



SAS[®] Viya[®] 3.5 Administration: Tuning

3.5

This document might apply to additional versions of the software. Open this document in [SAS Help Center](#) and click on the version in the banner to see all available versions.

<i>Tuning: Overview</i>	2
<i>Tuning: Apache HTTP Server</i>	3
Overview	3
Recommendations	3
<i>Tuning: Files Service</i>	6
Overview	6
Recommendations	6
<i>Tuning: Java Runtime Environment on Linux</i>	8
Overview	8
Recommendation	8
<i>Tuning: JDBC Connection Pool</i>	8
Overview	8
Deployment Size Definitions	9
Recommendations	9
<i>Tuning: LDAP Connection Pool on Linux</i>	12
Overview	12
Recommendations	12
<i>Tuning: Operating System</i>	13
Overview	13
Linux	13
Windows	14
<i>Tuning: Report Data Service</i>	15

Overview	15
Recommendations	15
Configuration	15
<i>Tuning: SAS Cache Server</i>	16
Overview	16
Recommendations	17
Configuration	17
<i>Tuning: SAS Cloud Analytic Services</i>	19
Overview	19
Recommendation	19
<i>Tuning: SAS Infrastructure Data Server</i>	19
Overview	19
Recommendations	20
<i>Tuning: SAS Message Broker</i>	22
Overview	22
Recommendations	22
<i>Tuning: SAS Studio 5.2 (Enterprise)</i>	24
Overview	24
Recommendations	24

Tuning: Overview

In SAS Viya, you can tune your environment for performance and scalability. This document includes tuning methodologies and parameters for:

- Apache HTTP Server
- Java Runtime Environment (JRE)
- Java Database Connectivity (JDBC) connection pool
- Lightweight Directory Access Protocol (LDAP) connection pool
- Operating system
- SAS servers and services

Performance requirements are usually identified in terms of transaction response time, number of transactions per second, throughput, resource utilization, total cost per transaction, availability, and more. Scalability often refers to the ability of a component to adapt readily to a greater or lesser intensity of use, volume, or demand, while meeting integral business objectives. The common objective of scaling a component or system is to increase the capacity for growth, increase the speed of the component, improve the efficiency, or shift or reduce the load on the component.

Tuning: Apache HTTP Server

Overview

You can improve the performance of the Apache HTTP Server by configuring other aspects of the web server. For example, to improve performance, rotate log files and configure the Multi-Processing Modules (MPMs). You can also change the time-out interval.

For more information about MPMs, see <http://httpd.apache.org/docs/2.4/mpm.html>.

For more information about deploying Apache httpd, see “Apache httpd” in *SAS Viya for Linux: Deployment Guide*.

Note: If you are deploying on Red Hat Enterprise Linux or Oracle Enterprise Linux 8.x, SAS recommends updating an Apache httpd configuration file to change the active MPM setting from `mod_mpm_event` to `mod_mpm_prefork`. For more information, see [Support for Apache HTTP Server in SAS Viya 3.5](#).

Recommendations

MPM Configuration (Linux)

- 1 For sites with upward of 400 users, it is recommended that you enable the following Apache HTTP modules:

- Apache 2.2: **worker**

In `/etc/sysconfig/httpd`, uncomment the following line:

```
HTTPD=/usr/sbin/httpd.worker
```

- Apache 2.4: **mod_mpm_worker.so**

In `/etc/httpd/conf.modules.d/00-mpm.conf`, *comment* the line ending in `mod_mpm_prefork.so`, and *uncomment* the line ending in `mod_mpm_worker.so`:

```
#LoadModule mpm_prefork_module modules/mod_mpm_prefork.so
LoadModule mpm_worker_module modules/mod_mpm_worker.so
#LoadModule mpm_event_module modules/mod_mpm_event.so
```

- Apache 2.4 on SUSE: **worker**

Verify that the worker MPM is installed. It can be installed with Zypper, using the following command:

```
zypper install apache2-worker
```

Then, edit the `/etc/sysconfig/apache2` file and define the **APACHE_MPM** property.

```
APACHE_MPM="worker"
```

2 Configure the Apache HTTP Server to use the worker MPM as follows:

- For Apache 2.2, modify the `/etc/httpd/conf/httpd.conf` file to adjust worker MPM properties. Add the **ServerLimit** property and change the value for the other properties that are highlighted in the sample file below:

```
# worker MPM
# StartServers: initial number of server processes to start
# MaxClients: maximum number of simultaneous client connections
# MinSpareThreads: minimum number of worker threads which are kept spare
# MaxSpareThreads: maximum number of worker threads which are kept spare
# ThreadsPerChild: constant number of worker threads in each server process
# MaxRequestsPerChild: maximum number of requests a server process serves
<IfModule worker.c>
    ServerLimit      32
    StartServers     10
    MaxClients       1024
    MinSpareThreads  25
    MaxSpareThreads  75
    ThreadsPerChild  32
    MaxRequestsPerChild 0
</IfModule>
```

- For Apache 2.4, add the following configuration block to the existing configuration file. It is recommended that you modify either the `/etc/httpd/conf.modules.d/00-mpm.conf` file or the `/etc/httpd/conf/httpd.conf` file. On SUSE, edit the `/etc/apache2/httpd.conf` file.

```
<IfModule mpm_worker_module>
    ServerLimit      32
    StartServers     10
    MaxRequestWorkers 1024
    MinSpareThreads  25
    MaxSpareThreads  75
    ThreadsPerChild  32
    MaxConnectionsPerChild 0
</IfModule>
```

3 Identify a suitable log rotation strategy and modify the `/etc/httpd/conf/httpd.conf` file to configure the Apache `rotatelogs` tool to perform log rotation. For information about rotation strategies and configuration options, see <http://httpd.apache.org/docs/2.4/programs/rotatelogs.html>.

The following are sample `httpd.conf` file entries for configuring daily log rotation:

```
#ErrorLog logs/error_log
ErrorLog "|/usr/sbin/rotatelogs logs/error_log 86400"

#CustomLog logs/access_log combined
CustomLog "|/usr/sbin/rotatelogs logs/access_log 86400" combined
```

Note: For SUSE, the **ErrorLog** property is defined in the `/etc/apache2/httpd.conf` file. The **CustomLog** property is defined in the `/etc/sysconfig/apache2` file, as **APACHE_ACCESS_LOG**:

```
APACHE_ACCESS_LOG="\\"|/usr/sbin/rotatelog /var/log/apache2/access_log 86400\\" combined"
```

- 4 Restart the Apache HTTP Server by running the following commands:

```
apachectl stop
apachectl start
```

MPM Configuration (Windows)

For sites with upward of 400 users, it is recommended that you apply the following tuning changes to the configuration of the WinNT MPM:

- 1 Edit the `conf\extra\httpd-mpm.conf` file
- 2 Add the `ThreadLimit` directive and increase the `ThreadsPerChild` within the `mpm_winnt_module` conditional block as highlighted in the sample file below:

```
# WinNT MPM
# ThreadsPerChild: constant number of worker threads in the server process
# MaxConnectionsPerChild: maximum number of connections a server process serves
<IfModule mpm_winnt_module>
    ThreadLimit          3072
    ThreadsPerChild      3072
    MaxConnectionsPerChild 0
</IfModule>
```

Change Time-Out Interval

When SAS web applications, such as SAS Visual Data Builder, return HTTP 502 proxy errors, you might have to change the time-out interval for your Apache HTTP Server. Complete the following steps:

- 1 Sign in as the SAS install user (`sas`) or with `sudo` privileges, to the Apache HTTP Server machine.
- 2 Using a text editor, open `/etc/httpd/conf.d/petrichor.conf` file.
- 3 Modify **Timeout** and **Keepalive** to read as follows:

```
Timeout 2400
Keepalive On
```

- 4 Add the following two parameters, and save the file:

```
ProxyTimeout 2400
ProxyBadHeader Ignore
```

- 5 Restart Apache HTTP Server

IMPORTANT The `ProxyTimeout` setting of 2400 seconds (40 minutes) is a recommendation. A longer time-out might be needed depending on your usage of SAS Viya REST APIs.

CAUTION

When you edit the `petrichor.conf` file, be aware that the time-out interval changes might be lost, if the file is upgraded during a maintenance update.

Tuning: Files Service

Overview

By default, the first 5 KB of text in an attachment is extracted and indexed. If you upload an attachment that contains more text, only the configured number of bytes of text are extracted and indexed.

There are three categories for file extraction that are based on the size and type of the file. The size and type are configurable and as an administrator, you can change the default settings. Currently, the default values of these properties are set based on a maximum heap size of 256 MB. The maximum heap size can also be changed. If you increase the file extraction limit, then you must also increase the maximum heap size.

Use SAS Environment Manager to change the value of how much of an attachment is extracted and indexed by editing the `sas.files.extractionLimit` setting. For details, see [“Configuration Properties: How to Configure Services” in SAS Viya Administration: Configuration Properties](#).

Recommendations

Default Configuration Settings

The following table provides the default settings for the Files Service definition:

Table 1 Default Configuration Settings

File Size	Default Limit	Details
Large	10 MB	If a file is larger than 10 MB, only 10 MB of the file is sent for text extraction. The following properties define the file size and type: ■ <code>sas.files.extraction.largeFileSize</code>: 10485760

File Size	Default Limit	Details
		■ <code>sas.files.extraction.largeFileTypes: application/vnd.ms-powerpoint,application/vnd.openxmlformats-officedocument.presentationml.presentation</code>
Medium	5 MB	If a file is larger than 5 MB, only 5 MB of the file is sent for text extraction. The following properties define the file size and type: ■ <code>sas.files.extraction.mediumFileSize: 5242880</code> ■ <code>sas.files.extraction.mediumFileTypes: application/vnd.ms-excel,application/vnd.openxmlformats-officedocument.spreadsheetml.sheet,text/csv</code>
Small	2.5 MB	If a file is larger than 2.5 MB, only 2.5 MB of the file is sent for text extraction. The following properties define the file size and type: ■ <code>sas.files.extraction.smallFileSize: 2621440</code> ■ <code>sas.files.extraction.smallFileTypes: application/msword,application/vnd.openxmlformats-officedocument.wordprocessingml.document,application/pdf</code>

Note: If the file size does not fall under any of the above size categories, the complete stream of the file is sent to Apache Tika for extraction.

Note: A limitation exists with Microsoft Word document files (and possibly Microsoft PowerPoint and Microsoft Excel files) where the partial extraction of content does not work. Word document files fall under the small category. If you send partial Word document files that are greater than 2.5 MB for extraction, Apache Tika does not return any extracted text.

Modify the File Extraction Properties

- 1 From SAS Environment Manager, navigate to the **Definitions** view.
- 2 In the **Definitions** list, select **sas.files.extraction**.
- 3 In the top right corner of the Files service window, click .
- 4 In the Edit `sas.files.extraction` Configuration dialog box, make the required changes to the appropriate file size and file type.
- 5 Click **Save**.

See Also

[“Edit Configuration Instances” in SAS Viya Administration: Configuration Properties](#)

Tuning: Java Runtime Environment on Linux

Overview

The goal of JRE tuning is to improve performance in the services, particularly in the area of memory usage and garbage collection cycles. The goal is to also maximize the number of clients that the SAS web applications can support.

Recommendation

The default JRE tuning options that are applied for each service should be sufficient. However, you might need to limit how much the native memory usage grows for each Java process. To limit the growth, add the following lines to the `viya-installation-directory/etc/sysconfig/sas-javaesnt1/sas-java-services` file:

```
# Limit the number of "malloc arenas" to 1 (default behavior is to use (# of cores * 8))
export MALLOC_ARENA_MAX=1
```

Tuning: JDBC Connection Pool

Overview

In JDBC connection pooling, instead of creating connections every time they are requested, connections are reused. The JDBC connection pool is a collection of database connection objects that are available for reuse. It is maintained by a connection pooling module as a layer on top of the JDBC driver.

Deployment Size Definitions

The suggested configurations in this chapter are based on the following deployment size definitions:

- A small deployment receives more than one concurrent query or update, and creates hundreds of database records daily.
 - A medium deployment receives several concurrent queries or updates (most of which use indexes), and creates thousands of database records per hour.
 - A large deployment receives many concurrent queries or updates, creates thousands of database records per minute, receives many queries to scan tables, and processes complex queries and regular bulk loads.
-

Recommendations

Configure Deployment Size

SAS Viya provides default JDBC connection pool settings for small, medium, and large deployments. By default, all services are configured to use the medium deployment settings.

To configure a different deployment size for a service, complete the following tasks:

- 1 In SAS Environment Manager, click **Configuration** and select the **Definitions** view.
- 2 In the **Definitions** list, select **jvm**.
- 3 Verify that the jvm service does not have an existing configuration. If nothing has been configured yet, a message states that `The selected item has no configurations.`
- 4 If the jvm service has at least one configuration already, select the SAS Viya service whose jvm settings you want to modify by clicking  next to that service.
If the service has no existing configuration, click **New Configuration**.
- 5 Click **+Add Property**.
- 6 In the **Name** field, specify `java_option_springdatasource_default`.
- 7 In the **Value** field, specify `-Dsas.deployment.springdatasource.defaults=size`, where *size* is small, medium, or large.
- 8 Click **Save** in the Add Property window.
- 9 Click **Save** to save the configuration change.
- 10 Restart SAS Viya services.

See Also

- [“Create Configuration Instances” in SAS Viya Administration: Configuration Properties](#)
- [“Start and Stop All Servers and Services” in SAS Viya Administration: General Servers and Services](#)

Data Source Properties

By default, predefined property settings for small, medium, and large deployments are applied. You can override these values. For example, you can set the Preferences Service to use the default property values for a small system, but override the default value of the `spring.datasource.tomcat.maxIdle` property by changing it from 2 to 3.

Override Default Property Values

To change the default property settings, complete the following tasks:

- 1 In SAS Environment Manager, click **Configuration** and select the **Definitions** view.
- 2 In the **Definitions** list, select **spring**.
- 3 Verify that the spring service does not have an existing configuration. If nothing has been configured yet, a message states that The selected item has no configurations.
- 4 If the spring service has a configuration already, select a SAS Viya service whose spring settings you want to modify by clicking  next to that service.

If the spring service has no existing configuration, click **New Configuration**.

Note: Not all services use the default property settings. Instead, those services specify a scaling factor that enables them to have larger pool sizes, based on the deployment size that is specified in the `sas.deployment.springdatasource.defaults` property. For more information, see [“Data Source Scaling Factor” on page 11](#).

- 5 Click **+Add Property**.
- 6 In the **Name** field, specify a property from the [Property Settings Table on page 10](#).
- 7 In the **Value** field, specify the new size that you want to set for the property.
- 8 Click **Save** in the Add Property window.
- 9 Click **Save** to save the configuration setting.

See Also

[“Create Configuration Instances” in SAS Viya Administration: Configuration Properties](#)

Default Data Source Property Settings

The following table provides the default property settings for the JDBC connection pool, based on the deployment size:

Table 2 Property Settings Table

Property	Small Deployment	Medium Deployment	Large Deployment
<code>datasource.tomcat.initialSize</code>	2	2	2
<code>datasource.tomcat.maxActive</code>	6	10	20
<code>datasource.tomcat.maxIdle</code>	2	2	2
<code>datasource.tomcat.minIdle</code>	2	2	2

A service can also provide default files for small, medium, and large deployments in the service resource directory.

Data Source Scaling Factor

Not all services use the default property settings. Instead, those services specify a scaling factor that enables them to have larger pool sizes, based on the deployment size that is specified in the `sas.deployment.springdatasource.defaults` property. For example, the Authorization Service specifies a scaling factor of 10. Therefore, its `maxActive` value is 60 for small, 100 for medium, and 200 for large deployments. For a list of the default property values, see [Table 2 on page 11](#)

Configure the Scaling Factor

To define a scaling factor for a service, complete the following tasks:

- 1 In SAS Environment Manager, click **Configuration** and select the **Definitions** view.
- 2 In the **Definitions** list, select **jvm**.
- 3 Follow the steps that are described above in [“Configure Deployment Size” on page 9](#) to add a new configuration or to modify a service definition to add a configuration property.
- 4 In the **Name** field, specify the `java_option_datasource_factor` property.
- 5 In the **Value** field, specify `-Dsas.datasource.custom.factor=multiplier`, where *multiplier* is the multiplier factor by which the property in the [Property Settings Table on page 11](#) is multiplied.
- 6 Click **Save** in the Add Property window.
- 7 Click **Save** to save the configuration change.
- 8 Restart SAS Viya services.

See Also

- [“Edit Configuration Instances” in SAS Viya Administration: Configuration Properties](#)
- [“Start and Stop All Servers and Services” in SAS Viya Administration: General Servers and Services](#)

Scaling Factor Example

You can specify a multiplier factor for a service by using the `sas.datasource.custom.factor` property. The default value for a property is multiplied by the value that you specify. For example, for a medium deployment, the default value for the `spring.datasource.tomcat.maxActive` property is 10. If you set the multiplier factor to 5, the new `maxActive` value is 50. The factor must be greater than 0. The resulting `maxActive` value is no less than `spring.datasource.tomcat.initialsize` and no more than 200.

Tuning: LDAP Connection Pool on Linux

Overview

The LDAP service provider supports connection pooling. In LDAP connection pooling, the service provider maintains a pool of previously used connections. When a connection is closed or goes to garbage collection, it goes back to the pool to be used again.

By default, no configuration is required for the LDAP service provider to use connection pooling. However, configuration is needed to customize the setting for optimal performance.

Recommendations

- 1 In SAS Environment Manager, edit the **Identities service**.
- 2 Navigate to the **sas.identities.providers.ldap.connection** configuration instance and configure the following:

Table 3 *Properties and Values*

Property	Value
pool.maxActive	30
pool.maxIdle	30

See Also

[“Edit Configuration Instances” in SAS Viya Administration: Configuration Properties](#)

Tuning: Operating System

Overview

A number of configuration changes and variables can be modified in order to tune SAS Viya performance and scalability. The following sections describe how to change the operating system settings that are relevant to SAS Viya post-deployment.

Linux

Recommendations

Tuning TCP/IP

For sites with 400 or more users, SAS recommends the following settings:

- Ensure that IPv6 is enabled.
- Permanently set the SAS-recommended TCP/IP settings by running the following commands:

```
/sbin/sysctl -w net.ipv4.tcp_fin_timeout=30
/sbin/sysctl -w net.core.netdev_max_backlog=3000
/sbin/sysctl -w net.core.somaxconn=3000
/sbin/sysctl -w net.ipv4.tcp_keepalive_intvl=15
/sbin/sysctl -w net.ipv4.tcp_keepalive_probes=5
```

Tuning for SAS Studio

The following options can be modified in the `/etc/sysctl.conf` file, when these conditions exist:

- For sites with upward of 40 concurrently logged-on users, who are running tasks that require rendering of graphs, the SEMMNI parameter should be increased to 4096.
- For sites with upward of 600 logged-on users, increase the PID_MAX parameter to 131072.

Windows

Recommendations

Update Windows Registry

The Windows registry must be updated. Microsoft recommends performing a system backup before editing the registry. To set the SAS recommended parameters, use the REGEDIT command as follows:

- 1 Access the **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\Tcpip\Parameters** registry subkey. Add the DWORD value with a name of **TcpTimedWaitDelay** and a value of 30 (0xle).
- 2 Access the **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\AFD\Parameters** registry subkey. Add the following DWORD values:

Table 4 AFD Service Parameters

Name	Recommended Value
EnableDynamicBacklog	1 decimal
MinimumDynamicBacklog	20 decimal
MaximumDynamicBacklog	1000 decimal
DynamicBacklogGrowthDelta	10 decimal

The recommended values specify the number of connections that you want to be available. These values request a minimum of 20 and a maximum of 1000 available connections. The number of available connections is increased by 10 each time there are fewer than the minimum number of available connections.

- 3 In addition, the user port range should be updated. From a command prompt, run the following commands, based on the version of your internet protocol:

```
netsh int ipv4 set dynamicport tcp start=32768 num=32767
netsh int ipv4 set dynamicport udp start=32768 num=32767
```

The value of *n* indicates the version of your internet protocol and is either 4 or 6.

- 4 Restart Windows.

Update System Configurations

The following list includes general recommendations for configuring Windows systems:

- Disable Windows indexing on any directories that are used by SAS software.
- Set Windows performance settings so that background processes are favored.
- Set the maximum power profile in the system BIOS for all systems, except Intel Sandy Bridge.
- Disable the C1E BIOS setting on Dell systems.

Tuning: Report Data Service

Overview

By default, the heap size for the Report Data service has been set for a small to moderate user workload in an effort to keep the overall memory usage to a minimum. As the user workload increases, you might need to increase the initial and maximum heap sizes for the Report Data service. This helps users avoid experiencing longer transaction response times when viewing a SAS report.

Recommendations

If you increase the initial heap size option (-Xms) for the Report Data service, increase the value so that it is equal to the maximum heap size option (-Xmx). These values can be increased if there is enough demand, but it is recommended that these two options are set to the same value to avoid the overhead of expansion and contraction. The default value for the maximum heap size for the Report Data service is -Xmx1536m. By default, the initial heap size value is not set.

The following recommendations exist for environments with a moderate to large number of users:

- Configure the initial heap size to -Xms2048m.
- Configure the maximum heap size to -Xmx2048m.

Note: Depending on the usage patterns observed and the number of users, you might need to increase these values.

Configuration

Configure Initial Heap Size

- 1 From SAS Environment Manager, navigate to the **All services** view.
- 2 In the **All services** list, select **Report Data service**.

- 3 In the Configuration window, at the top of the content pane, enter `jvm` in the **Filter** field.
- 4 Click .
- 5 In the Edit jvm Configuration window, click **+**.
- 6 In the Add Property window:
 - a In the **Name** field, specify `java_option_xms`.
 - b In the **Value** field, specify the new initial heap size as `-Xmssize`, where *size* is the number of bytes. Append *k* for kilobytes, *m* for megabytes, or *g* for gigabytes (for example, `-Xms60m`).
For more information, see [Java Documentation](#).
- 7 Click **Save**.
- 8 Restart the Report Data service.

Configure Maximum Heap Size

- 1 From SAS Environment Manager, in the Configuration window, click .
- 2 In the Edit jvm Configuration window, click **+**.
- 3 In the Edit jvm Configuration window, update the `java_option_xmx` property with the new maximum heap size value.
- 4 Click **Save**.
- 5 Restart the Report Data service.

See Also

- [“Edit Configuration Instances” in SAS Viya Administration: Configuration Properties](#)
- [“Create Configuration Instances” in SAS Viya Administration: Configuration Properties](#)

Tuning: SAS Cache Server

Overview

An Apache Geode Resource Manager runs inside the SAS Cache Server, monitoring the current heap usage against the current maximum heap that is available. If the SAS Cache Server is configured with different values of initial heap size and maximum heap size, the Resource Manager might detect a premature critical heap situation.

For more information about Geode Resource Manager, see [Managing Heap and Off-heap Memory](#).

Recommendations

The following recommendations exist for SAS Cache Server:

- Configure the initial heap size to `-Xms60m`.
- Configure the maximum heap size to `-Xmx1024m`.
- Disable the `-Dsas.cache.config.overflowEnabled` option.

The heap size values can be increased if the demand on the server is high. By default, they are set to `-Xms60m` and `-Xmx256m`. Depending on the usage patterns observed and the number of users, you might need to increase the value of the maximum heap size option. For example, `-Xmx1000m` is a reasonable value for a multi-tenant deployment with a large number of users. For highly intensive environments, such as SAS Visual Investigator in a multi-tenant environment, you can configure the maximum heap size up to `-Xmx4000m`.

Configuration

Configure Initial Heap Size

- 1 From SAS Environment Manager, navigate to the **All services** view.
- 2 In the **All services** list, select **Cache Server service**.
- 3 In the Configuration window, at the top of the content pane, enter `jvm` in the **Filter** field.
- 4 Click .
- 5 In the Edit jvm Configuration window, click **+**.
- 6 In the Add Property window:
 - a In the **Name** field, specify `java_option_xms`.
 - b In the **Value** field, specify the new initial heap size as `-Xmssize`, where *size* is the number of bytes. Append *k* for kilobytes, *m* for megabytes, or *g* for gigabytes (for example, `-Xms60m`).
For more information, see [Java Documentation](#).
- 7 Click **Save**.
- 8 Restart all SAS Viya services.

Configure Maximum Heap Size

- 1 From SAS Environment Manager, in the Configuration window, click .
- 2 In the Edit jvm Configuration window, click **+**.
- 3 In the Edit jvm Configuration window, update the `java_option_xmx` property with the new maximum heap size value.

- 4 Click **Save**.
- 5 Restart all SAS Viya services.

Disable Overflow Option

- 1 From SAS Environment Manager:
 - a In the Configuration window, at the top of the content pane, enter **overflow** in the **Filter** field.
 - b Click .
 - c In the Edit sas.cache.config Configuration window, scroll to the **overflowEnabled** option and disable it.
 - d Click **Save**.
- 2 On the server:
 - a Stop all SAS Viya services.
 For Red Hat Enterprise Linux 6.7:

```
sudo service sas-viya-all-services stop
```

 For Red Hat Enterprise Linux 7.x or later and SUSE Linux:

```
sudo systemctl stop sas-viya-all-services
```

 For Windows, in Windows Services Manager, right-click *service_name* and select **Stop**.
 - b Remove the files from the SAS Cache Server **/tmp** directory.

```
rm -rf /opt/sas/viya/config/tmp/sas-cacheserver/*
```
 - c Restart all SAS Viya services.
 For Red Hat Enterprise Linux 6.7:

```
sudo service sas-viya-all-services start
```

 For Red Hat Enterprise Linux 7.x or later and SUSE Linux:

```
sudo systemctl start sas-viya-all-services
```

 For Windows, in Windows Services Manager, right-click *service_name* and select **Start**.

See Also

- [“Edit Configuration Instances” in SAS Viya Administration: Configuration Properties](#)
- [“Create Configuration Instances” in SAS Viya Administration: Configuration Properties](#)

Tuning: SAS Cloud Analytic Services

Overview

You can update the SAS Cloud Analytic Services (CAS) configuration file options to control how much CPU a server should allocate for CAS.

Recommendation

A recommended value for `cas.CPUSHARES` for a SAS Viya full deployment on Windows is 70.

See Also

[“Configuration File Options” in SAS Viya Administration: SAS Cloud Analytic Services](#)

Tuning: SAS Infrastructure Data Server

Note: SAS Infrastructure Data Server is not currently supported in a programming-only deployment.

Overview

SAS Infrastructure Data Server provides a transactional store that is used to support SAS Viya. The server is configured automatically during deployment. However, to optimize its performance, it is recommended that you perform the tuning recommendations in this section.

Recommendations

Connection Settings

In SAS Environment Manager, edit the **SAS Infrastructure Data Server** service and modify the following properties to change the number of connections available to client:

Table 5 *Properties and Values*

Configuration Instance	Property	Value
sas.dataserver.conf: common	max_connections For more information, see http://www.postgresql.org/docs/9.1/static/runtime-config-connection.html#RUNTIME-CONFIG-CONNECTION-SETTINGS	1027
sas.dataserver.conf: common	max_prepared_transactions For more information, see http://www.postgresql.org/docs/9.4/static/runtime-config-resource.html	1027
sas.dataserver.pool: common	num_init_children For more information, see http://www.pgpool.net/docs/pgpool-II-3.5.4/doc/pgpool-en.html#NUM_INIT_CHILDREN Note: This setting does not apply to a Windows deployment.	1024

Note: The `max_connections` value should be slightly higher than the `num_init_children` value to allow for direct connections outside pgpool for administrative use, such as backup and recovery.

See Also

“Edit Configuration Instances” in *SAS Viya Administration: Configuration Properties*

ulimit Settings

SAS recommends changing the default ulimit settings for the sas user on Red Hat Enterprise Linux. These settings can be modified in the `/etc/security/limits.conf` file.

```
sas    -    nofile    150000
sas    -    nproc     100000
sas    -    stack     10240
```

Note: You might need to adjust additional Linux operating system settings in order to support these recommended ulimit settings.

Semaphore Settings

SAS recommends changing the default semaphore settings for Red Hat Enterprise Linux and SUSE Linux deployments. These settings can be modified in the `/etc/sysctl.conf` file.

```
kernel.sem=512 32000 100 1024
net.core.somaxconn=2048
```

```
for SEMMSL, SEMMNS, SEMOPM, and SEMMNI
```

For more information, including formulas and minimum values, see <http://www.postgresql.org/docs/9.5/static/kernel-resources.html>.

Note: Changing Linux semaphore settings requires a machine reboot.

Note: You might need to adjust additional Linux operating-system settings in order to support these recommended semaphore settings. To optimize your PostgreSQL resources, you should also scale the server's working memory settings in accordance with "[Special Considerations](#)" on page 21.

Special Considerations

Specialized solutions or use cases might require additional tuning. If you need to experiment with parameters in order to optimize system performance, the most important parameters are as follows:

shared_buffers

specifies the amount of memory to be used for caching data. PostgreSQL also benefits from the file system cache, so `shared_buffers` should not be so large that they interfere with the file system cache. For a large database, set this parameter between 1 GB and up to 25% of the total system memory.

work_mem

specifies the amount of memory to be used for sorts, hashing, and materialization, before writing to temporary disk files. Several running sessions can perform operations concurrently. Therefore, the total memory used might be many times the value of `work_mem`. Keep this in mind when

choosing the value for this parameter. Set this parameter between 16 MB and 64 MB or more, for a specialized use case (for example, frequent very large sorts).

`maintenance_work_mem`

specifies the maximum amount of memory to be used for vacuuming (reclaiming storage used by rows that are marked for deletion) and index builds. For a large database, set this parameter to 256 MB or more.

If your application can tolerate losing a transaction if the computer or storage crashes, you can set the `synchronous_commit` parameter to **Off** for faster updates.

Special Considerations - SAS Visual Investigator

If you are deploying SAS Visual Investigator, PostgreSQL must be installed and tuned. For more information see [Welcome to SAS Visual Investigator](#).

Tuning: SAS Message Broker

Overview

SAS Message Broker, which is based on RabbitMQ, is an intermediary program that converts messages from the protocol of the sender of the message to the protocol of the receiver. The server is configured automatically during deployment. However, to optimize its performance, it is recommended that you perform the tuning recommendations in this section.

Recommendations

Modify Allocations

Memory Allocation

By default, SAS Message Broker is configured to use up to 40% of the physical RAM on the machine on which an instance runs. This value does not guarantee that more than 40% will be used, but it sets a threshold at which publishers are throttled (notified to slow down message sending). You must decide what percentage of memory to dedicate to the message broker.

For example, if your system has 250 GB and you want to dedicate 50 GB to SAS Message Broker, use the following calculation to begin throttling back at 40% of the dedicated memory:

$$(0.4 * 50 \text{ GB}) / 250 \text{ GB} = 0.08 \sim 0.10$$

In the above example, you start throttling back the message broker when it has consumed more than 10% of the available memory. It is difficult to determine the value that the memory threshold, **vm_memory_high_watermark**, should be set to on a system where SAS Message Broker is sharing resources with other services. When the threshold is reached, producers are blocked from sending additional messages until used memory falls below this threshold again. Alternatively, an absolute limit high watermark might be set. However, this value must be less than the amount of available RAM. Otherwise, the message broker will not start.

To set the memory threshold for SAS Message Broker, complete the following tasks:

- 1 On Linux, set the **vm_memory_high_watermark** parameter by editing one of the following files:

- If your environment is enabled for Transport Layer Security (TLS), edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.ssl` file.
- If your environment is not enabled for TLS, edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.tcp` file.

On Windows, set the **vm_memory_high_watermark** parameter by editing the `C:\ProgramData\SAS\Viya\var\lib\rabbitmq-server\rabbitmq.conf` file.

- 2 Specify the following in the configuration file:

```
vm_memory_high_watermark, percentRAM
```

For more information, see [Configuring the Memory Threshold](#).

Disk Space Allocation

By default, SAS Message Broker requires at least 50 MB of free disk space to operate. If this threshold is reached, SAS Message Broker slows down message sending and blocks connections. Therefore, it is recommended that you set the minimum free disk size to the amount of memory that is installed on the machine, if it is available. By configuring a large amount of free disk space, a constrained system is more likely to recover under heavy usage scenarios by providing adequate space for paging considerations. By default, paging of transient messages, which are written to the disk under high memory consumption, starts when the system gets halfway to the **vm_memory_high_watermark**.

To set the free disk size, complete the following tasks:

- 1 On Linux, set the **disk_free_limit** parameter by editing one of the following files:

- If your environment is enabled for Transport Layer Security (TLS), edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.ssl` file.
- If your environment is not enabled for TLS, edit the `/opt/sas/deploymentId/config/etc/rabbitmq-server/rabbitmq.config.tcp` file.

On Windows, set the **disk_free_limit** parameter by editing the `C:\ProgramData\SAS\Viya\var\lib\rabbitmq-server\rabbitmq.conf` file.

- 2 Specify the following in the configuration file:

```
disk_free_limit, {mem_relative, 1.0}
```

For more information, see [Configuring the Disk Free Space Limit](#).

Note: Using a disk space setting that is relative to the memory size assumes that the available disk space is greater than the amount of available memory.

Tuning: SAS Studio 5.2 (Enterprise)

Overview

SAS Studio is a development application for SAS that you access through your web browser. Heap size increases are recommended if you have more than 25 users accessing the SAS Studio 5.2 (Enterprise) server.

Recommendations

It is recommended that you increase the initial heap size option (-Xms) and the maximum heap size option (-Xmx) for SAS Studio 5.2 (Enterprise). However, the initial heap size option (-Xms) should not exceed the value of the maximum heap size option (-Xmx). The maximum heap size option also needs to be increased for SAS Compute Server.

Configure Initial Heap Size

- 1 From SAS Environment Manager, in the navigation bar, click  **Configuration**.
- 2 Navigate to the **All services** view.
- 3 In the **All services** list, select **SAS Studio Viya**.
- 4 At the top of the content pane, click **New Configuration**.
- 5 In the Select Definition window, select **jvm**.
- 6 In the New jvm Configuration window, click **+**.
- 7 In the Add Property window:
 - a In the **Name** field, specify `java_option_xms`.
 - b In the **Value** field, specify the new initial heap size.

CAUTION

The initial heap size option (-Xms) should not exceed the value of the maximum heap size option (-Xmx)

For more information, see [Java Documentation](#).

See Also

- “Edit Configuration Instances” in *SAS Viya Administration: Configuration Properties*
- “Create Configuration Instances” in *SAS Viya Administration: Configuration Properties*

Configure Maximum Heap Size for SAS Studio 5.2 (Enterprise)

- 1 In the New jvm Configuration window, click **+**.
- 2 In the Add Property window:
 - a In the **Name** field, specify `java_option_xmx`.
 - b In the **Value** field, specify the new maximum heap size as `-Xmx4096m`.

CAUTION

The initial heap size option (-Xms) should not exceed the value of the maximum heap size option (-Xmx)

For more information, see [Java Documentation](#).

- c Click **Save**.
- 3 In the New jvm Configuration window, click **Save**.
- 4 Restart SAS Studio 5.2 (Enterprise).

For Red Hat Enterprise Linux 6.7:

```
sudo service sas-viya-studiov-default stop
sudo service sas-viya-studiov-default start
```

For Red Hat Enterprise Linux 7.x or later and SUSE Linux:

```
sudo systemctl restart sas-viya-studiov-default
```

For Windows, in Windows Services Manager, right-click **SAS Studio** and select **Restart**.

Configure Maximum Heap Size for SAS Compute Server

- 1 In the **All services** list, select **compute**.
- 2 At the top of the content pane, click .
- 3 In the Edit jvm Configuration window, click **+**.
- 4 In the Add Property window:
 - a In the **Name** field, specify `java_option_xmx`.

- b In the **Value** field, specify the new maximum heap size as **-Xmx1024m**.
 - c Click **Save**.
- 5 In the New jvm Configuration window, click **Save**.
- 6 Restart the SAS Compute Service.

For Red Hat Enterprise Linux 6.7:

```
sudo service sas-viya-compute-default stop  
sudo service sas-viya-compute-default start
```

For Red Hat Enterprise Linux 7.x or later and SUSE Linux:

```
sudo systemctl restart sas-viya-compute-default
```

For Windows, in Windows Services Manager, right-click the **SAS Compute Service** and select **Restart**.