SAS® Event Stream Processing 4.1: Deployment Guide

Introduction
About This Guide .................................................. 2
What’s New in SAS Deployment ............................ 2
Support for Upgrades ............................................. 2
Contact SAS Technical Support ............................ 3

System Requirements
Hardware Requirements ......................................... 4
Operating System Requirements ............................ 4
Software Requirements ........................................... 6
User Accounts ....................................................... 6

Installing SAS Event Stream Processing
Deploy with Yum .................................................. 8
Set Environment Variables ..................................... 8
Start SAS Event Stream Processing Studio ............ 9
View Deployment Logs .......................................... 9
Uninstall SAS Event Stream Processing ................ 10

Post-Installation Configuration
Directory Structure and Permissions ...................... 12
Change the Default Port ....................................... 12
Enable Database Connectivity ............................... 12
Configure Logging .................................................. 13

Validating the Deployment
Verify RPM Packages ............................................. 14
Error Indicators ....................................................... 14

Next Steps ............................................................ 16
Introduction

About This Guide
SAS Event Stream Processing enables developers to build applications that can quickly process and analyze a large number of continuously flowing events in real time. The deployment installs the programming tools that are required to build and execute event stream processing applications.

SAS Event Stream Processing 4.1 is the first release to include compatibility with the SAS Viya platform. However, even though SAS Event Stream Processing now uses the SAS Viya deployment tools and process, it is a stand-alone product that can coexist with either SAS Viya or SAS 9.4.

Use this guide to deploy SAS Event Stream Processing in your environment.

To use this guide successfully, you should have a working knowledge of the Linux operating system and basic commands.

What’s New in SAS Deployment

SAS Repositories
To ensure that you deploy the latest software, SAS provides the SAS Event Stream Processing software in repository packages that are maintained by SAS. Specifically, the software is packaged in the RPM Package Manager (RPM) format, which simplifies installation, uninstallation, and upgrade tasks. Each time you deploy or update your software, you automatically receive the latest RPM packages that are available.

Note: The RPM-based deployment model works with repositories that are native to your operating system. As a result, a SAS Software Depot is no longer required in your environment.

Industry Standard Tools
You can now deploy SAS Event Stream Processing with tools that are designed for deploying and updating software on Linux operating systems. SAS Viya deployment takes advantage of yum, a software package manager for Linux operating systems. Yum commands are used for secure access to RPM packages and for deploying and updating software in your environment.

Note: The SAS Deployment Wizard and the SAS Deployment Manager that supported SAS 9.4 are not used to install and configure SAS Event Stream Processing 4.1.

One Deployment Guide
This guide includes all the information that is needed to deploy a working environment: system requirements, installation instructions, and information about post-installation steps. In previous releases, this information was provided in separate documents.

Support for Upgrades
The migration of models and data that you generated from a previous release of SAS Event Stream Processing is supported on a limited basis. If you plan to import files from SAS Event Stream Processing 3.1 or 3.2, be aware of the following issues:
Multiple XML elements in 4.1 have changed since 3.1 or 3.2. You must replace the elements that differ.
Opening a legacy project in SAS Event Stream Processing Studio does not automatically upgrade your XML code to a valid format.
You can use the dfesp_xml_validate script to locate elements that are no longer valid so that you can replace them with the new elements.

Any C++ code that you used with a previous release of SAS Event Stream Processing includes a function that must be replaced with a newer function.

The default date format of %Y-%m-%d %H:%M:%S for CSV timestamp and datetime fields is no longer valid. The new ESP_DATETIME fields contain a 64-bit integer that represents seconds since UNIX epoch. The new ESP_TIMESTAMP fields contain a 64-bit integer that represents microseconds since UNIX epoch.
In addition, you can no longer specify an alternative date format when initializing a SAS Event Stream Processing engine. To pass CSV events using an alternative date format, that format must now be specified on the connector or adapter that is the source or sink of CSV data. All connectors and adapters that support CSV include an optional dateformat parameter for this purpose.

Contact SAS Technical Support
If you need assistance with deploying your software, it is important that only SAS support personnel call the Technical Support Division.

For US and Canadian customers, support is provided from the corporate headquarters in Cary, North Carolina. You can call (919) 677-8008, Monday through Friday.

Customers outside of the US can obtain local-language technical support through the local office in their countries. Customers in these locations should contact their local office for specific support hours. See support.sas.com/techsup/contact/index.html for contact information for local offices.
Before you call, explore the SAS Support website at support.sas.com/techsup/. This site offers access to the SAS Knowledge Base, as well as discussion forums, Technical Support contact options, and other support materials that might answer your questions.
System Requirements

Hardware Requirements

SAS Event Stream Processing can be installed as a stand-alone product. It can also coexist with either SAS Viya or with SAS 9.4.

A single machine for the SAS Event Stream Processing components (SAS Event Stream Processing Engine, the web application server, and SAS Event Stream Processing Studio) is the minimum requirement. SAS Event Stream Processing can be deployed on a redundant machine for failover, or it can be distributed across multiple machines. On-premises deployments as well as cloud deployments are supported. You can also deploy the software on the compute layer of a Hadoop cluster, or even at the edge (on a gateway node) of a Hadoop cluster.

SAS Event Stream Processing software is installed in the `/opt/sas/viya` directory. Additional space for logs is also required in `/var/log/sas/viya`. The amount of space that is required depends on the specified logging level. However, the minimum amount of disk space that is required for the installation before accounting for logging is 1 GB.

The following table describes a standard set of specifications for a machine where SAS Event Stream Processing Engine is deployed:

Requirements for SAS Event Stream Processing Components

<table>
<thead>
<tr>
<th>Item</th>
<th>Recommended Minimum Level*</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>4 cores (x86 architecture)</td>
</tr>
<tr>
<td></td>
<td>Intel Xeon chip set with a minimum speed of 2.6 GHz</td>
</tr>
<tr>
<td>Memory</td>
<td>64 GB of RAM</td>
</tr>
<tr>
<td></td>
<td>Memory clock speed of 1600 MHz</td>
</tr>
<tr>
<td>Disk Space and</td>
<td>10 GB</td>
</tr>
<tr>
<td>Speed</td>
<td>10,000 RPM</td>
</tr>
</tbody>
</table>

*The bare minimum requirements for an installation of SAS Event Stream Processing are 2 cores, 4 GB of memory, and 1 GB of disk space. However, a minimum configuration is not recommended.

An additional machine can be used as a thin client from which end users can access the user interface for SAS Event Stream Processing Studio. This machine requires minimal processing power and storage space and can run on Windows or UNIX.

Operating System Requirements

Supported Operating Systems

The following operating systems are supported:

- Red Hat Enterprise Linux versions 6.7 (64-bit) and later within 6.x
- Red Hat Enterprise Linux versions 7.1 and later within 7.x
Previous releases of SAS Event Stream Processing included support for Microsoft Windows. SAS intends to restore this support in a future release of SAS Event Stream Processing.

**Linux Prerequisites**

If you have enabled Security-Enhanced Linux (SELinux) in your environment, you must also enable permissive mode on all of the machines on which you install SAS software. SELinux is not supported by SAS Viya.

To find out whether SELinux is enabled on your system, run the following command:

```
sudo sestatus
```

If the mode that is returned is not permissive, run the following command to enable permissive mode:

```
sudo setenforce 0
```

The SAS deployment architecture requires the operating system to be registered with the Red Hat Network. Registration enables you to receive periodic software updates. For purposes of SAS software deployment, registration also enables yum to download software from SAS repositories. Verify that the machine where you perform the deployment is registered and that your subscription has been activated.

The typical Linux installation includes all of the packages that SAS requires. Problems can occur if default packages were removed from the base operating system (for example, X11 libraries and system utilities). The following libraries are required:

- `libXp`
- `libXmu`
- GLIBC 2.12
- the `numactl` package
- the X11/Xmotif (GUI) packages

**Additional Linux Requirements**

The Event Stream Processing Engine libraries were built using gcc-4.4.7-16 and the Boost library 1.58. The Boost library 1.58 is automatically installed along with SAS Event Stream Processing. The libraries are compiled using the following compiler options:

```
-D_REENTRANT
-D_THREAD_SAFE
```

All of the SAS Event Stream Processing applications that you build with SAS Event Stream Processing Studio must also use the same compiler options.

The Event Stream Processing 4.1 libraries have been built using gcc-4.4.7-16 on Red Hat Enterprise Linux Server release 6.7 using libc-2.12.so, libstdc++.so.6.0.13 and libgcc_s-s-4.4.7-20120601.so.1

**SAS Support for Alternative Operating Systems**

Some variants of operating systems are alternatives to the list of officially supported environments. These variants are sometimes derived from a supported distribution’s source code that might become part of a future release of a supported distribution.

SAS support for an alternative operating system distribution is limited to installation or functional issues. In addition, SAS software uses technologies from various third-party vendors, which might not support these alternative operating systems at the same level as SAS software. Any attempt to re-create a customer’s scenario at SAS is done on an officially supported operating system distribution and third-party vendor software stack. If SAS is unable to reproduce the problem, customers must perform further diagnostics on their own in
order to isolate the problem, up to and including reproducing the problem on a supported operating system
distribution and third-party vendor software stack.

If you use an alternative operating system, you must have the appropriate skills to resolve differences between
the supported operating system and the alternative operating system. By using an alternative operating system,
you acknowledge that you can resolve the differences inherent in that alternative system. These restrictions do
not apply to virtual applications supplied by SAS.

Software Requirements

Java Requirements
The Java Runtime Environment (JRE) must be installed on each machine where you install SAS Event Stream
Processing components. The following versions are supported:

- Oracle JRE SE version 8.x
- OpenJDK version 1.8.x

Note: This open-source version of Java is included with Red Hat Enterprise Linux.

Previous releases of SAS provided a private JRE in the deployment package. In contrast, SAS Viya supports
some alternative versions of the JRE or JDK and does not include Java in the installation package. However, the
deployment process includes a prerequisite check to detect a preinstalled version of Java that meets or exceeds
the requirements. Multiple locations, including /user/java and the definition for the JAVA_HOME environment
variable, are checked. If the required version of Java is located, it is used. Otherwise, the deployment process
attempts to automatically install the OpenJDK. If the installation of the OpenJDK fails, a message is displayed
that instructs you to install the JRE.

Web Browsers
SAS Event Stream Processing Studio includes some advanced user interface features, which require one of the
following web browsers:

- Google Chrome 47.0 or a later version
- Mozilla Firefox 38.0 or a later version

Note: Firefox 45.3 or a later version is required to use the Streamviewer component.

If you cannot install one of the supported web browsers for use with SAS Event Stream Processing, be aware of
possible unexpected user interface behavior. Because session cookies are required in order to maintain session
state, be sure to enable cookies in your browser.

User Accounts
Verify the following prerequisites before you start the deployment:

- Administrator privileges for the Linux machine where you are launching the SAS software deployment are
required.

- The user account that you are using for the deployment must have super user (sudo) access. To verify that
the user ID is included in the sudoers file, run the following command:

  sudo -v

To verify your sudoers privileges, run the following command:

  sudo -l

Note: The ability to start a shell (with the SHELL entry in some sudoers files) as root is not required.
During the software deployment, one required user account (sas) and one group (also named sas) are created for you unless they already exist. Because the sas account is required for the SAS Event Stream Processing Studio component to run during normal product operation, you must not delete it or change its name. It does not run as root. If you must log on to this account, use sudo to access it.

The following table describes the predefined sas user account:

<table>
<thead>
<tr>
<th>Account Name and Group</th>
<th>Parameters</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| sas; member of sas group | UID: 1002  
Group ID: 1001  
Non-login service account without user restrictions.  
No password; can add password after installation if desired.  
Password does not expire.  
Default user name is required until the installation is complete. Any post-installation changes to this account do not prevent future software updates that use SAS RPM packaging. | Required for the installation.  
The installation process sets user and group ownership permissions on all of the installation files. This user must exist to enable ownership.  
After the installation has completed, this user account enables required components to run, including the web application server for SAS Event Stream Processing Studio. |
Installing SAS Event Stream Processing

Deploy with Yum

Use the procedures in this section to deploy your SAS software using yum.

Run the Deployment Script

When you order SAS software, SAS sends a Software Order Email (SOE) to your business or organization. Your SOE includes information about the software order, including several file attachments. The following files are required for deployment:

- the license file
- certificates that enable access to your software
- the customized_deployment_script.txt file, which contains customized commands that are required for accessing and downloading software from SAS repositories

The SOE instructed you to save the customized_deployment_script.txt file as a shell script. Take the following steps to install all SAS Event Stream Processing components on the same machine:

1. Use your preferred text editor to open the customized deployment shell script.
2. Add the directory path where you saved the certificates.
   
   Here is an example:
   ```
   CERTDIR=/opt/sas/installfiles
   ```
3. Save and close the revised shell script.
4. Run the script:
   ```
   sudo ./customized_deployment_script.sh
   ```
5. Run the command to install the product user interface component, SAS Event Stream Processing Studio:
   ```
   sudo yum install -y sas-espvm
   ```

As an option, you can install SAS Event Stream Processing Studio on a separate machine, as follows:

1. Use your preferred text editor to open the customized deployment shell script.
2. Remove the command with the following format from the script:
   ```
   yum groupinstall "SAS_Component_Name"
   ```
3. Add the following command to the script, in place of the command that you removed:
   ```
   yum install -y sas-espvm
   ```
4. Save the revised script.
5. Follow the instructions in the SOE to copy the required files to the machine where you want to install SAS Event Stream Processing Studio. These files include the customized_deployment_script.sh file, the license file, and the certificates.
6. Run the script on the target machine:
   ```
   sudo ./customized_deployment_script.sh
   ```
Apply the License

A valid license file is required in order to run any applications that use SAS Event Stream Processing.

Your SOE contained a license file that you were instructed to save. Now you must apply the license file to the local machine by saving it to the default license directory, which is /opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0/etc/license.

1. Locate the license file that you previously saved.

2. Copy the license file to the default license directory.
   
   **Note:** Substitute the actual name of the license file in the command:
   
   ```
   sudo cp license-filename /opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0/etc/license
   ```
   
   **Note:** If you move the license file to a non-default location, SAS Event Stream Processing will not be able to locate it. Be sure to specify a changed location in any engine object that you create.

Set Environment Variables

Set the following environment variables before you start SAS Event Stream Processing:

```bash
export DFESP_HOME=/opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0
export LD_LIBRARY_PATH=$DFESP_HOME/lib:/opt/sas/viya/home/SASFoundation/sasexe
export PATH=$PATH:$DFESP_HOME/bin
```


Depending on the shell environment that you use, you can also add these export commands to your .bashrc file or .profile file to configure the required settings automatically.

Start SAS Event Stream Processing Studio

SAS Event Stream Processing Studio generates XML code that is based on the visual models that you create. It is not automatically started during the installation. Be sure to set the required environment variables first. For more information, see [Set Environment Variables on page 9](#).

1. To start SAS Event Stream Processing Studio, run the following command:

   ```bash
   sudo service sas-viya-espvm-default start
   ```

   A message states that the sas-esp-visualmodeler Micro Service is running. Visual Modeler is the former name of SAS Event Stream Processing Studio.

   The service runs under the sas user account. For more information, see [User Accounts on page 6](#).

2. After you have started the service, open SAS Event Stream Processing Studio from a URL with the following format:

   ```
   http://esp-studio-hostname:port/SASEventStreamProcessingStudio
   ```

   **Note:** For `esp-studio-hostname` and `port`, specify values that are appropriate for your deployment. The default port is 8080. For information about changing the default port, see [Change the Default Port on page 12](#).

3. Before you can open or create a model in SAS Event Stream Processing Studio, you must start the XML factory server. Change directories to the following location:
cd /opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0/bin

4 Run the following command:

dfesp_xml_server -pubsub n -http-admin adminport -http-pubsub pubsubport

The -pubsub argument specifies a port for publish and subscribe actions. Replace n with the appropriate port number.

The -http-admin argument runs the XML server as a factory server that supports the creation of projects. For adminport, specify the port that you want to use for HTTP administration requests.

Note: If you have a project that is predefined, use the -model argument to run the project as a stand-alone engine.

The -http-pubsub argument sets up a publish/subscribe HTTP server that uses the specified port pubsubport. For more information about the XML server, see SAS® Event Stream Processing 4.1: Using the XML Layer.

5 To create the default directory for storing projects that you create or import in SAS Event Stream Processing Studio and to set the permissions, run the following commands:

sudo mkdir -p /opt/sas/viya/config/data/espvm/esp_projects
sudo chown sas:sas /opt/sas/viya/config/data/espvm/esp_projects
sudo chmod g+w /opt/sas/viya/config/data/espvm/esp_projects

6 (Optional) To check the status of SAS Event Stream Processing Studio, run the following command:

sudo service sas-viya-espvm-default status

Note: The message that is returned references sas-esp-visualmodeler.

View Deployment Logs

To view the logs of your yum deployment, run the following commands:

sudo yum history
sudo less /var/log/yum.log

Uninstall SAS Event Stream Processing

Use yum to uninstall your SAS software:

1 Stop the espvm service:

   sudo service sas-viya-espvm-default stop

2 Uninstall the SAS Event Stream Processing software:

   sudo yum groupremove 'SAS Event Stream Processing'

3 Remove the files for SAS Event Stream Processing:

   sudo yum remove sas-espvm

Any files that are required for other SAS software deployments are not removed.

If you have questions about removing a specific file, you might want to consult with a professional. For more information, see Contact SAS Technical Support on page 3.

4 Rename the /opt/sas/viya/ directory to preserve project files for a possible future use:

   sudo mv /opt/sas/viya /opt/sas/viya_numerical-identifier
The *numerical-identifier* should be unique to the SAS software that is being removed. For example, the identifier might correspond to a timestamp.
Post-Installation Configuration

Directory Structure and Permissions

After you install SAS Event Stream Processing, the files for the engine and the authentication package are located in the following directory:

/opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0

The project files that you create are saved in the following directory:

/opt/sas/viya/config/data/espvm/esp_projects

Note: This directory must be created before you import projects from a previous version of SAS Event Stream Processing or create projects with SAS Event Stream Processing Studio. For more information, see Start SAS Event Stream Processing Studio on page 9. Be aware that imported projects are not automatically migrated to the required format to run with SAS Event Stream Processing 4.1. For more information about upgrading from a previous version, see SAS® Event Stream Processing 4.1: What’s New.

Configuration files for adapters and logs are located in the following directory:

/opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0/etc

The basic directory path enables write access per user group, and it is owned by the sas user. To grant permission to users to edit the configuration files, the administrator must add them to the sas UNIX group. For more information, see User Accounts on page 6.

Later, if you use yum to update your deployment, the configuration files are not altered.

For more information about log settings and modification of the log configuration file, see Configure Logging on page 13.

Change the Default Port

(Optional) You can change the port settings for SAS Event Stream Processing Studio. The default port, 8080, is appropriate for most environments.

1. Use your preferred text editor to open the following file for modification:
   
   /opt/sas/viya/home/bin/sas-espvm

2. Locate the following line in the file:
   
   export java_option_server_port="-Dserver.port=8080"

3. Change the default port, 8080, to the appropriate port.

4. Save and close the file.

5. Restart the espvm service:

   sudo service sas-viya-espvm-default stop
   sudo service sas-viya-espvm-default start

Enable Database Connectivity

(Optional) To facilitate database connectivity, perform these configuration tasks after installation has been completed:
Configure Logging

The SAS Event Stream Processing Engine writes event information to a log file in the following location:

/opt/sas/viya/home/SASEventStreamProcessingEngine/4.1.0/etc/esp-logger.xml

By default, the logging level is INFO, and event information is sent to standard output.

You can configure the esp-logger.xml file by enabling (removing comment tags) or disabling (adding comment tags) specific log settings in the configuration section of the file.

For example, to prevent logging from being displayed on the local terminal screen (standard output), comment out the following line in the esp-logger.xml file:

```xml
<! <appender-ref ref="consoleAppender"/> ->
```

As another example, log rollover is disabled (commented out) by default. To enable log rollover, uncomment the following lines in the esp-logger.xml file:

```xml
<appender class="RollingFileAppender" name="timeBasedRollingFileAppender">
  <param name="Append" value="true"/>
  <param name="ImmediateFlush" value="true"/>
  <rollingPolicy class="TimeBasedRollingPolicy">
    <param name="fileNamePattern" value="%S{OSENV.DFESP_HOME}/log/esp-%S{jobid}-%d{yyyy-MM-dd}.log" />
  </rollingPolicy>
  <layout>
    <param name="HeaderPattern" value="Host: '%S{hostname}', Process: '%S{pid}', OS: '%S{os_bits} %S{os_family}', OS Release: '%S{os_release}"">
    <param name="ConversionPattern" value="%d; %-5p; %t; %c; (%F:%L); %m"/>
  </layout>
</appender>
```

The SAS Event Stream Processing Engine produces a new log file at midnight each day. The date stamp is applied to the filename in the format \textit{PID-yyy-MM-dd}, and each file is saved in the same directory as the esp-logger.xml configuration file. The \textit{PID} is an auto-generated process identifier. Here is an example of a filename:

/var/log/sas/viya/esp/esp-PID-2016-08-25

Restart the server to apply the changes to the esp-logger.xml file. If you do not change any log settings, SAS Event Stream Processing behavior is the same as it has been in previous releases. For more information about logging, see \textit{SAS Event Stream Processing 4.1: Troubleshooting}. 
Validating the Deployment

Verify RPM Packages

Your SAS software was delivered in RPM (Red Hat Package Manager) packages. Use this basic command to verify an RPM package:

```
rpm -Vv package-name
```

For example, to verify the contents of the sas-esp package, run the following command:

```
rpm -Vv sas-esp
```

You can also create a for loop command for verifying multiple packages that share a common naming convention. For example, to verify all packages whose names begin with `sas-`, use the following query:

```
for i in $(rpm -qa | grep -e "^sas-");do rpm -Vv $i;done
```

A successful verification shows the list of files that make up the RPM and with no error indicators, as follows:

```
# rpm -Vv sas-esp
........... /opt/sas/viya/home/lib/esp/sas-init-functions
```

An unsuccessful verification provides error indicators that are next to the filename. Here is an example:

```
# rpm -Vv sas-esp
package sas-esp is not installed
```

The error indicators are shown in the following format:

```
SM5DLUGT c
```

In addition, if a file is missing, the error message contains the word “missing”:

```
missing /opt/sas/viya/home/lib/esp/sas-init-functions
```

Error Indicators

RPM verification produces various error indicators. You might see the following error indicators when you perform RPM verification:

- **S** - file size
  - RPM keeps track of file sizes. A difference of even one byte triggers a verification error.

- **M** - file mode
  - The permissions mode specifies access for the file's owner, group members, and others. Two additional bits determine whether a user's group or user ID should be changed if they execute the program that is contained in the file. These bits permit any user to become root for the duration of the program.

- **5** - MD5 checksum
  - The MD5 checksum of a file is a 128-bit number that is mathematically derived from the contents of the file. The MD5 checksum conveys no information about the contents of the original file, but any change to the file results in a change to the checksum. RPM creates MD5 checksums for all files that it manipulates and stores them in its database. If one of these files is changed, the checksum changes and the change is detected by RPM.

- **D** - major and minor numbers
Device character and block files contain a major number. The major number is used to communicate information to the device driver that is associated with the special file. For example, under Linux, the special files for SCSI disk drives should have a major number of 8. The major number for an IDE disk drive's special file should be 3.

**CAUTION!** Any change to a file’s major number could produce unwanted results. RPM tracks these changes.

A file’s minor number is similar to the major number, but conveys different information to the device driver. For disk drives, this information can consist of a unit identifier.

- **L** - symbolic link
  If a file is a symbolic link, RPM checks the text string that contains the name of the symbolically linked file.

- **U** - file owner
  Most operating systems keep track of each file's creator, primarily for resource accounting. Linux and UNIX also use file ownership to help determine access rights to the file. In addition, some files, when executed by a user, can temporarily change the user's ID, normally to a more privileged ID. Therefore, any change of file ownership might have significant effects on data security and system availability.

- **G** - file group
  Similar to file ownership, a group specification is attached to each file. Primarily used for determining access rights, a file's group specification can also become a user's group ID if that user executes the file's contents. Therefore, any changes in a file's group specification are important and should be monitored.

- **M** - modification time
  Most operating systems keep track of the date and time that a file was last modified. RPM keeps modification times in its database.

- **c** - configuration file
  This error indicator is useful for quickly identifying the specified configuration file.
Next Steps

After you install, configure, and verify the deployment, you are ready to begin writing applications that capture and analyze streaming event data in real time.

The next step is to consult the product documentation. Start with *SAS Event Stream Processing 4.1: Overview*, which provides an introduction to product features and explains how to proceed with creating event stream processing models and incorporating them into applications.

You can also set up the optional Streamviewer component. Streamviewer provides a user interface that enables you to display events that are passing through selected window event streams from one or more processing engines. You can display each event as a row in a table or as an element of a graph. Streamviewer has a separate configuration procedure. For a full set of instructions, see *SAS® Event Stream Processing 4.1: Visualizing Event Streams with Streamviewer*.

The product documentation is included in SAS Help Center. A link to all SAS Event Stream Processing documentation is available on the SAS Event Stream Processing product page. All product user documentation is available via single sign-on from the SAS Event Stream Processing user interfaces (SAS Event Stream Processing Studio and Streamviewer). To view secure user documentation (indicated by a lock icon) from the product page, send an email to SAS Technical Support to request an access key.