

Paper 305-2010

The SAS[®] Middle Tier: Providing Integration Services for the SAS[®] Intelligence Platform

Casey Hadden and Marty Tomasi, SAS Institute Inc., Cary, NC

ABSTRACT

As part of its mission to provide an integrated set of software applications, solutions, and services, SAS[®] added a new level of integration capabilities with its release of 9.2. The SAS middle tier, which includes the SAS Web Infrastructure Platform and SAS Shared Services, contains a collection of middle-tier services and applications that provide infrastructure and integration features. Services include common authentication, auditing, mail, events and alerts, configuration, collaboration, and monitoring of SAS servers. The SAS middle tier makes use of standard technologies such as Java EE (JMS, JDBC), SOAP-based Web services, and IP multicasting.

This paper presents an overview of the SAS middle tier and how it enables integration not just with SAS applications, but with enterprise technologies such as security, messaging, mail, and administration. The presentation highlights other benefits, such as common diagnostic techniques and manual configuration that enables support for enterprise-scale requirements.

INTRODUCTION

SAS[®] Integration Technologies has long been the cornerstone of the SAS blueprint for Enterprise Application Integration, which enables an effective distribution of analytics reporting and business intelligence. Integration Technologies is just that — a set of integration and systems development tools that are used to connect SAS technologies with other applications in the enterprise. Through a combination of industry-standard technologies and technology developed by SAS, organizations can develop custom applications that use SAS.

Beginning with its 9.2 release, SAS introduced a new middle-tier component as the core of its Intelligence Platform. In addition to the support for client/server technologies such as Distributed Component Object Model (DCOM), Common Object Request Broker Architecture (CORBA), and Java remote method invocation (RMI) that SAS Integration Technologies had long provided, the addition of these new middle-tier components in SAS 9.2 also extended support to *n*-tier architectures built around HTTP-based technologies. The middle-tier components not only provide for common infrastructure, but also support distributed deployments and configurations that support enterprise requirements.

This paper provides a broad overview of the SAS 9.2 middle tier, how that middle tier fits in with the rest of the SAS platform, and how it facilitates integration across the suite of SAS applications. The paper also presents an inventory of the industry-standard technologies that enable the SAS middle tier, and it describes the services and applications that comprise the middle-tier components.

WEB INFRASTRUCTURE PLATFORM AND SHARED SERVICES

The centerpiece of the SAS middle tier is its Web Infrastructure Platform, a collection of services and applications that provides primary integration capabilities. The Web Infrastructure Platform includes shared Web applications to anchor an integrated suite. Standard infrastructure services are also included, and some provide access via standard Simple Object Access Protocol (SOAP)-based Web services. These services include:

- audit service to support application auditing, integration with Application Response Measurement (ARM), and so on
- authentication service for integrated, single sign-on security capabilities across applications
- configuration service for common, centralized management of all application settings for SAS applications and solutions
- DAV content service for client access (including via SOAP) to content stored in the SAS[®] Content Server
- mail service for sending e-mail via a JavaMail session that is configured in the application server, with complete support for inline data and attachments

- template service to support custom notification messages
- prompting service to facilitate cascading prompt support over Web services
- status and monitoring service to support system validation, license checking, capability verification, and SAS server status via Java Management Extensions (JMX)
- theme service to support customizable Web browser styles, including support for right-to-left locales, for high-performing caching, and for deploying themes effectively in various environments configured with Secure Sockets Layer (SSL)

For application integration, the SAS applications that provide direct middle-tier access to SAS servers, such as SAS BI Web Services or the SAS Stored Process Web Application, are simply available for use in a standard deployment that includes the Web Infrastructure Platform.

Another middle-tier component is bundled with SAS enterprise offerings such as the SAS[®] Enterprise BI Server. This component, called SAS Shared Services, is a collection of services and applications that provide a common set of business intelligence. The set of services includes:

- alert registration and notification services, which provide user opt-in alerts, client registration, and support for sending user notifications to e-mail, via Short Message Service (SMS) text messages, or for display in a portlet
- attachment service for managing attached documents and storage in the SAS Content Server
- comment service for collaboration and annotation capabilities
- workflow services for application integration with business process management features

TECHNOLOGIES

The SAS middle tier makes use of industry-standard technologies in order to provide features and integration across the SAS application suite.

JAVA EE

The Java Platform, Enterprise Edition (Java EE) is a platform for delivering enterprise applications with the Java programming language. Java EE is defined by a specification and includes several other independent specifications as well as those unique to Java EE. SAS requires the Java EE 1.4 specification version.

The Java EE specification is implemented by Java EE application server vendors. SAS requires an application server that implements the Java EE 1.4 specification and supports IBM WebSphere Application Server, JBoss Application Server, and Oracle WebLogic Server. Detailed information about the platforms and supported Java EE application servers can be found in "SAS 9.2 Support for Web Application Servers and HTTP Servers" (2010).

HTTP

SAS Web applications make use of Java servlet functionality to receive and process HyperText Transfer Protocol (HTTP) requests. Client applications communicate to the middle tier, and middle-tier applications communicate to each other, via HTTP. This provides a well-known protocol for IT staff to track and secure from the end-user machine.

The middle tier communicates to the server tier processes through the protocol that is appropriate for the server.

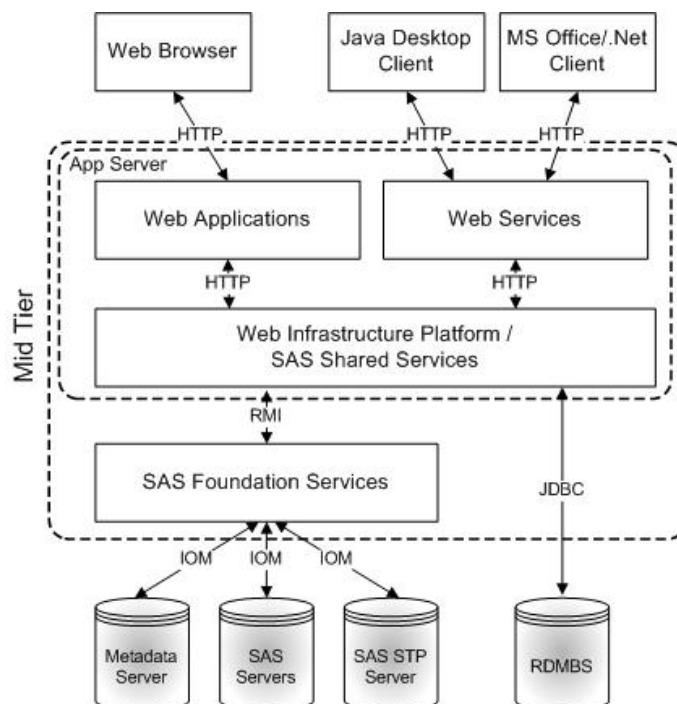


Figure 1. The Standard Communication Protocols Used between the Various Components in the Different Tiers of a Typical SAS Environment.

SAS applications also support HyperText Transfer Protocol Secure (HTTPS) and SSL for secured client access.

JMS

The Java Message Service (JMS) allows distributed application components to be loosely coupled and provides reliability and asynchronous execution features. JMS provides a point-to-point model with queues or a publish-and-subscribe model with topics.

SAS applications use JMS functionality to implement event-driven architectures. By definition and design, an event-driven architecture is extremely loosely coupled, and highly distributed. The source or sender of an event is aware only that the event transpired. The sender has no knowledge of the event's subsequent processing, nor does the sender know or understand the interested parties.

The middle tier defines and makes use of several JMS queues for performing user-level notifications and driving workflows.

JDBC

The Java Database Connectivity (JDBC) API provides database-independent connections between the Java environment and many popular database implementations.

JTA

The Java Transaction API (JTA) specifies standard Java interfaces between a transaction manager and the resources involved in a distributed transaction system.

The middle tier's JDBC access works with the transaction manager implementation provided by the Java EE application server. This allows multiple different resources to be updated in an all-at-once manner. Any failure results in the operations being undone across all resources. In this manner, the middle tier can ensure that a consistent view is maintained across the software.

JMX

Java Management Extensions (JMX) is a Java technology supplying tools for management and monitoring of software based on the Java platform. JMX managed beans, known as MBeans, have been implemented to provide a standard way of managing SAS resources.

Any standard JMX tools can be used to access the MBeans that manage SAS resources.

SAS Servers and JMX: The ServerFactory MBean is the starting point for managing SAS servers. The deployment and initialization of the Web Infrastructure Platform Web applications registers this MBean with the name:

```
com.sas.services:type=ServerFactory
```

The ServerFactory MBean reads the list of available SAS servers from the SAS[®] Metadata Server. This MBean can be used to register additional MBeans that enable the running servers to be managed and monitored directly.

JMX and Service Management: Several middle-tier services provide MBean access under `com.sas.svcs` in the MBeans hierarchy. These MBeans provide access to basic system information such as the list of applications. They also contain operations for reloading cached information or purging certain caches when necessary. For example, these can be used when the Configuration Manager functionality within the SAS[®] Management Console is used to update application metadata.

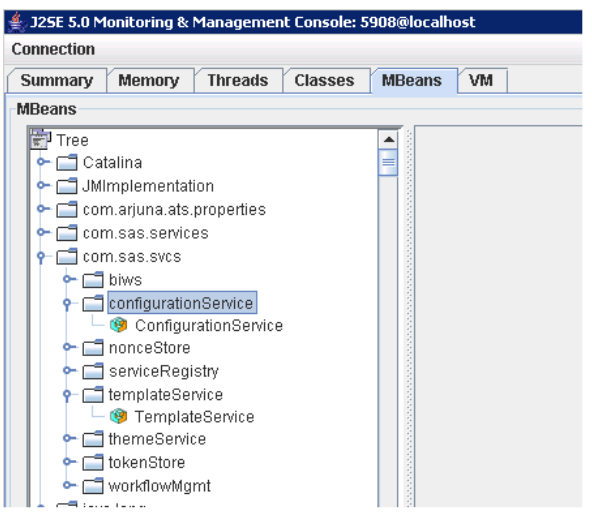


Figure 2. A JConsole View of Several of the SAS JMX Managed Beans

E-MAIL

The JavaMail API provides a portable, protocol-independent means for sending e-mail from a Java EE application server. Rather than each application defining their own interaction with the JavaMail API, the SAS middle tier defines an e-mail session on its Java EE application server instance(s). Individual applications use a simple interface to send e-mail when necessary and avoid the complexities of the basic definition.

Specific considerations, such as whether a Simple Mail Transfer Protocol (SMTP) requires authentication, are exposed in a limited number of locations. Individual applications are not required to be aware of all possible topologies or communication requirements.

The alert notification implementation sends e-mail messages to users when appropriate using this mechanism. Messages can also be sent to a site-defined administrator when some errors occur, or an administrator can send a message to all current users when needing to perform maintenance or other tasks.

IP MULTICAST

Internet Protocol (IP) multicast provides an efficient means of sending a network transmission to multiple interested receivers in a single message. IP multicast is a standard technology used to implement clustering requirements in Java EE application servers. SAS makes use of the JGroups library for its multicast implementation; this library is also utilized by JBoss Application Server clustering. Other application servers use a similar multicast implementation to support their own management.

The entire JGroups configuration is customizable. For example, a standard SAS middle-tier deployment sends multicast traffic using the User Datagram Protocol (UDP). A site can choose to use Transmission Control Protocol (TCP). For more information about using TCP for multicast, contact your on-site SAS support personnel.

A SAS middle-tier deployment is centered around a single SAS Metadata Server and a shared, unique multicast address and port. The middle tier uses multicast communication to share common information among all Web applications; prior to SAS 9.2, this information was repeated in each application, often with different formats.

The password-change process is a key beneficiary of this information sharing.

This method allows the information to be updated through the password tooling, and a simple server restart gets the new information to the applications that require it. Applications used to require individual changes, a complete rebuild of the archive, and finally a redeployment of the new archive to the Java EE application server.

SCALING, LOAD BALANCING, AND FAILOVER

The Web Infrastructure Platform contains an internal load balancing and failover mechanism. This uses the multicast group to communicate HTTP endpoint information for individual service types. When a service instance starts, it publishes its own Uniform Resource Locator (URL) information to the multicast group. Group members can see all available URLs for a particular service interface and will load balance and failover between multiple URLs for a single service type.

SECURITY

By default, the SAS middle tier sets up a secure multicast environment. All traffic is encrypted and therefore not viewable as plaintext for any network sniffer. The multicast group also requires an authentication token be presented in order to join. This token can be customized during the installation process and updated using the standard SAS update password tool.

In multicast communication, time-to-live (TTL) decides how “far” a multicast packet should be forwarded from the originating host. The SAS middle tier uses a default value of '1', which restricts traffic to the same subnet; this value is customizable during the installation process. For information about appropriate values for multicast TTL, refer to Meyer (1998).

STORAGE

The middle tier makes use of several storage mechanisms depending on the type of data, access required, and cross-application integration concerns.

SAS Metadata Server: The SAS Metadata Server is a multi-user server that serves one or more SAS[®] Metadata Repositories to all of the SAS Intelligence Platform client applications in your environment. It is a common denominator used and accessed by all applications in the SAS application suite and contains information used simultaneously by many applications.

The middle tier stores structured data in the SAS Metadata Server, which does not require transaction support. This data is read often but is written to rarely. For example, the application configuration and service registration information is stored in the SAS Metadata Server.

Relational Database: A relational database is used to store structured data that requires transaction support. The SAS middle tier supports several database vendors including IBM DB2, Microsoft SQL Server, MySQL, Oracle, PostgreSQL, and SAS Table Server.

Relational tables exist to support alert notifications, collaboration, and auditing, among other features.

WebDAV: Web-based Distributed Authoring and Versioning (WebDAV) is a set of extensions to HTTP that allows editing and other management of files on HTTP servers.

The middle tier uses WebDAV for the storage of data that does not have a strict data model and structure, including templates for e-mail and notification messages.

COMMON FEATURES AVAILABLE IN THE SAS MIDDLE TIER

The middle tier provides several Web applications implementing common features and requirements across the SAS application suite.

SAS LOGON MANAGER

The SAS Logon Manager is a Web application that provides the authentication challenge and single sign-on functionality for all SAS Web applications. All initial requests to the SAS environment are directed through the Logon Manager, whether a user is logging into a specific Web application (such as SAS[®] Web Report Studio) or selecting a link they received in a notification e-mail message.

This single front door allows a quicker and easier customization experience. These customizations might be as simple as changing the appearance or as complex as integrating a third-party single sign-on provider such as SiteMinder, WebSeal, or Netegrity.

The Logon Manager supports third-party single sign-on products through cooperation with the Java EE application server. In this scenario, the Java EE application server is configured with a strong authentication mechanism external to SAS, such as an LDAP server or a third-party provider's software. The SAS applications are then able to establish a connection into SAS servers through the trusted relationship that exists with the Java EE application server.

For further information about supported providers and their configuration, please see "SAS 9.2 Support for Web Application Servers and HTTP Servers" (2010).

SAS BI WEB SERVICES

The SAS BI Web Services application allows selection of one or more SAS Stored Processes to expose as a SOAP-based Web service. These Web services can then be called by a custom application to gain access to the stored process result. The programmatic interface is based on "XML for Analysis (XMLA) Specification, Version 1.1" (Microsoft Corporation, Hyperion Solutions Corporation 2002).

SAS CONTENT SERVER

The SAS Content Server provides a WebDAV repository for use by the SAS application suite to store content such as documents, reports, and images. This repository is integrated with the authentication and authorization mechanisms provided by the SAS Metadata Server, which results in a seamless management environment for SAS application content.

SAS WORKFLOW

SAS Workflow is a suite of applications and services that work together to model, automate, integrate, and streamline business processes, providing a platform for more efficient and productive business solutions. The primary purpose of SAS Workflow is to enable business process management within SAS applications. It can also be used to integrate two or more applications that are driven by human workflow.

CONFIGURATION MANAGER

The Configuration Manager is available in SAS Management Console. The Configuration Manager can perform administrative tasks such as configuring properties and values for SAS applications. The Configuration Manager offers a consistent interface to set properties for all SAS applications. Some of the values that are specified during installation and configuration can be modified at post-deployment time using this tool. Settings changed in the Configuration Manager are then exposed to applications via the configuration service.

SAS WEB ADMINISTRATION CONSOLE

The Web Infrastructure Platform also includes a Web-based administration console, which provides access to common tasks and system level information. The SAS Web Administration Console requires an administrative user ID in order to allow access.

Other SAS applications can provide their own administrative views to the console via a plug-in mechanism.

Viewing Current Users: The SAS Web Administration Console provides access to a list of the currently logged in users. From this list, the administrator can send e-mail to one or more of the users or force them off the system.

Managing the SAS Content Server: The SAS Content Server provides an administrative view that is used to manage content and associated tasks.

Performing System Maintenance: When a SAS Metadata Server restart or other system maintenance is required, the SAS Web Administration Console provides wizard interfaces to aid in the successful completion. These wizards

step through stopping any new logins, notifying current users of the impending stoppage, and finally allowing user logins once the maintenance is complete.

UTILITIES

The Web Infrastructure Platform provides several command line and graphical user interface (GUI) utilities for obtaining information about the system or making targeted changes. The scripts used to start these utilities are located in:

SAS-configuration-directory/Levn/Web/Utilities

DAVTree: The DAVTree utility is a Java application that displays a tree view of the SAS Content Server resources. The utility enables manipulation of current content and creation of new content.

Package Clean-Up: The Package Clean-Up utility is a simple command line interface for manipulating packages published via a publication channel or in the SAS Content Server. The utility displays basic information about a persisted package and can delete both the metadata and the actual package.

Quiesce: The SAS Remote Services contain a utility for stopping and resuming the ability for new users to log on. This utility can be used in conjunction with the system maintenance tools in the SAS Web Administration Console. It allows system maintenance to be performed in a batch manner.

The quiesce utility is located in the directory housing the SAS Remote Services:

SAS-configuration-directory/Levn/Web/Applications/RemoteServices

DEPLOYMENT TOPOLOGY

The SAS middle-tier infrastructure consists of several applications. This infrastructure is designed to give organizations the flexibility to distribute these components according to wide-ranging requirements. For small implementations, the middle-tier software and other SAS servers can all run on the same machine. In contrast, a large enterprise might have multiple servers that are distributed across multiple platforms. In addition, the components of the different tiers, such as Web applications that run in a Web application server, might be distributed on separate machines. The SAS middle tier makes such deployments possible — and in a high availability, clustered environment as well.

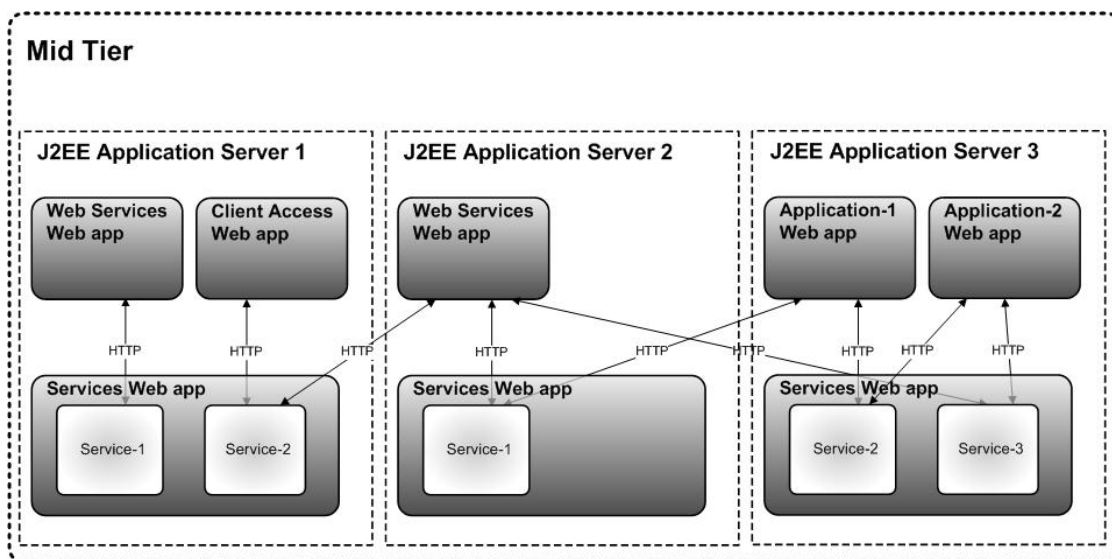


Figure 3. One Possible Deployment Topology Supported by SAS 9.2 Middle-Tier Components

For more information about clustered topologies, see Sweetland (2010).

SAS FOUNDATION SERVICES

The SAS® Foundation Services (also referred to as SAS Remote Services) runs as its own Java Virtual Machine (JVM) process and exposes core services to other applications via RMI. The SAS Remote Services serve as the single sign-on hub for SAS middle-tier applications and maintain shared state information.

WEB INFRASTRUCTURE PLATFORM

The Web Infrastructure Platform contains many core infrastructure services such as configuration, e-mail, and themes that are used by a wide range of SAS applications. These services are deployed to a Java application server as part of a Web archive (WAR) that is contained inside an enterprise archive (EAR). The following EAR and WAR files are deployed as part of the Web Infrastructure Platform:

sas.storedprocess9.2.ear	
sas.storedprocess.war	Is an application for executing SAS Stored Processes
sas.wip.apps9.2.ear	
sas.preferences.war	Provides a common application to manage individual user preferences
sas.svcs.logon.war	Contains the SAS Logon Manager
sas.wip.admin.war	Contains the SAS Web Administration Console
sasweb.war	Is a Web application that contains the SAS/GRAPH applet JARs and resources
sas.wip.scs9.2.ear	
sas.svcs.scs.war	Contains the SAS Content Server, a WebDAV implementation
sas.wip.services9.2.ear	
sas.biws.war	Hosts SAS BI Web Services to support deployment of SAS Stored Processes as a Web service
sas.wip.access.war	Provides Java client access to middle-tier services
sas.wip.services.war	Provides the services
sas.wip.soapservices.war	Provides access to services via SOAP for non-Java clients

SAS SHARED SERVICES

SAS Shared Services provides a set of services that are oriented toward common business intelligence (BI) functionality including user notifications and collaboration. These services are also deployed on a Java application server. They are delivered via a single EAR file:

sas.shared9.2.ear	
sas.shared.apps.war	Contains a common application for comment and collaboration management
sas.shared.services.war	Provides the shared services
sas.workflow.war	Hosts the engine for workflow process management
sas.workflow.webservice.war	Provides access to key workflow APIs via SOAP

CONCLUSION

The SAS middle tier — and specifically its Web Infrastructure Platform and SAS Shared Services components — provides an extensive set of features and capabilities. Common, centralized configuration is available across all SAS applications that have a Web or other middle-tier component via the configuration service. Integrated security services, including the authentication service, provide single sign-on capabilities for SAS Web applications and other clients. Cross-cutting concerns such as events processing and auditing are also addressed. Workflow capabilities support human interaction and loose coupling of SAS applications, which provides another avenue for integration.

When combined together, the result is simple and rather evident: the SAS middle tier is a true platform for delivering analytical reporting and other business intelligence in support of enterprise standards.

REFERENCES

- Albanna, Z., Almeroth, K., Meyer, D, and Schipper, M. 2001. "IANA Guidelines for IPv4 Multicast Address Assignments." Internet Engineering Task Force (IETF) document RFC3171. Available <http://tools.ietf.org/html/rfc3171>.
- Ban, B. 2010. "JGroups - A Toolkit for Reliable Multicast Communication." Available <http://www.jgroups.org/>.
- Chandran, Saravana R. 2010. "The Art of Managing the SAS 9.2 Middle Tier." *Proceedings of the SAS Global Forum 2010 Conference*. Cary, NC: SAS Institute Inc.
- Meyer, D. 1998. "Administratively Scoped IP Multicast." Internet Engineering Task Force (IETF) document RFC2365. Available <http://tools.ietf.org/html/rfc2365>.
- Microsoft Corporation, Hyperion Solutions Corporation. 2002. "XML for Analysis Specification, Version 1.1." Available <http://www.xmlforanalysis.com/>. Oracle/Sun Developer Network. 2010. Java EE at a Glance. Available <http://java.sun.com/javaee/>.
- Oracle/Sun Developer Network. 2010. JavaMail. Available <http://java.sun.com/products/javamail/>.
- Oracle/Sun Developer Network. 2010. Java Message Service (JMS). Available <http://java.sun.com/products/jms/>.
- Oracle/Sun Developer Network. 2010. Java SE Technologies - Database. Available <http://java.sun.com/javase/technologies/database/>.
- Oracle/Sun Developer Network. 2010. Java Transaction API (JTA). Available <http://java.sun.com/javaee/technologies/jta/index.jsp>.
- SAS Institute Inc. 2010. "SAS 9.2 Support for Web Application Servers and HTTP Servers." Available <http://support.sas.com/resources/thirdpartysupport/v92m2/appservers/>.
- Sweetland, Scott. 2010. "Scaling the SAS 9.2 EBI Web Tier with Multiple Managed Servers and Server Clusters." *Proceedings of the SAS Global Forum 2010 Conference*. Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Casey Hadden
Principal Software Developer
SAS Institute Inc.
100 SAS Campus Dr
R4472
Cary, NC 27513
E-mail: casey.hadden@sas.com

Marty Tomasi,
Director, Middle-Tier Services
SAS Institute Inc.
100 SAS Campus Dr
R4455
Cary, NC 27513
E-mail: marty.tomasi@sas.com

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.