CLI to create the order-specific tools for your deployment.
- Any remaining directories are the software repositories that are organized by native deployment tools:
  - The repository repos contains yum files for Linux.
  - The repository win contains MSI files for Windows.
  - The repository deb contains APT files for Debian.
  - The repository bosh contains BOSH releases for BOSH.

(Optional) After the initial download is complete, copy the file structure to a web server or shared NFS mount. The destination machine does not have to be connected to the internet.
Using a Proxy

If your environment requires a proxy, and a proxy has already been set up, SAS Mirror Manager commands can be used (as described in the preceding section). However, if a proxy has not been set up in your environment, you can add an environment variable to the SAS Mirror Manager command line to specify the location of the proxy.

The environment variable that you use should be appropriate for the target of the query that goes through the proxy: HTTP or HTTPS.

Here is the syntax for the setup of the environment variable for an HTTP site:

```
set http_proxy=http://user-name:password@internet-proxy-server-FQDN:proxy-port
```

Here is an example:

```
set http_proxy=http://proxyid:password@proxy.company.com:443
```

Here is the syntax for the setup of the environment variable for an HTTPS site:

```
set https_proxy=http://user-name:password@internet-proxy-server-FQDN:proxy-port
```

Here is an example:

```
set https_proxy=http://proxyid:password@proxy.company.com:3129
```

If you use HTTPS, the run command for SAS Mirror Manager might also require the `--cacert` option, which specifies the location of the certificate that the proxy must use. The proxy certificate will be one that your company manages.

---

**Note:** In most cases, if you are trying to reach a SAS repository, you should use the HTTPS environment variable since the SAS repository is on an HTTPS site.

---

For more information, see "Options for the mirrormgr mirror Command" on page 214.

Specify a Log Location

The default location for the logs for SAS Mirror Manager is `user-home-directory\.local\share\mirrormgr\mirrormgr.log`.

To specify an alternative log location:

```
mirrormgr.exe mirror --deployment-data path-to-deployment-zip-file-from-SOE
                       --path location-of-mirror-repository
                       --log-file location-of-logs\mirrormgr.log --latest
```

**Note:** SAS recommends that you store the logs in the same location as the mirror repository.
### Options for the mirrormgr mirror Command

The following table lists the options for the mirrormgr mirror command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--cacert &quot;location&quot;</td>
<td>The CA certificate for communicating with upstream repository server. The default is &quot;$HOME.config\mirrormgr\ca.pem&quot;</td>
</tr>
<tr>
<td>--cert &quot;location&quot;</td>
<td>The location of the certificate for authenticating with upstream repository server. The default is &quot;$HOME.config\mirrormgr\cert.pem&quot;</td>
</tr>
<tr>
<td>--deployment-data &quot;location&quot;</td>
<td>The deployment data archive location.</td>
</tr>
<tr>
<td>-h, --help</td>
<td>Help for mirror.</td>
</tr>
<tr>
<td>--httpdebug</td>
<td>Enables debugging of HTTP requests to log file.</td>
</tr>
<tr>
<td>--latest</td>
<td>Mirrors only the latest versions of content.</td>
</tr>
<tr>
<td>--no-verify</td>
<td>Skips the verification steps during the download.</td>
</tr>
<tr>
<td>-p, --path &quot;location&quot;</td>
<td>Path to which repositories are downloaded. The default is &quot;$HOME\sas_repos&quot;.</td>
</tr>
<tr>
<td>--platform string</td>
<td>Filter repositories to mirror by platform. Here are the values that can be used for the --platform option:</td>
</tr>
<tr>
<td></td>
<td>- Use <code>x64-redhat-linux-6</code> for all supported versions of Red Hat Enterprise Linux and its equivalent such as Oracle Linux.</td>
</tr>
<tr>
<td></td>
<td>- Use <code>x64-suse-linux-12</code> for all supported versions of SUSE Linux.</td>
</tr>
<tr>
<td></td>
<td>- Use <code>x64-ubuntu-linux-14</code> for all supported versions of Ubuntu Linux.</td>
</tr>
<tr>
<td>--remove-old</td>
<td>If set with --latest, it removes any older packages from the local mirror.</td>
</tr>
</tbody>
</table>
## Options for the mirrormgr mirror Command

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-r, --repo strings</td>
<td>The list of repositories to mirror. If not specified, all repositories are mirrored. Note: --type is required if --repo is used.</td>
</tr>
<tr>
<td>-t, --type string</td>
<td>Mirror repositories of this type (yum or APT). Note: --type is required if --repo is used.</td>
</tr>
<tr>
<td>--url &quot;url&quot;</td>
<td>The base URL of upstream repositories. The default is &quot;<a href="https://ses.sas.download/ses">https://ses.sas.download/ses</a>&quot;.</td>
</tr>
<tr>
<td>--workers int</td>
<td>The number of download threads to use. The default value is 8.</td>
</tr>
</tbody>
</table>

The following options are inherited from the parent command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--config &quot;location&quot;</td>
<td>The configuration file location. The default is &quot;$HOME.config\mirrormgr\mirrormgr.yaml&quot;.</td>
</tr>
<tr>
<td>-d, --debug</td>
<td>Enables debug level logging.</td>
</tr>
</tbody>
</table>
| --log-file "location" | The log file location. The default is "$HOME\.local\share\mirrormgr\mirrormgr.log"  
Note: SAS recommends that you store the logs in the same location as the mirror repository. |
SAS Embedded Process Environment Variables

Dictionary

SAS_INSTALL_EP Environment Variable

Specifies whether the SAS Embedded Process cartridge file for Netezza is installed and registered when the Netezza RPM file is installed.

Valid in: Command line
Used by: Netezza SAS Embedded Process deployment
See: “Deploying SAS Embedded Process for Netezza” on page 135

Syntax

**SAS_INSTALL_EP** YES | NO

Required Arguments

**YES**

specifies that the SAS Embedded Process cartridge file for Netezza is created, installed, and registered when the Netezza RPM file is installed. This is the default value.

Example

```bash
export SAS_INSTALL_EP=yes
```

**NO**

specifies that the SAS Embedded Process cartridge file for Netezza is created but not installed or registered when the Netezza RPM file is installed

Example

```bash
export SAS_INSTALL_EP=no
```