

SAS 9.3 Web Applications: Tuning for Performance and Scalability



Copyright Notice

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS 9.3 Web Applications: Tuning for Performance and Scalability*, Cary, NC: SAS Institute Inc., 2011.

SAS 9.3 Web Applications: Tuning for Performance and Scalability

Copyright © 2011, SAS Institute Inc., Cary, NC, USA.

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc. Limited permission is granted to store the copyrighted material in your system and display it on terminals, print only the number of copies required for use by those persons responsible for installing and supporting the SAS programming and licensed programs for which this material has been provided, and to modify the material to meet specific installation requirements. The SAS Institute copyright notice must appear on all printed versions of this material or extracts thereof and on the display medium when the material is displayed. Permission is not granted to reproduce or distribute the material except as stated above.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

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

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries.

® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

Table of Contents

Chapter 1 — Overview	1
Establish Your Performance Objective	1
Common JVM Options	3
JBoss JVM Options	4
SAS Remote Services JVM Options	4
Chapter 2 — Pattern 1 - Scaling the Entire SAS Middle Tier	5
Pattern 1: Tuning Parameters for HP and Sun JDKs	7
JVM #1: SAS Web Report Studio	7
JVM #2: SAS Information Delivery Portal, SAS Package Viewer, and BI Portlets	7
JVM #3: SAS BI Dashboard and SAS Shared Services	7
JVM #4: SAS Web Infrastructure Platform	7
JVM #5: SAS Content Server and SAS Workflow Services	7
JVM #6: SAS Remote Services	8
Pattern 1: Tuning Parameters for the IBM SDK	8
JVM #1: SAS Web Report Studio	8
JVM #2: SAS Information Delivery Portal, SAS Package Viewer, and BI Portlets	8
JVM #3: SAS BI Dashboard and SAS Shared Services	8
JVM #4: SAS Web Infrastructure Platform	8
JVM #5: SAS Content Server and SAS Workflow Services	8
JVM #6: SAS Remote Services	8
Chapter 3 — Pattern 2 - Scaling SAS Information Delivery Portal	9
Pattern 2: Tuning Parameters for HP and Sun JDKs	10
JVM #1: SAS Information Delivery Portal and SAS Web Infrastructure Web Applications	10
JVM #2 SAS Web Infrastructure Platform Services, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS Package Viewer	10
JVM #3 SAS Content Server and SAS Workflow Services	10
JVM #4: SAS Remote Services	10
Pattern 2 Tuning Parameters for an IBM JDK	10
JVM #1 SAS Information Delivery Portal and SAS Web Infrastructure Web Applications	10
JVM #2 SAS Web Infrastructure Platform Services, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS Package Viewer	10
JVM #3 SAS Content Server and SAS Workflow Services	11
JVM #4: SAS Remote Services	11
Chapter 4 — Pattern 3 - Scaling SAS Web Report Studio	12
Pattern 3: Tuning Parameters for Sun and HP JDKs	12
JVM #1 SAS Web Report Studio	12
JVM #2 SAS Web Infrastructure Platform	12
JVM #3 SAS Content Server	13
JVM #4: SAS Remote Services	13
Pattern 3: Tuning Parameters for an IBM JDK	13
JVM #1	13
SAS Web Report Studio	13
JVM #2 SAS Web Infrastructure Platform	13
JVM #3 SAS Content Server	13
JVM #4: SAS Remote Services	13
Chapter 5 – SAS Workflow Tuning	14
Chapter 6 – Reconfiguring the SAS 9.3 Middle Tier	15

Stage 1 – Plan the SAS 9.3 Middle Tier Application and JVM Distribution	15
Example 1 – Three JVMs	16
Example 2 – Four JVMs	16
Stage 2 – Prepare and Configure the Web Application Server Environment .	17
Stage 3 – Deploy SAS Themes and SAS Flex Themes to an HTTP Server	17
Stage 4 – Change the Connections for the SAS Web Applications	18
Stage 5 – Change the Connection for SAS Content Server	18
Stage 6 – Change the WebDAV Repository URL	19
Chapter 7 – SAS Application Tuning Parameters	20
Chapter 8 – Web Application Server Tuning Parameters	21
Chapter 9 – Java Virtual Machine Tuning Parameters.....	22
Setting JVM Options for JBoss	23
Setting JVM Options for WebLogic Server.....	23
Setting JVM Options for WebSphere Application Server	24
Chapter 10 – Operating System Tuning Parameters	25
Solaris SPARC and Solaris x86-64	25
Linux	25
AIX	25
HP-UX.....	25
Windows	27

Chapter 1 — Overview

The SAS 9.3 middle tier environment provides an execution environment for SAS Web applications that are used to perform business analytics. Optimizing the middle tier and SAS Web applications for scalability and performance requires careful planning and effort to carry out the plan. The planning and execution can be complex as they balance the demands for availability, reliability, security, and performance. The purpose of this document is to walk through the SAS 9.3 tuning for performance and scalability. This document discusses the tuning methodology and tuning parameters that are involved in SAS Web applications, the Web application server, Java Virtual Machines (JVM), and operating systems.

Establish Your Performance Objective

Establishing your performance objective is the most important step of the entire process for tuning the performance and scalability of the middle tier. Performance requirements are usually identified in terms of transaction response time, number of transactions per second, throughput time, resource utilization, total cost per transaction, availability, and more. Scalability often refers to a component's ability 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 for performance and scalability is approached from the perspective of enhancing the end-user experience with the following SAS Web applications:

- SAS Web Report Studio
- SAS Information Delivery Portal
- SAS BI Dashboard

This document provides an approach for improving performance and scalability based on common patterns. The end-user Web applications make use of Web infrastructure software, and the Web infrastructure software makes use of SAS Foundation software. Each of these layers has an impact on performance. Each deployment pattern in this document provides guidelines on how to tune the software in the lower layers to enhance the user experience with the selected end-user Web application. The following figure shows the software layering described in this paragraph. However, this figure is not a detailed architecture diagram of SAS software; it is a visual aid to assist with tuning the SAS Web applications.

Note: The deployment patterns detailed in this document represent advanced/custom configuration scenarios that require manual steps in order to deploy them. It is not possible to use Software Deployment Wizard to automatically distribute the applications in this manner.

Figure 1 SAS Middle Tier Software Layers

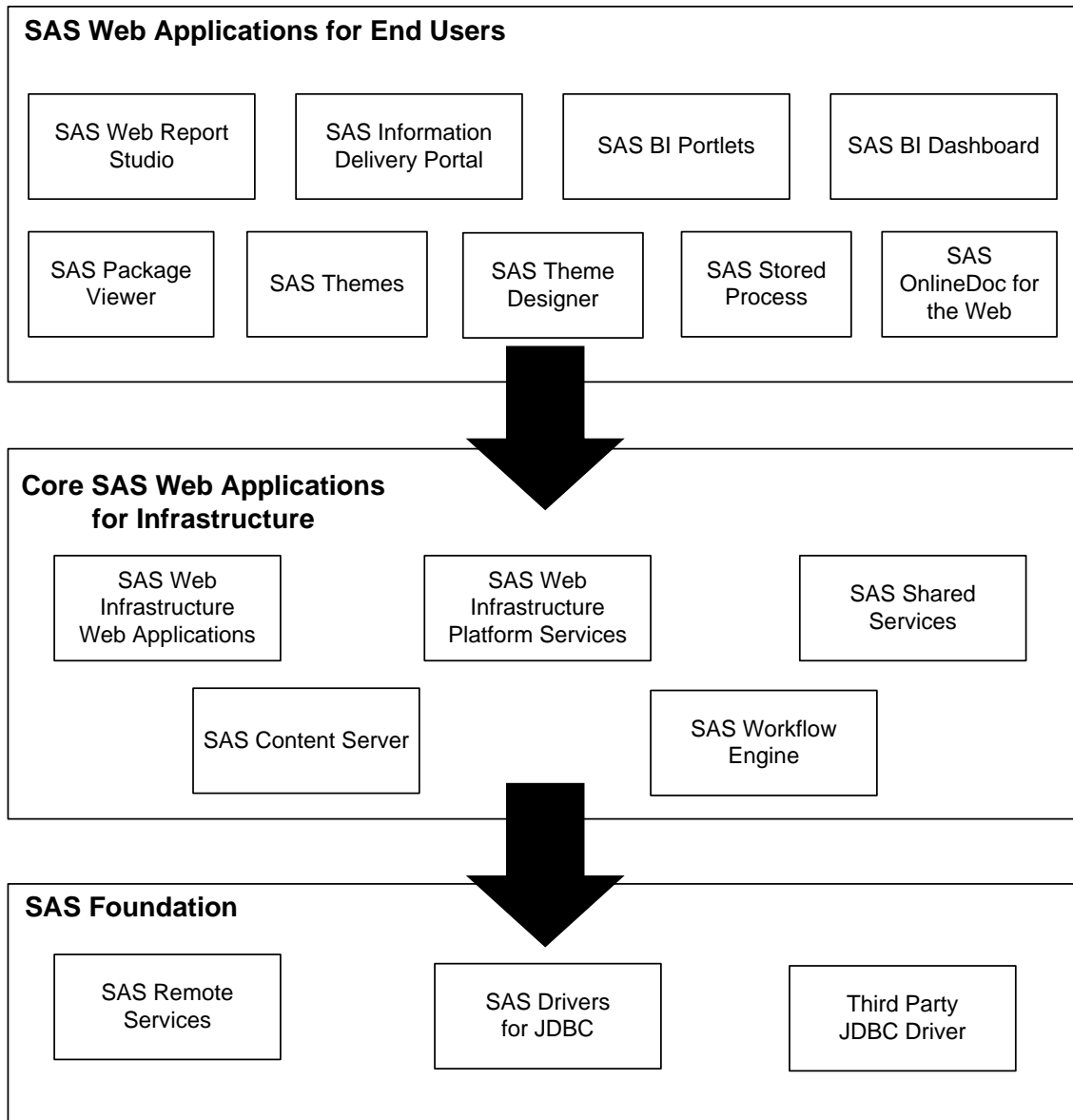


Table 1 categorizes the scalability and performance objectives for the end-user SAS Web applications. The first column identifies the tuning patterns that are provided in this document.

Table 1 Scalability and Performance Objectives

Deployment Pattern	Usage Pattern		
	SAS Information Delivery Portal	SAS Web Report Studio	SAS BI Dashboard
Scaling the entire SAS middle tier	Moderate to Aggressive	Moderate to Aggressive	Moderate to Aggressive
Scaling SAS Information Delivery Portal	Aggressive	None or Low	None or Low
Scaling SAS Web Report Studio	None, Low, or Moderate	Aggressive	None, Low, or Moderate
Scaling SAS BI Dashboard	None, Low, or Moderate	None, Low, or Moderate	Aggressive

The default configuration of SAS 9.3 Web applications in a Web application server environment and its operating system is sufficient in most cases to meet moderate and low classifications of performance and scalability requirements.

Many of the performance and scalability improvements for each pattern are accomplished by adjusting the JVM options for the Web application server. For each pattern, the tuning options are provided according to the JDK vendor:

- Oracle and HP JDKs – These JDKs use the same JVM options.
- IBM JDK – An IBM JDK is used for all Web application servers deployed on AIX and is also used for IBM WebSphere Application Server, unless WebSphere Application Server is deployed on a Sun Solaris or HP-UX Itanium platform.

For more information about the supported Web application servers and the supported operating systems and JDKs, see the [Third Party Software for SAS 9.3 Foundation](#) Web site. This document provides suggested JVM options that will assist you in meeting your scalability and performance objectives. Once you have applied the JVM options to your topology, perform testing to confirm that your objectives have been met, and adjust the JVM options accordingly.

Common JVM Options

The following list identifies the common JVM options that are recommended for all Web application servers that host the SAS 9.3 Web applications. In the subsequent sections of this document, JVM options that are specific to a JDK or application are provided. Make sure that the common options provided here are also included when you reconfigure the Web application server:

```
-Dsas.container.identifier=jboss|weblogic|websphere
-Djava.awt.headless=true
-Dsun.rmi.transport.connectionTimeout=2000
```

```
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-Dsas.auto.publish.port=web_container_port_of_wip
-Dcom.sas.services.logging.disableRemoteList=true
-Dcom.sas.services.logging.disableRemoteLogging=true
-Djava.net.preferIPv4Stack=true
-Djava.net.preferIPv6Addresses=false
-Dmulticast_udp_ip_ttl=1
-Dmulticast.address=multicast_host_address
-Dmulticast.port=multicast_port
-Dcom.sas.log.config.url=file:///SAS-config-dir/Config/Levl/Web/
Common/LogConfig/
```

JBoss JVM Options

The following list identifies the JVM options that apply only to JBoss and must be added to the other JVM tuning options for any deployments that use JBoss:

```
-Djboss.platform.mbeanserver
-Djavax.management.builder.initial=
org.jboss.system.server.jmx.MBeanServerBuilderImpl
-Djboss.home.dir="JBOSS_HOME"
-Djboss.server.home.dir="JBOSS_HOME\server\SASServerN"
-Dsas.deploy.dir="JBOSS_HOME\server\SASServerN\deploy_sas"
```

SAS Remote Services JVM Options

The JVM is provided by either the Sun JDK or IBM SDK depending upon the operating system on which Remote Services runs. For details see the [Third Party Software for SAS 9.3 Foundation](#) Web site. The common JVM options mentioned earlier in this section do not apply for SAS Remote Services. SAS Remote Services is used in all deployment patterns, and the following example shows recommended JVM tuning options that should be used as a starting point in all cases, regardless of the pattern:

Sun JDK

```
-Xms512m -Xmx512m -XX:PermSize=64m -XX:MaxPermSize=64m -Xss120k
-XX:-UseTLAB -XX:+UseConcMarkSweepGC -XX:NewSize=64m -XX:MaxNewSize=64m
-Dcom.sun.management.jmxremote -Djava.net.preferIPv4Stack=true
```

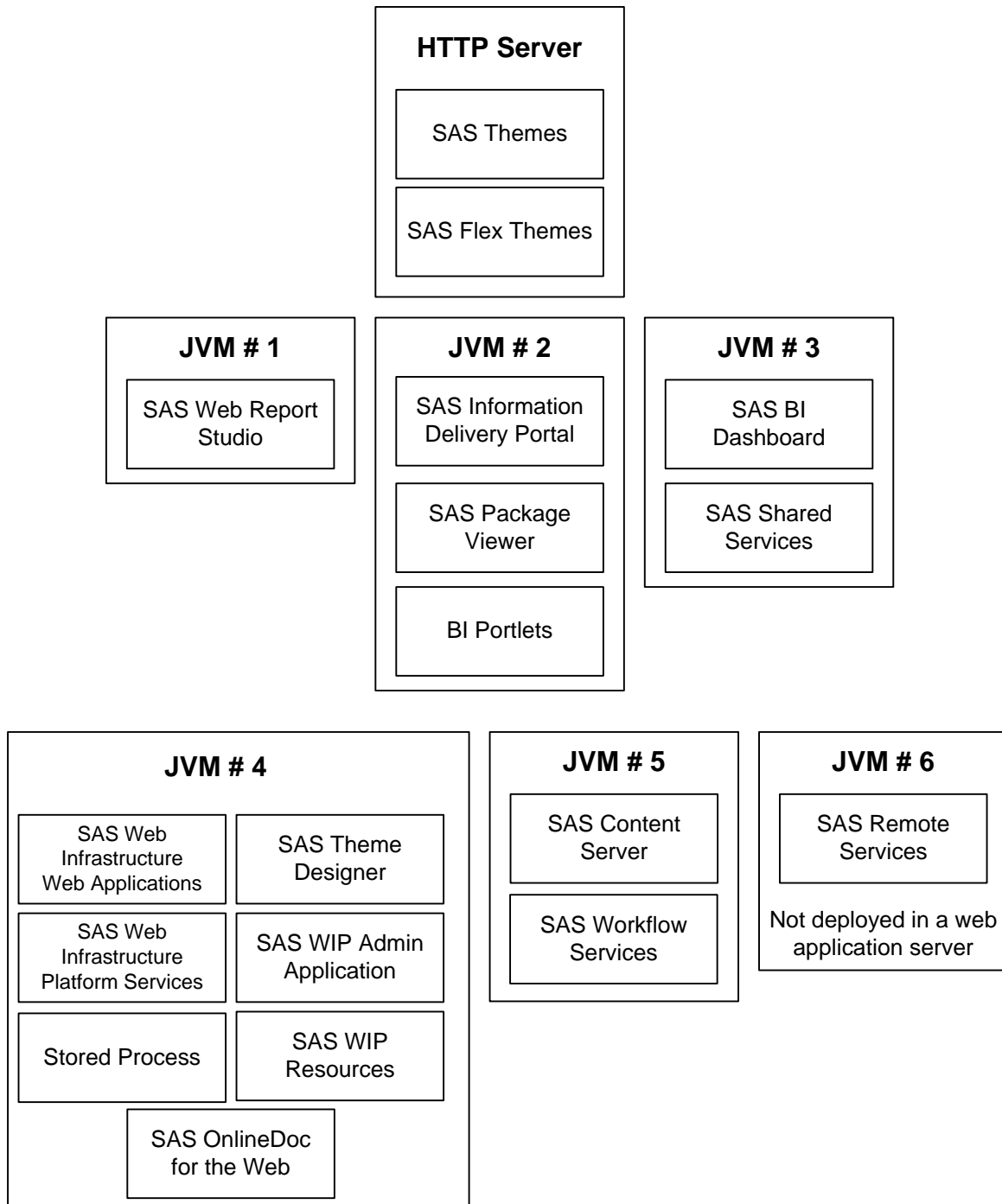
IBM JDK

```
-Xms512m -Xmx512m -Xss120k -Xmsol20k -Xgcpolicy:gencon
```


Chapter 2 — Pattern 1 - Scaling the Entire SAS Middle Tier

Deployment pattern 1 describes how to tune the SAS middle tier to achieve moderate to aggressive performance and scalability objectives for the four key end-user SAS Web applications. In this pattern, each of the four Web applications is run in its own JVM. In this configuration, you can have all the servers on a single physical machine, or multiple physical machines. The decision to use more than one machine depends on the capacity of the machine. Some contributing factors include the number of CPUs on the machine, RAM, CPU clock speed, and L2 cache size. The following figure shows the application topology that scales the entire SAS 9.3 middle tier.

Figure 2 Application Topology for Scaling the Entire SAS Middle Tier



Pattern 1: Tuning Parameters for HP and Sun JDKs

Note: The Common JVM Options on page 3 must be added along with the following application-specific JVM options.

JVM #1: SAS Web Report Studio

```
-server -Xms1280m -Xmx2048m -XX:PermSize=256m
-XX:MaxPermSize=256m -Xss256k -XX:NewSize=256m -XX:MaxNewSize=256m
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC
```

JVM #2: SAS Information Delivery Portal, SAS Package Viewer, and BI Portlets

```
-server -Xms1280m -Xmx2048m -XX:PermSize=512m
-XX:MaxPermSize=512m -Xss256k -XX:+UseTLAB -XX:+UseConcMarkSweepGC
-XX:+DisableExplicitGC -XX:NewSize=128m -XX:MaxNewSize=128m
```

JVM #3: SAS BI Dashboard and SAS Shared Services

```
-server -Xms1024m -Xmx2048m -XX:PermSize=512m
-XX:MaxPermSize=512m -XX:NewSize=256m -XX:MaxNewSize=256m -Xss256k
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC
-XX:+CMSCClassUnloadingEnabled
```

JVM #4: SAS Web Infrastructure Platform

This JVM hosts the SAS Web Infrastructure Platform Applications, SAS Web Infrastructure Platform Services, SAS Stored Process, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS OnlineDoc for the Web applications.

```
-server -Xms1280m -Xmx2048m -XX:PermSize=512m
-XX:MaxPermSize=512m -Xss256k -XX:NewSize=256m -XX:MaxNewSize=256m
-XX:+UseConcMarkSweepGC
-XX:+UseTLAB -XX:+DisableExplicitGC
```

JVM #5: SAS Content Server and SAS Workflow Services

```
-server -Xms1280m -Xmx2048m -XX:PermSize=96m -XX:MaxPermSize=96m
-Xss256k -XX:+UseTLAB -XX:+UseConcMarkSweepGC
-XX:+DisableExplicitGC -XX:NewSize=256m -XX:MaxNewSize=256m
```

JVM #6: SAS Remote Services

SAS Remote Services runs in its own JVM that is provided by Sun JDK and IBM JDK in respective platforms. The tuning options for

Sun JDK

```
-Xms512m -Xmx512m -XX:PermSize=64m -XX:MaxPermSize=64m  
-Xss120k -XX:-UseTLAB -XX:+UseConcMarkSweepGC  
-XX:NewSize=64m -XX:MaxNewSize=64m  
-Dcom.sun.management.jmxremote  
-Djava.net.preferIPv4Stack=true
```

Pattern 1: Tuning Parameters for the IBM SDK

This series of tuning parameters uses the topology identified in deployment pattern #1. The parameters are suited to a Web application server that is using an IBM JDK.

Note: The Common JVM Options on page 3 must be added along with the following application-specific JVM options.

JVM #1: SAS Web Report Studio

```
-Xms1280m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #2: SAS Information Delivery Portal, SAS Package Viewer, and BI Portlets

```
-Xms1280m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #3: SAS BI Dashboard and SAS Shared Services

```
-Xms1024m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #4: SAS Web Infrastructure Platform

This JVM hosts the SAS Web Infrastructure Platform Applications, SAS Web Infrastructure Platform Services, SAS Stored Process, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS OnlineDoc for the Web applications.

```
-Xms1280m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #5: SAS Content Server and SAS Workflow Services

```
-Xms1280m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #6: SAS Remote Services

SAS Remote Services runs in its own JVM that is provided by Sun JDK and IBM JDK in respective platforms. The tuning options for

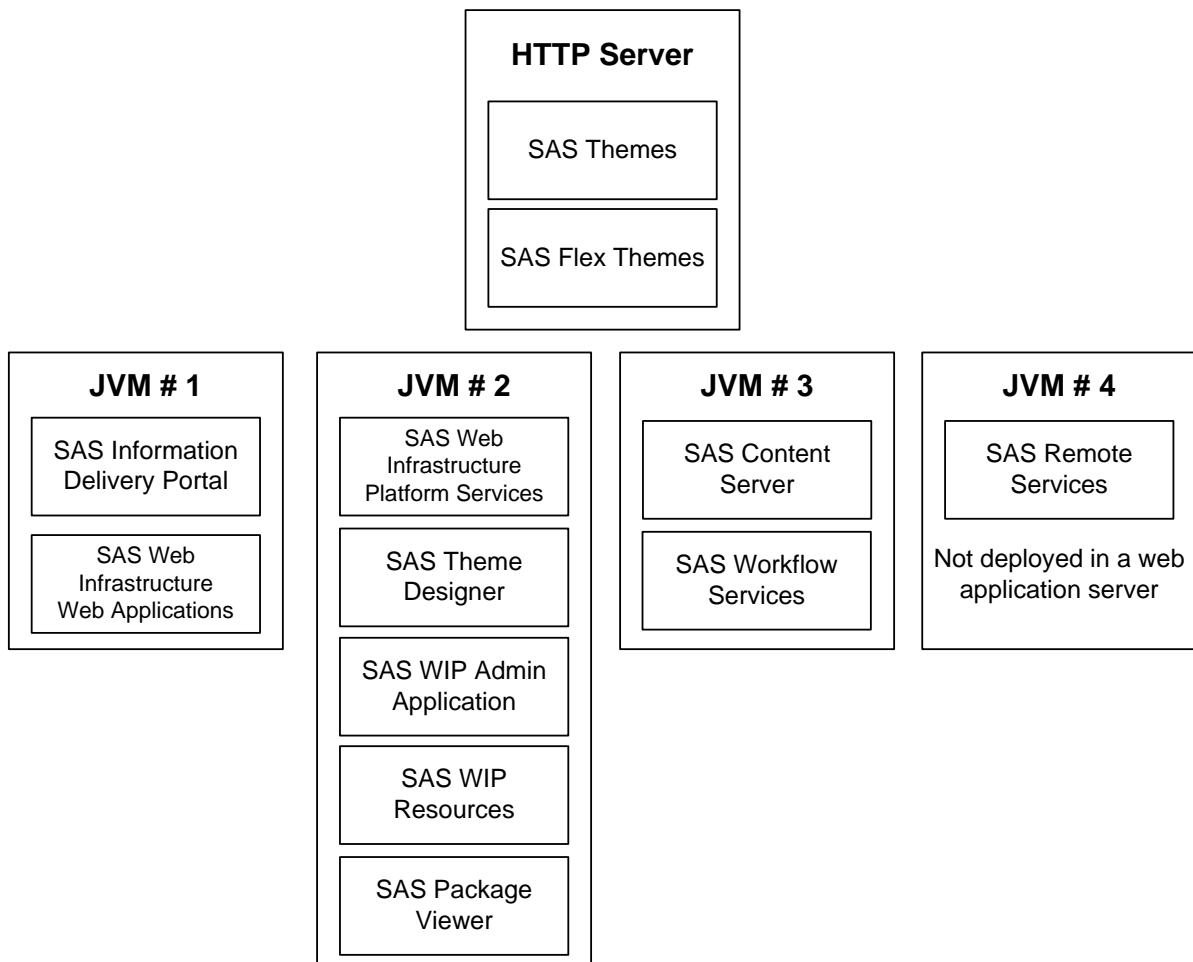
IBM JDK

```
-Xms512m -Xmx512m -Xss120k -Xmso120k -Xgcpolicy:gencon
```

Chapter 3 — Pattern 2 - Scaling SAS Information Delivery Portal

Deployment pattern 2 describes how to configure the SAS middle tier to focus on the SAS Information Delivery Portal Web application. The application is scaled for high performance requirements such as a large number of users, low response time, and high throughput. In this topology, the SAS Information Delivery Portal and the SAS Web Infrastructure Web Applications are deployed together in the same JVM. SAS Web Infrastructure Platform Services, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS Package Viewer are deployed in another JVM. SAS Content Server and SAS Workflow Services are deployed in another JVM. SAS Remote Services uses its own JVM. SAS Themes and SAS Flex Themes are deployed in an HTTP Server. SAS Content Server and SAS Workflow Services are deployed in the same JVM. The following figure shows the Web application topology that scales the SAS Information Delivery Portal application.

Figure 3 Application Topology for Scaling SAS Information Delivery Portal



Pattern 2: Tuning Parameters for HP and Sun JDKs

Note: The Common JVM Options on page 3 must be added along with the following application-specific JVM options.

JVM #1: SAS Information Delivery Portaland SAS Web Infrastructure Web Applications

```
-server -Xms1024m -Xmx2048m -XX:PermSize=512m  
-XX:MaxPermSize=512m -Xss256k -XX:NewSize=128m -XX:MaxNewSize=256m  
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC  
-XX:+CMSClassUnloadingEnabled
```

JVM #2 SAS Web Infrastructure Platform Services, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS Package Viewer

```
-server -d64 -Xms2048m -Xmx2048m -XX:PermSize=512m  
-XX:MaxPermSize=512m -Xss256k -XX:+UseTLAB -XX:+UseConcMarkSweepGC  
-XX:+DisableExplicitGC -XX:NewSize=256m -XX:MaxNewSize=256m
```

JVM #3 SAS Content Server and SAS Workflow Services

```
-server -Xms1024m -Xmx2048m -XX:PermSize=512m  
-XX:MaxPermSize=512m -Xss256k -XX:NewSize=128m -XX:MaxNewSize=256m  
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC  
-XX:+CMSClassUnloadingEnabled
```

JVM #4: SAS Remote Services

SAS Remote Services runs in its own JVM that is provided by a 32-bit Sun JDK.

```
-Xms512m -Xmx512m -XX:PermSize=64m -XX:MaxPermSize=64m -Xss120k  
-XX:+UseTLAB -XX:+UseConcMarkSweepGC -XX:NewSize=64m -XX:MaxNewSize=64m  
-Dcom.sun.management.jmxremote -Djava.net.preferIPv4Stack=true
```

Pattern 2 Tuning Parameters for an IBM JDK

Note: The Common JVM Options on page 3 must be added along with the following application-specific JVM options.

JVM #1 SAS Information Delivery Portal and SAS Web Infrastructure Web Applications

```
-Xms1024m -Xmx4096m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #2 SAS Web Infrastructure Platform Services, SAS Theme Designer, SAS WIP Admin Application, SAS WIP Resources, and SAS Package Viewer

```
-Xms1024m -Xmx3072m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #3 SAS Content Server and SAS Workflow Services

`-Xms1280m -Xmx1280m -Xss256k -Xmso256k -Xgcpolicy:gencon`

JVM #4: SAS Remote Services

SAS Remote Services runs in its own JVM that is provided by a 32-bit IBM JDK. The tuning options for

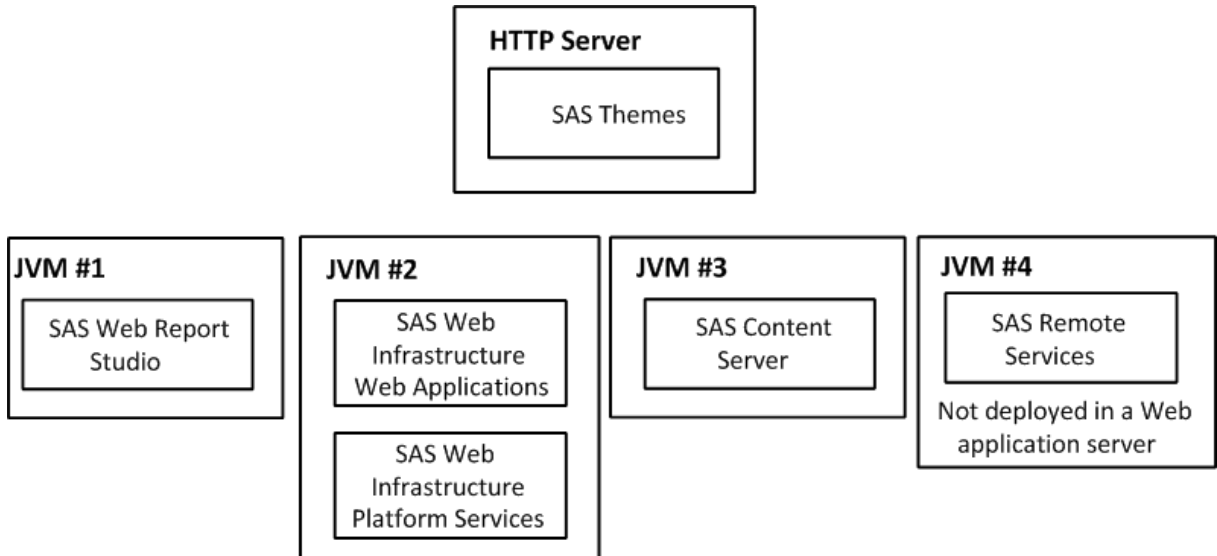
IBM JDK

`-Xms512m -Xmx512m -Xss120k -Xmso120k -Xgcpolicy:gencon`

Chapter 4 — Pattern 3 - Scaling SAS Web Report Studio

Deployment pattern 3 describes how to configure the SAS middle tier to focus on the SAS Web Report Studio application. This application is the focus for high performance requirements such as a large number of users, fast response and high throughput. The following figure shows the Web application topology that scales the SAS Web Report Studio application.

Figure 4 Application Topology for Scaling SAS Web Report Studio



Pattern 3: Tuning Parameters for Sun and HP JDKs

Note: The Common JVM Options on page 3 must be added along with the following application-specific JVM options.

JVM #1 SAS Web Report Studio

```
-server -Xms2048m -Xmx2048m -XX:PermSize=256m
-XX:MaxPermSize=256m -Xss256k -XX:NewSize=256m -XX:MaxNewSize=256m
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC
```

JVM #2 SAS Web Infrastructure Platform

This JVM hosts the SAS Web Infrastructure Platform Services and SAS Web Infrastructure Web Applications.

```
-server -Xms2048m -Xmx2048m -XX:PermSize=320m
-XX:MaxPermSize=320m -Xss256k -XX:NewSize=256m -XX:MaxNewSize=256m
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC
```


JVM #3 SAS Content Server

```
-server -Xms1024m -Xmx2048m -XX:PermSize=512m
-XX:MaxPermSize=512m -Xss256k -XX:NewSize=128m -XX:MaxNewSize=256m
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC
-XX:+CMSClassUnloadingEnabled
```

JVM #4: SAS Remote Services

SAS Remote Services runs in its own JVM that is provided by Sun JDK.

```
-Xms512m -Xmx512m -XX:PermSize=64m -XX:MaxPermSize=64m -Xss120k
-XX:-UseTLAB -XX:+UseConcMarkSweepGC -XX:NewSize=64m -XX:MaxNewSize=64m
-Dcom.sun.management.jmxremote -Djava.net.preferIPv4Stack=true
```

Pattern 3: Tuning Parameters for an IBM JDK

Note: The Common JVM Options on page 3 must be added along with the following application-specific JVM options.

JVM #1**SAS Web Report Studio**

```
-Xms2048m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #2 SAS Web Infrastructure Platform

This JVM hosts the SAS Web Infrastructure Platform Services and SAS Web Infrastructure Web Applications.

```
-Xms1024m -Xmx3072m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #3 SAS Content Server

```
-Xms1024m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon
```

JVM #4: SAS Remote Services

SAS Remote Services runs in its own JVM that is provided by IBM JDK. The tuning options for IBM JDK

```
-Xms512m -Xmx512m -Xss120k -Xmso120k -Xgcpolicy:gencon
```

Chapter 5 – SAS Workflow Tuning

SAS Workflow is a suite of services that provides a platform for the business process augmentation. Use SAS Workflow to model, automate, integrate, monitor, and streamline business processes. SAS Workflow is used primarily by the SAS solutions such as SAS Enterprise Case Management. When SAS Workflow is used directly through custom application development or through integration with standalone or enterprise level business process, it must be tuned based on the usage volume. This section provides tuning recommendations that apply to a high-usage SAS Workflow environment. SAS Workflow is packaged with the SAS Shared Services Web application as Web archive (WAR). For high-usage environments, SAS recommends deploying SAS Workflow in a separate JVM and to use the following JVM tuning options.

Sun JDK

```
-server -Xms2048m -Xmx2048m -XX:PermSize=256m  
-XX:MaxPermSize=256m -Xss256k -XX:NewSize=256m -XX:MaxNewSize=256m  
-XX:+UseConcMarkSweepGC -XX:+UseTLAB -XX:+DisableExplicitGC  
-XX:+UseCompressedOops
```

IBM JDK

```
-Xms2048m -Xmx2048m -Xss256k -Xmso256k -Xgcpolicy:gencon -Xmn384m  
-Xcompressedrefs -Xdisableexplicitgc
```

SAS Workflow Cache Tuning

SAS Workflow uses a cache. The cache is populated based on the SAS Workflow instance creation and loading. By default, the maximum number of cache elements is 100000. This default cache size can process up to 10000 workflow instances. If the deployment requires more than 10000 workflow instances, then the cache size must be increased. The cache size is set in the **ehcache.xml** configuration file. This file is stored in the `sas.workflow.engine.server.jar` under `WEB-INF/lib` directory of SAS Workflow WAR. To change the cache size, you must extract the contents of the `sas.workflow.war` file, edit the `ehcache.xml` file, and then re-create the `sas.workflow.war` file.

If you want to reduce the memory requirement and possibly the need to have a different JVM for Workflow, you can reduce the default cache size by editing the `ehcache.xml` file and re-creating the `sas.workflow.war` file.

Chapter 6 – Reconfiguring the SAS 9.3 Middle Tier

In the previous chapters of this document, several deployment patterns are presented that describe how to achieve improved performance and scalability from the SAS 9.3 middle tier. As demonstrated by the variety of deployment patterns, the SAS 9.3 Web applications have the flexibility to be distributed amongst several JVMs. The SAS Deployment Wizard is used to install and configure SAS software, and it supports creating a topology that uses two JVMs. This chapter describes how to distribute the SAS Web applications amongst three and four JVMs using a manual deployment process. The information in this chapter assumes that SAS 9.3 was already installed with the SAS Deployment Wizard.

Stage 1 – Plan the SAS 9.3 Middle Tier Application and JVM Distribution

As you plan SAS 9.3 environment based on the performance, functional and nonfunctional requirements, decide how many JVMs you will use. The following list identifies the EAR files that are available with SAS 9.3:

- SAS BI Dashboard (sas.bidashboard4.3.ear and sas.bidashboardjsr1684.2.ear)
- SAS Package Viewer (sas.packageviewer4.3.ear)
- SAS Portal (sas.portal4.3.ear)
- SAS Shared Services (sas.shared9.3.ear)
- SAS Stored Process (sas.storedprocess9.3.ear)
- SAS Themes (sas.themes.ear)
- SAS Web Report Studio (sas.webreportstudio4.3.ear)
- SAS WIP Applications (sas.wip.apps9.3.ear)
- SAS WIP Content Server (sas.wip.scs9.3.ear)
- SAS WIP Admin Application (sas.wip.admin9.3.ear)
- SAS WIP Resources (sas.wip.resources9.3.ear)
- SAS WIP Services (sas.wip.services9.3.ear)
- SAS Flex Themes (sas.flexthemes2.53.ear)
- SAS BI Portlets (sas.biportlets4.3.ear)
- SAS Workflow Engine (sas.workflow9.3.ear)
- SAS Theme Designer (sas.themedesigner2.53.ear)
- SAS Workflow Services (sas.workflow9.3.ear)

Note: The SAS Themes application is actually a WAR file with static content such as images and cascading style sheets. The SAS Flex Themes application is also a WAR file with static .swf (Small Web Format, formerly known as ShockWave Flash) files. For the topologies described in this document, SAS recommends that you serve the static content from SAS Themes and SAS Flex Themes through an HTTP server. Information about performing this reconfiguration is provided in this chapter. In addition, SAS Remote Services is not identified in the following examples, but it must be present in the topology and it runs in its own JVM.

Example 1 – Three JVMs

The following table shows a recommended distribution of SAS 9.3 middle tier applications in a three JVM (Servers) topology.

Server number	SAS 9.3 Middle Tier Applications
Server1	SAS WIP Applications (sas.wip.apps9.3.ear) SAS WIP Content Server (sas.wip.scs9.3.ear) SAS WIP Services (sas.wip.services9.3.ear) SAS WIP Admin Application (sas.wip.admin9.3.ear) SAS WIP Resources (sas.wip.resources9.3.ear) SAS Workflow Services (sas.workflow9.3.ear) SAS Theme Designer (sas.themedesigner2.53.ear)
Server2	SAS BI Dashboard (sas.bidashboard4.3.ear) SAS Shared Services (sas.shared9.3.ear) SAS Portal (sas.portal4.3.ear) SAS Package Viewer (sas.packageviewer4.3.ear)
Server3	SAS Web Report Studio (sas.webreportstudio4.3.ear) SAS Stored Process (sas.storedprocess9.3.ear)

Example 2 – Four JVMs

The following table shows a recommended distribution of SAS 9.3 middle tier applications in a four JVM (Servers) topology.

Server number	SAS 9.2 Middle Tier Applications
Server1	SAS WIP Applications (sas.wip.apps9.3.ear) SAS WIP Content Server (sas.wip.scs9.3.ear) SAS WIP Services (sas.wip.services9.3.ear) SAS WIP Admin Application (sas.wip.admin9.3.ear) SAS WIP Resources (sas.wip.resources9.3.ear) SAS Workflow Services (sas.workflow9.3.ear) SAS Theme Designer (sas.themedesigner2.53.ear)
Server2	SAS Portal (sas.portal4.3.ear) SAS Package Viewer (sas.packageviewer4.3.ear)
Server3	SAS Web Report Studio (sas.webreportstudio4.3.ear)

	SAS Stored Process (sas.storedprocess9.3.ear)
Server4	SAS BI Dashboard (sas.bidashboard4.3.ear) SAS Shared Services (sas.shared9.3.ear)

Stage 2 – Prepare and Configure the Web Application Server Environment

This section assumes that you are familiar with configuring your Web application server (JBoss, WebLogic Server, or WebSphere Application Server). The following list identifies the high-level steps that you must perform on each Web application server:

1. Create the servers that you decided in stage 1. For each of the servers, perform the following steps.
2. Create and Configure a Java Mail Session.
3. Create and Configure a JMS Server and resources.
4. Configure JDBC Driver - SAS Table Server.
5. Configure Data Source – Shared Service.
6. Deploy the respective SAS 9.3 middle tier applications (EAR files) to the server.
7. Load content to SAS Content Server. Configure the JAAS Login Module for Application Logins – PFS and SCS.
8. Set the `-Dsas.auto.publish.port=JVM` option to the Web application server's own port number.

For detailed instructions about how to perform the steps in the previous list, see the `SAS-config-dir/Levn/Documents/Instructions.html` file. In addition, sample `Instructions.html` files are available on the [Third Party Software for SAS 9.3 Foundation](#) Web site.

Stage 3 – Deploy SAS Themes and SAS Flex Themes to an HTTP Server

In all the deployment patterns provided in this document, SAS recommends deploying the SAS Themes and SAS Flex Themes content to an HTTP server. By serving the SAS Themes and SAS Flex Themes content from an HTTP server, you shift the processing load of serving static HTML files from the Web application server to the HTTP server. The following instructions describe how to perform these deployments on Apache HTTP Server. Performing this reconfiguration on other HTTP servers is similar. To configure Apache HTTP Server to serve the static HTML files for SAS Themes perform the following steps:

1. Change directory to `APACHE_HOME/htdocs` and make a new directory named `SASTheme_default`.
2. Copy the contents of `SAS-config-dir/Levn/Web/Staging/exploded/sas.themes.ear` into `APACHE_HOME/htdocs/SASTheme_default`.

To configure Apache HTTP Server to serve the static HTML files for SAS Flex Themes perform the following steps:

1. Change directory to `APACHE_HOME/htdocs` and make a new directory named `SASFlexThemes`.
2. Copy the contents of `SAS-config-dir/Levn/Web/Staging/exploded/sas.flexthemes2.5.3.ear` into `APACHE_HOME/htdocs/SASFlexThemes`.

Stage 4 – Change the Connections for the SAS Web Applications

After SAS Themes and SAS Flex Themes are deployed to the HTTP server and the SAS Web applications are distributed to different servers, information about accessing the applications such as host and port must be updated in SAS metadata. Change the connection information to a URL that includes the Web application server host name and port number for the Web application server that is hosting the SAS Web application. Information about the SAS Themes application must be changed to the HTTP server host name and port.

To change the connection access point, follow these steps in SAS Management Console:

1. Select **Application Management > Configuration Manager**.
2. Right-click on the SAS Web application you want to reconfigure, and select **Properties**.
3. Click the **Connection** tab, set **Host Name** and **Port Number** to the host name and port number of the Web application server that is hosting the SAS Web application (use the HTTP server host name and port for `SASTheme_default`), and then click **OK**.

When IBM HTTP Server uses a reverse proxy for WebSphere Application Server, that IBM HTTP Server must be configured to accept all content, such as attachments. Follow these steps in the WebSphere administrative console:

1. Change the virtual host for the SAS Content Server application to the default host.
2. Select **Servers > Web servers > webserver1 > Plug-in properties > Request and Response**.
3. Select the **Accept content** for all requests check box.
4. Generate and propagate the Web server plug-in configuration file.

Stage 5 – Change the Connection for SAS Content Server

If the SAS Content Server application (`sas.wip.scs9.3.ear`) is not deployed on the original Web application server and port, then you must reconfigure SAS metadata with the connection information. This change is similar to the change required for each of the SAS Web applications. SAS metadata must be changed to identify the host name and port of the Web application server that is hosting the SAS Content Server application. To reconfigure the host name and port of the SAS Content Server in SAS metadata, perform the following steps in SAS Management Console:

1. Select **Server Manager > SAS Content Server**.
2. Right-click the **Connection: SAS Content Server** icon in the right panel and select **Properties**.
3. Click the **Options** tab and set the **Host name** and **Port number** fields to the host name and port number of the Web application server that is hosting the `sas.wip.scs9.3.ear` EAR file or the HTTP Server that is performing a reverse proxy for the SAS Content Server.
4. Click **OK**.

Stage 6 – Change the WebDAV Repository URL

Just as in Stage 5, if the SAS Content Server application (`sas.wip.scs9.3.ear`) is not deployed on the original Web application server and port, then you must reconfigure SAS metadata with the connection information for the WebDAV repository. There are five applications that use SAS metadata to identify the connection information for the SAS Content Server. These applications are identified in the following list:

- Remote Services
- SASPackageViewer4.3 Local Services
- SASPortal4.3 Local Services
- SASStoredProcess9.3 Local Services
- SASWebReportStudio4.3 Local Services

To reconfigure the WebDAV URL for the applications, perform the following steps in SAS Management Console:

1. Select **Environment Management > Foundation Services Manager**.
2. Select the application and then select **Core > Information Service**.
3. Right-click **Information Service** and select **Properties**.
4. On the **Information Service Properties** dialog box, click the **Service Configuration** tab and then click **Configuration**.
5. On the **Information Service Configuration** dialog box, click the **Repositories** tab.
6. Select **WebDAV** and then click **Edit**.
7. Change the **Host** and **Port** values to the host name and port of the Web application server that is hosting the `sas.wip.scs9.3.ear` EAR file.
8. Click **OK** to close the **Information Service Configuration** dialog box.
9. Click **OK** to close the **Information Service Properties** dialog box.

Chapter 7 – SAS Application Tuning Parameters

The following table provides JVM options that are used to configure the SAS Web applications. A description is provided for each option to help you decide whether modifying a value is appropriate for your deployment.

JVM Option	Recommended Value	Description
-Dhttp.proxyHost	proxy server host name	The HTTP proxy options are used by the SAS Information Delivery Portal to access DTDs over the Internet. These options are used only when a proxy server for Internet connections exists. If you use a proxy server, make sure that the local domain, within the network boundary, is included in the exclude list. For example: -Dhttp.proxyHost=<host name> -Dhttp.proxyPort=80 -Dhttp.nonProxyHosts= "*.*example.com localhost"
-Dhttp.proxyPort	proxy server port	
-Dhttp.nonProxyHosts	non-proxy host (exclude list)	
-Djgroups.bind_addr	host name or IP address	Use this option on a system that has multiple network interfaces (physical or virtual). When this option is used, also use <code>-Dresolve.dns=false</code> .
-Dmulticast_udp_ip_ttl	1 (default value is 64)	This option sets the time to live on UDP packets that are used for multicast calls.
-Dlogoff.corePoolSize	50	Use this option in an environment where concurrent user connections are significantly high (for example, over 500 connections).
-Dlogoff.maxPoolSize	50	Use this option with the <code>-Dlogoff.corePoolSize</code> parameter if the concurrent user connections and the frequency of user logoffs is significantly high (for example, over 500 connections and logoffs).

Chapter 8 – Web Application Server Tuning Parameters

The following table provides a set of JVM options that are identified by the Web application server vendor, and that are relevant to the SAS 9.3 middle tier.

Web Application Server	Parameter	Recommended Value	Description
JBoss	Thread Pool	25 min and 75 max	Controls the number of threads in the Web container thread pool.
JBoss	Disable deployment scanner		Disable the deployment scanner that is used for hot deployment and undeployment.
WebLogic	<code>-Dweblogic.ChunkSize</code>	a multiple of the network maximum transfer unit (MTU), minus the TCP header size	Set this value as the number of bytes and provide it with the JVM options for the server start up.
WebLogic	Accept Backlog		Increase the default value by 25% when you see “connection refused or dropped” message at the client, but no errors on the server.
WebSphere	Thread Pool		Controls the number of threads in the Web container thread pool.
WebSphere	<code>HttpSessionIdReuse</code>	<code>true</code>	Add this custom property to the JVM options for the application server.
WebSphere	<code>prependSlashToResource</code>	<code>true</code>	Add this custom property to the application server.
WebSphere	<code>com.ibm.ws.webcontainer.channelwritetype</code>	<code>sync</code>	Add this custom property to the Web container.
WebSphere	JSP enable class reloading	<code>disable</code>	Disable this property for each of the SAS Web applications.

Chapter 9 – Java Virtual Machine Tuning Parameters

SAS 9.3 supports 64-bit platforms. The total heap, Java heap plus the native heap, cannot exceed a 2GB limit in a Windows environment. For information about how to increase the limit, see the following URL:

<http://technet.microsoft.com/en-us/library/bb124810.aspx>

SAS recommends increasing the limit to 3GB for any aggressive performance and scalability requirements. The goal of the JVM tuning is to minimize the garbage collection cycles while maximizing the performance and number of clients that the SAS Web applications can support.

The following table summarizes general guidelines for JVM tuning in the SAS 9.3 middle tier when the applications are distributed to multiple JVMs. If you plan to deploy the SAS 9.3 middle tier to one or two JVMs, then use the default JVM options that were provided by the SAS Deployment Wizard during the installation of SAS software.

JVM Option	64-Bit Platform Value
Heap size, min and max	-Xms1024m -Xmx2048m
Stack size	-Xss256k
Permanent generation size (Sun and HP JDKs)	-XX:PermSize=256m
Young generation size (Sun and HP JDKs)	-XX:NewSize=256m -XX:MaxNewSize=256m
Garbage collection (Sun and HP JDKs)	-XX:+UseConcMarkSweepGC
Garbage collection (IBM JDK)	-Xgcpolicy:gencon
Distributed garbage collection intervals for 64-bit	-Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000

The following table describes the JVM options that are used in the SAS 9.3 middle tier.

JVM Option	Description
-Xms	Minimum heap
-Xmx	Maximum heap
-Xss	Stack size
-XX:PermSize	For Sun and HP JDKs this option sets the initial permanent generation storage size.
-XX:MaxPermSize	For Sun and HP JDKs this option sets the maximum permanent storage size.
-XX:NewSize -XX:MaxNewSize	For Sun and HP JDKs this option sets the young generation sizes.
-XX:UseConcMarkSweepGC	For Sun and HP JDKs these options modify garbage collection to use the concurrent low pause collector.

Setting JVM Options for JBoss

This section describes how to set JVM options for JBoss.

When JBoss is deployed on UNIX or Linux, the JVM options are specified in the `JAVA_OPTS` variable that is found in `JBOSS_HOME/bin/SASServer1.sh`.

When JBoss is deployed on Windows as a service, the JVM options are specified in the `JBOSS_HOME\server\SASServer1\wrapper.conf` file. Create a new entry for any new JVM options. The following example shows a few of the lines from the file:

```
# Java Additional Parameters
wrapper.java.additional.1=-Xms768m
wrapper.java.additional.2=-Xmx768m
wrapper.java.additional.3=-XX:PermSize=512m
wrapper.java.additional.4=-XX:MaxPermSize=512m
wrapper.java.additional.5=-Xss160k
wrapper.java.additional.6=-XX:NewSize=128m
```

When JBoss is deployed on Windows, but run from scripts rather than as a service, the JVM options are specified in the `JAVA_OPTS` variable that is found in the `JBOSS_HOME\bin\SASServer1.bat` file.

Setting JVM Options for WebLogic Server

For Oracle WebLogic Server, you can set JVM options by editing a start script or by using the Administration Console. Use whichever approach you have used to set other server options. The paragraphs below explain how to use each approach.

Note: If you are not running the Node Manager, then you must specify JVM options in a start script.

To set the JVM options in a start script, follow these steps:

1. Change directories to your WebLogic Server domain directory that is configured for SAS 9.3. For a SAS Deployment Wizard configured domain, this directory is `SAS-config-dir/Levn/Web/SASDomain/bin`.
2. Open the server start script in an editor. The script is named `startManagedWebLogic` with a `.sh` or `.cmd` suffix.
3. Uncomment the line reserved for setting JVM options and add them. For a SAS Deployment Wizard configured domain, use the `JAVA_OPTIONS` variable.
If the same script is used to start more than one managed server and you want each managed server to start with different JVM options, then you need to perform additional editing so that the preferred JVM options are applied to each managed server.
4. Save your changes, and close the file.
5. Restart the server so that the new settings are used.

To set the JVM options using the Administration Console, follow these steps:

1. In the **Domain Structure** panel, expand the **Servers** node.
2. Click the name of the server that you want to configure.
3. In the right pane, click **Server Start**.
4. Select **Lock & Edit**.
5. In the **Arguments** text field, provide the JVM options. After inserting your options, click **Save** and then click **Activate Changes**.
6. Restart the server so that the new settings are used.

Setting JVM Options for WebSphere Application Server

Use the WebSphere Application Server administration console to set JVM options. The following procedure describes how to set the parameters for WebSphere Application Server version 6.1 and version 7.0. For more information, see the IBM WebSphere Application Server documentation.

1. On the left panel, expand the **Servers** node For WebSphere Application Server 6.1. For WebSphere Application Server 7.0, there is an additional expansion needed called **Server Types**.
2. Click **Application Servers** and then click the name of the server that you want to configure.
3. For WebSphere 6.1, select **Process Definition > Java Virtual Machine**. For WebSphere 7.0, select **Java and Process Management > Java Virtual Machine**.
4. Provide the JVM options in the **Generic JVM Arguments** text field.
5. Click **Apply** and then **Save** the changes to the master configuration.
6. Restart the server so that the new settings are used.

Note: Some parameters can be specified in other boxes on the Java Virtual Machine page. Avoid placing the same settings in different text fields on this page; this can have unpredictable results.

Chapter 10 – Operating System Tuning Parameters

There are a number of configuration changes and variables that you can set to tune the SAS 9.3 middle tier for your performance and scalability needs. The following sections show how to configure the settings that are relevant to the SAS 9.3 middle tier and the Web application servers that SAS 9.3 supports.

Note: For UNIX and Linux operating environments, you must also confirm (or configure) that the ulimit resource limit for open file descriptors is at least 20480. Information about this resource limit is provided at the [Pre-Installation Steps for JBoss, WebLogic, and WebSphere Application Servers on the SAS 9.3 Web site](#). This step is especially important for deployments that use WebLogic Server.

Solaris SPARC and Solaris x86-64

Use the following commands to set the SAS recommended parameters:

```
ndd -set /dev/tcp tcp_keepalive_interval 300000
nnd -set /dev/tcp tcp_time_wait_interval 60000
nnd -set /dev/tcp tcp_conn_req_max_q 256
nnd -set /dev/tcp tcp_conn_req_max_q0 2500
nnd -set /dev/tcp tcp_max_buf 4000000
```

Linux

Use the following commands to set the SAS recommended parameters:

```
echo 30 > /proc/sys/net/ipv4/tcp_fin_timeout
echo 3000 > /proc/sys/net/core/netdev_max_backlog
echo 3000 > /proc/sys/net/core/somaxconn
echo 15 > /proc/sys/net/ipv4/tcp_keepalive_intvl
echo 5 > /proc/sys/net/ipv4/tcp_keepalive_probes
```

AIX

Use the following commands to set the SAS recommended parameters:

```
/usr/sbin/no -o tcp_timewait =1
/usr/sbin/no -o tcp_keepidle=600
/usr/sbin/no -o tcp_keepintvl=10
/usr/sbin/no -o tcp_keepinit=40
```

HP-UX

Use the following commands to set the SAS recommended parameters:

```
nnd -set /dev/tcp tcp_conn_request_max 8192
nnd -set /dev/tcp tcp_keepalive_interval 7200000
nnd -set /dev/tcp tcp_keepalives_kill 1
```

In addition, SAS recommends tuning the kernel parameters by using the SAM utility or the kctune utility.

Kernel Parameter	Recommended Value
swchunk	8192
shmseg	512
maxdsiz	3221225472
maxdsiz_64bit	64424509440
maxfiles_lim	10000 (Change this one before maxfiles.)
maxfiles	8192
semume	512
semmsl	3072
msgssz	512 (Change this one before msgmax)
nkthread	10000
max_thread_proc	4096
nproc	8192 (Change this one before maxuprc)
maxuprc	4096
nflocks	11585
ninode	8110
msgmap	13109
msgseg	32767 (Change this one before msgmax)
msgmnb	65535 (0x10000) (Change this one before msgmax)
msgmnb	131070 (when running multiple JVMs on the same system)
msgmax	65535 (0x10000)
msgmax	131070 (when running multiple JVMs on the same system)
msgmni	4634
semmns	11586
semmni	8192
semmnu	8180
shmmax	1.85514E+11
shmmni	8192
STRMSGSZ	65535
dbc_max_pc	10

nstrpty	60
cmc_plat_poll	15
msgtql	13107

Windows

Microsoft recommends performing a system backup before editing the registry. Use **regedit** to set the SAS recommended parameters:

1. Access HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TCPIP\Parameters. Add the following **DWORD Values**:

Name	Value
TcpTimedWaitDelay	0x1e (30 seconds)
MaxUserPort	greater than 32768

2. Set values for the number of available connections. Request a minimum of 20 and a maximum of 1000 available connections. Increase the number of available connections by 10 each time that there are fewer than the minimum numbers of available connections. Access HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\AFD\Parameters. Add the following **DWORD Values**:

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

3. For Windows 2000 hosts, access HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\. Add a **DWORD Value** with a name of TcpDelAckTicks and a value of 0.
4. For Windows 2003 hosts, access HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\. Add a **DWORD Value** with a name of TcpAckFrequency and a value of 1.
5. Restart Windows.



THE
POWER
TO KNOW.

support.sas.com

SAS is the world leader in providing software and services that enable customers to transform data from all areas of their business into intelligence. SAS solutions help organizations make better, more informed decisions and maximize customer, supplier, and organizational relationships. For more than 30 years, SAS has been giving customers around the world The Power to Know®. Visit us at **www.sas.com**.