SAS® Financial Management 5.3
System Administrator's Guide
Second Edition

Contents

Chapter 1
Overview

What's in This Book

This book is intended for system administrators of SAS Financial Management. It contains the following topics:

- post-configuration tasks

- loading sample data that can be used to verify the installation and demonstrate the software

- securing the configuration directories

- groups, roles, and capabilities that are required for SAS Financial Management tasks

- performance improvement by means of multiple query processors and secondary ODCS servers

    Note: For information about other performance improvements, see the *SAS Financial Management: Performance Guide*.

- configuring log files for SAS Financial Management

- administering MySQL

- configuring the SAS environment files

Required Skills

To administer SAS Financial Management, you must be familiar with the operating system on which it is installed. For example, you must know how to create folders, run

scripts, and update environment variables. On Microsoft Windows, you must be an administrator of the machine.

Documentation Conventions

This book uses the following documentation conventions to identify paths in the SAS Financial Management configuration:

Path	Refers to	Example
SAS-config-dir	Path to the SAS configuration directory	Windows: `C:\SAS\Config` UNIX: `/usr/local/SAS/Config`
MySQL-install-dir	Path to the MySQL installation directory	Windows: `C:\mysql` UNIX: `/usr/local/mysql`

Note:

- The name of the configuration directory and the level number might be different at your site.

- If your configuration is the result of a migration from the previous release of SAS Financial Management, the `SASApp` directory might be called `SASMain` instead (for example, `C:\SAS\Config\Lev1\SASMain` rather than `C:\SAS\Config\Lev1\SASApp`). Please make the appropriate substitutions as you read this book.

- File system pathnames are typically shown with Windows separators (\); for UNIX, substitute a forward slash (/).

- Some code examples contain line breaks, so that the code fits on the line. If you copy the code, remove the line breaks.

Additional Documentation

SAS Financial Management

For additional information about SAS Financial Management, see the appropriate versions of the following books:

- *SAS Financial Management: Performance Guide*

- *SAS Financial Management: Data Administrator's Guide*

- *SAS Financial Management: Data Model Reference*

- *SAS Performance Management Solutions: Migration Guide*

- *SAS Financial Management: User's Guide*

- *SAS Financial Management: Process Administrator's Guide*

- *Installation Instructions for SAS Financial Management Release 5.3*

- *MySQL Database Server Installation Guide for SAS Financial Management 5.3, SAS Human Capital Management 5.21, and SAS Strategic Management 5.3*

These books are available at the following site: **http://support.sas.com/ documentation/onlinedoc/fm**

Note: This site is password-restricted. You can find the user name and password in the preinstallation checklist or by contacting SAS Technical Support at **http:// support.sas.com/techsup/contact**.

SAS Intelligence Platform

For information about administering the SAS Intelligence Platform, see the following documents (and others) at **http://support.sas.com/93administration**:

- *SAS Intelligence Platform: System Administration Guide*

- *SAS Intelligence Platform: Security Administration Guide*

- *SAS Intelligence Platform: Middle-Tier Administration Guide*

- *SAS Intelligence Platform: Web Application Administration Guide*

- *SAS Intelligence Platform: Installation and Configuration Guide*

- *What's New in SAS 9.3 Intelligence Platform*

Third-Party Software

For information about administering third-party software, such as the Web application servers, see **http://support.sas.com/resources/thirdpartysupport/ v93**.

Chapter 2

Post-Configuration Steps

Overview

About the Post-Configuration Tasks

This chapter describes the post-configuration tasks that need to be performed for SAS Financial Management. It also describes how to load sample data that can be used to verify the installation and demonstrate the software.

For more information, see the online Help and user's and administrator's guides, as well as the *SAS Financial Management: Data Administrator's Guide*. (See "Additional Documentation" on page 2.)

Migration from a Previous Release

The topics in this chapter apply to migrations from a previous release, as well as new installations. However, do not install the sample data on a migrated system. Doing so would overwrite your existing data.

Important! Check SAS Notes

After installation and configuration, check the SAS Notes for additional information and support fixes. Go to the product page at **http://support.sas.com/software/ products/fm/index.html** and select the **Samples and Notes** tab. The product page also has links to documentation and available hot fixes.

In particular, be sure to read Problem Note 47049 (**http://support.sas.com/kb/ 47/049.html**) which contains instructions for ensuring the accuracy of queries in SAS Financial Management forms and reports. Depending on your hot fix level, these instructions might need to be implemented at your site.

Web Application Server Modifications

About the Managed Servers

Depending on your choices during the configuration steps, your system will have several managed servers. The following table shows the default servers and a partial listing of the contents, consisting of enterprise archive (EAR) files.

Server (Default Name)	Partial Contents (EAR Files)
SASServer1	The Web Infrastructure Platform (WIP), the Logon Manager, the SAS Information Delivery Portal, SAS BI Dashboard
SASServer2	The SAS WebDoc application, SAS Web Report Studio

Server (Default Name)	Partial Contents (EAR Files)
SASServer3	SAS Financial Management and Solutions Services
SASServer4	ODCS
SASServer5	Planning Data Entry

You might also have installed one or more secondary ODCS servers on additional managed servers. See "Secondary ODCS Servers for Operational Planning" on page 34.

SASServer3, SASServer4, SASServer5, and any secondary ODCS servers make up the SAS Financial Management managed servers.

The SAS Remote Services application must be started before the managed servers. If you restart SASServer1, you must restart the other managed servers as well.

For additional information, see the following publications:

- *SAS Financial Management: Performance Guide*
- *SAS Intelligence Platform: Middle-Tier Administration Guide*

Increase the Permanent Generation Size for SASServer1

Note: These instructions apply only to the IBM WebSphere Server (Solaris only) and to the Oracle WebLogic Server.

In the JVM options for the SASServer1 managed server, modify the PermSize and MaxPermSize options as follows:

```
-XX:PermSize=768m -XX:MaxPermSize=768m
```

Then restart SASServer1 and the other managed servers.

For information about setting JVM options, see *SAS 9.3 Web Applications: Tuning for Performance and Scalability*. (See "Third-Party Software" on page 3.)

Deploy the Cross-Domain Policy File, If Necessary

If the SAS Financial Management application (sas.financialmanagement5.3.ear) is located on a different physical machine from the Web Data Entry application (sas.financialmanagementwde5.3.ear), the site must deploy a cross-domain policy file that permits the Adobe Flash applications to communicate with one another.

During installation and configuration, if the two applications are installed on different physical machines, the necessary policy file is created in this folder on the middle tier: **SAS-config-dir\Lev1\Web\Applications \SASFinancialManagementWebDataEntry5.3**. The file contains the host name for the SAS Financial Management application and the port number for the Web Data Entry application.

1. Deploy the cross-domain policy file to the root context of the managed server for the Web Data Entry application.

2. Restart SASServer3 and the other SAS Financial Management managed servers.

For more information, see the technical guide for Cross Domain Security at **http:// learn.adobe.com/wiki/display/security/Application+Security +Library**.

Note: Post-configuration, if you move one application to a different machine, you might need to create the cross-domain policy file. You can find a template in this folder: **SASHOME\SASFinancialManagementWebDataEntry\5.3\Config \Deployment\CrossDomain**. (SASHOME is the name of the folder in which you installed SAS.) Edit the file and deploy it as described above.

JBoss Web Application Server Configuration

Change Definition of SASSDM Data Source

For JBoss, change the definition of the SASSDM data source from **Local Tx** to **XA**. This change avoids performance problems and error and warning messages in the log files.

1. Open the SASSDM_Data_Source-ds.xml file for editing.

   This file is located in the **jboss-eap-5.1\jboss-as\server \SASServer3\deploy** directory where you installed JBoss.

2. Replace the file contents with the following:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<datasource>
  <xa-datasource>
    <xa-datasource-class>
        com.mysql.jdbc.jdbc2.optional.MysqlXADataSource
    </xa-datasource-class>
    <xa-datasource-property name="relaxAutoCommit">true</xa-datasource-property>
    <xa-datasource-property name="connectionCollation">utf8_bin</xa-datasource-property>
    <xa-datasource-property name="useServerPrepStmts">false</xa-datasource-property>
    <xa-datasource-property name="pinGlobalTxToPhysicalConnection">true</xa-datasource-property>
    <xa-datasource-property name="ServerName">mysql-host-name</xa-datasource-property>
    <xa-datasource-property name="PortNumber">port-number</xa-datasource-property>
    <xa-datasource-property name="DatabaseName">sassdm</xa-datasource-property>
    <metadata>
      <type-mapping>MySQL</type-mapping>
    </metadata>
    <exception-sorter-class-name>
      org.jboss.resource.adapter.jdbc.vendor.MySQLExcepti onSorter
    </exception-sorter-class-name>
    <use-java-context>false</use-java-context>
    <isSameRM-override-value>false</isSameRM-override-value>
    <track-statements/>
    <check-valid-connection-sql>SELECT 1</check-valid-connection-sql>
    <security-domain>SSSEncryptDBPassword</security-domain>
    <jndi-name>SASSDM</jndi-name>
  </xa-datasource>
</datasource>
```

- *mysql-host-name* is the fully qualified host name of the machine where your MySQL server with the SASSDM database resides.

- *portnumber* is the port where MySQL is running. The standard port number is 3306.

3. Open the login-config.xml file for editing.

   This file is located in the **jboss-eap-5.1\jboss-as\server
   \SASServer3\conf** directory where you installed JBoss.

4. In the **SSSEncryptDBPassword** section, find the following line:

   ```
   <module-option name="managedConnectionFactoryName">
      jboss.jca:name=SASSDM,service=LocalTxCM</module-option>
   ```

   Change the line as follows:

   ```
   <module-option name="managedConnectionFactoryName">
      jboss.jca:name=SASSDM,service=XATxCM</module-option>
   ```

   Note: Line breaks added for readability. The file contains more than one module-
   option with the name of "managedConnectionFactoryName". Change only the
   option in the **SSSEncryptDBPassword** section.

5. Restart SASServer3 and the other SAS Financial Management managed servers.

Change the transactionTimeout Setting

The default transactionTimeout setting for the JBoss Web application server is 300
seconds. This value can be too short for some long-running operations, such as form set
publishing, cycle seeding, or loading dimensions with a large number of members. If
you experience a time-out, try increasing the ttransactionTimeout setting for the server
on which the time-out occurred.

Follow these steps:

1. Open the appropriate file for editing.

   The relative path is as follows:

   • For SASServer3: **jboss-eap-5.1\jboss-as\server
     \SASServer3\deploy\transaction-jboss-beans.xml**

   • For SASServer4: **jboss-eap-5.1\jboss-as\server
     \SASServer4\deploy\transaction-jboss-beans.xml**

   The absolute path depends on where you installed JBoss.

2. Find this line:

   ```
   <property name=transactionTimeout>300</property>
   ```

   The time-out value is displayed in seconds.

3. To begin with, increase the time-out value to 1800 seconds (30 minutes).

4. Save the file.

   You can increase the settings for both servers if you like, but the value for
   SASServer4 should not be greater than the value for SASServer3.

5. Restart SASServer3 and the other SAS Financial Management managed servers.

If time-outs continue to occur, further increase the time-out value, to a maximum of
14400 seconds (4 hours).

Give the nodeIdentifier Property a Unique Value

In order for rollbacks to work correctly across servers, the
com.arjuna.ats.arjuna.xa.nodeIdentifier property must have a unique value for each
server. For each of the servers except SASServer1, follow these steps:

1. In the **conf** directory, open the jbossts-properties.xml file for editing.

2. Find the following property:

   ```
   <!-- (Must be unique across all Arjuna instances.) -->
           <property name="com.arjuna.ats.arjuna.xa.nodeIdentifier" value="1"/>
   ```

3. Make the value for each server unique.

   For example, change the value for SASServer2 to "2", the value for SASServer3 to "3", and so on.

Save your changes and restart the servers.

Oracle WebLogic Server Configuration

Support Loading Large Dimensions

If you experience errors in loading dimensions with 100,000 members or more, make the following changes in your WebLogic settings for SASServer3 and SASServer4:

- In the **Settings for SASServer3** ⇨ **Protocols** ⇨ **HTTP**, change the **Post Timeout** value to 60 (seconds).

- In the **Settings for SASServer3** ⇨ **Protocols** ⇨ **General**, change the **Maximum Message Size** value to 90000000.

Restart SASServer3 and the other SAS Financial Management managed servers.

Set the Frontend Host Parameter

WebLogic seems to treat domains differently if they are referenced differently (for example, **http://Dxxx/yyy** and **http://Dxxx.mycompany.com/yyy**). This situation causes problems when a Web application stores information in the HttpSession context. There is a configuration parameter called **Frontend Host** that addresses this issue.

According to the WebLogic documentation, this parameter should be set when the Host information coming from the URL might be inaccurate due to the presence of a firewall or proxy. If this parameter is set, the HOST header is ignored and this value is used instead. The **Frontend Host** parameter is part of the HTTP Protocols for a managed server. For instructions about modifying these protocols, see the WebLogic documentation.

IBM WebSphere Server Configuration

Set Total Transaction Lifetime Timeout

The Total Transaction Lifetime Timeout value specifies the length of time, in seconds, for a transaction to be completed before it is rolled back. The default value is 120 seconds. If your configuration needs additional time, increase this value for SASServer3 and SASServer4, as follows:

1. Select **Container Services** ⇨ **Transaction Service**.

2. Specify a new value for **Total transaction lifetime timeout**.

   You might need to experiment by increasing the value considerably. If that works, try decreasing the value until you find a limit that works at your site.

3. If necessary, modify the value for **Maximum transaction timeout**.

This value specifies the length of time, in seconds, for all transactions that run in this server. Its value should be greater than or equal to the Total Transaction Lifetime Timeout value.

4. If necessary, modify the value for **Client inactivity timeout**.

    This value specifies the length of time, in seconds, between transactional requests from a remote client.

Restart SASServer3 and the other SAS Financial Management managed servers.

For more information about these parameters, see the online Help for the WebSphere administrative console.

Increase the Log File Size

The default sizes for WebSphere log files and history files might be too small to capture substantial logging. To change the log settings:

1. Log on to the WebSphere administrative console.

2. In the navigation tree, select **Troubleshooting Logs and Trace**.

3. On the Logging and Tracing page, click the server name.

4. Click **JVM Logs**.

5. For the System.out log, find the **File Size** for the **Log File Rotation**, and change the **Maximum Size** from 1 MB to 10 MB. You can adjust this value to suit your configuration.

6. To save log files that have been rotated, increase the value of **Maximum Number of Historical Log Files**.

7. Make the same changes for the System.err log.

8. Save your changes.

SAS Financial Management Application Options

(UNIX Only) Add umask to Stored Process Server Options

By default, in a UNIX installation, users do not have Write permission for files that they did not create. However, generating a forecast requires users to re-create a file that might have been originally created by another user. The following steps set the default permissions to be applied when a user creates a file via code that is running on the stored process server.

1. On the data tier, open the StoredProcessServer_usermods.sh file for editing.

    This file is located in the ***SAS-config-dir*/Lev1/SASApp/ StoredProcessServer** directory.

2. Just before the USERMODS_OPTIONS= line, add code similar to the following:

```
CURR_GID=`/usr/bin/id -g`
GID=sas
if [ $CURR_GID -eq $GID ]; then
  umask 002
fi
```

> *Note:* Use a version of the id command that supports the -g option. Use back ticks
> (not single quotation marks) around **/usr/bin/id -g**.

3. Save the file. The umask settings give the sas group Read and Write permission for new files.

Instead of applying the umask to the sas group, you could create a group that includes all users who will be generating forecasts. Make it the primary group for these users, and substitute that group name for "sas" in the code above.

(Optional) Change the CTA Behavior for Intercompany Transactions

SAS Financial Management supports two methods of accounting for cumulative translation adjustments (CTAs) as they relate to intercompany transactions:

- **Elimination of CTA amounts related to intercompany transactions.** This is the default behavior. It assumes that revaluation of intercompany balances as a result of exchange rate fluctuations occurs within a customer's source accounting system. In SAS Financial Management, any translation adjustments that arise as the result of intercompany transactions are eliminated in the originating organization's functional currency via Intercompany Eliminations.

- **Persistence of CTA amounts related to intercompany transactions.** This functionality ignores CTA amounts related to intercompany transactions, allowing values to persist without being eliminated. This behavior might be necessary when revaluation of intercompany balances does not occur in a customer's source accounting system and reporting currency differs from the functional currency of the lowest common parent.

Choose a method based on a customer's practices for recording and managing intercompany transactions and balances. Selecting the appropriate method results in balanced, consolidated results.

The behavior is determined by a system property, odcs.cta.elim.behavior. The default value of this property, **TransactionCurrency**, corresponds to the elimination of CTA amounts related to intercompany transactions. To change this behavior so that it persists CTA amounts that are related to intercompany transactions, follow these steps:

1. Add the following argument to the JVM options for the ODCS managed servers:

   ```
   -Dodcs.cta.elim.behavior=ReportingCurrency
   ```

2. Restart the ODCS managed servers.

(Optional) Change the Default for Trader Security

> *Note:* This option applies only to member-level security for financial cycles. It does not affect object-level security settings for an Organization dimension.

In a financial cycle, the Trader dimension type mirrors the Organization dimension type. In the record of a two-organization transaction, the trader member identifies the second organization.

Members of a Trader dimension inherit the security settings of the corresponding Organization dimension members. As a result, a user who was associated with a member of one organization could not view the facts for transactions with a second organization.

As of SAS Financial Management 5.2, member-level security by default is effectively ignored in Trader dimensions, so that users can view facts for transactions with other organizations.

In previous releases, the Trader security settings (inherited from the Organization dimension) were honored. To revert to the previous behavior, follow these steps:

1. Add the following argument to the JVM options for the ODCS managed servers:

   `-Dodcs.member.security.ignoreTrader=false`

2. Restart the ODCS managed servers.

Note: We recommend that you enable Trader security (by setting **ignoreTrader** to **false**) only if you do not set member-level security on organization members.

(Optional) Change the Materiality Threshold

By default, automatic and explicit allocations, driver formulas, and most adjustment rules are subject to an absolute value threshold. The default threshold value is 0.001. To set a different threshold value, follow these steps:

1. Add the following argument to the JVM options for SASServer3 (the managed server where SAS Financial Management is running):

   `-Dfms.threshold.materiality=`*new-value*

   where *new-value* is a value of 0.0 or greater. If *new-value* is 0.0, no threshold is applied.

   Note: The number of decimal places in the threshold value has no effect on the precision of the output values.

2. Restart SASServer3 and the other SAS Financial Management managed servers.

For more information, see the *SAS Financial Management: Process Administrator's Guide*.

SAS PC Files Server

Installing SAS PC Files Server is mandatory for loading data from 32-bit PC files into 64-bit SAS. With this configuration, data administrators can use 32-bit Microsoft Excel or Microsoft Access files as input to jobs in SAS Data Integration Studio.

Note: If you use Microsoft Office 2007, you must install the Microsoft Office 2007 ODBC driver on the machine where you installed SAS PC Files Server. To download the driver, see the following SAS Note: **http://support.sas.com/kb/37/521.html**.

For information about SAS PC Files Server, see the *SAS/ACCESS Interface to PC Files* at **http://support.sas.com/documentation**.

Secure Your Installation

Overview

This section contains information about setting operating system protection for configuration directories, for both Windows and UNIX.

Additional information:

- For an overview and detailed information about security in the SAS Intelligence Platform, see the *SAS Intelligence Platform: Security Administration Guide*.

- If you installed SAS Web Report Studio, see "Configuring SAS Web Report Studio" in the *SAS Intelligence Platform: Web Application Administration Guide*. This chapter includes information about securing the folders that are used by SAS Web Report Studio.

(Windows) Security Settings for Configuration Directories

SAS Intelligence Platform

First, apply the operating system protections that are recommended for configuration directories on the SAS Intelligence Platform. For instructions, see "What to Do Next: Administration Tasks" in the *SAS Intelligence Platform: System Administration Guide*.

To configure security for configuration directories, follow these steps:

SAS Financial Management

For SAS Financial Management, apply the additional protections that are recommended below. These directories are located in the **SAS-config-dir\Lev1** directory on the data tier.

Table 2.1   Windows: Recommended Permissions for SAS Financial Management Directories

Directories	Permissions
Under **SASApp\SASEnvironment \FinancialManagement** and **SASApp \SASEnvironment\SolutionsServices**: **SASCode\Jobs** **SASFormats** **SASMacro**	Grant Modify permission to the SAS Server Users group.
SASApp\Data and its subdirectories	Grant Full Control to SAS General Server User (sassrv) . Grant Read/Write/Modify permission to users who run SAS Data Integration Studio jobs to update data in the data warehouse. These users should include the Solutions Host User.

MySQL Permissions

On the host where you installed MySQL, apply the following permissions:

Table 2.2 *Windows: Recommended Permissions for the MySQL Directories*

Directories	Permissions
`MySQL-install-dir`	Grant Full Control to SYSTEM and Administrators only.
`MySQL-install-dir\bin`	Grant Read and Execute permissions to the Everyone group.
	(During installation and configuration, you were asked to give this group Read, Execute, and Modify permissions. After installation and configuration, restrict these permissions.)

(UNIX) Security Settings for Configuration Directories

SAS Intelligence Platform

First, apply the operating system protections that are recommended for configuration directories on the SAS Intelligence Platform. For instructions, see "What to Do Next: Administration Tasks" in the *SAS Intelligence Platform: System Administration Guide*.

SAS Financial Management

Apply the additional permissions that are recommended in the following table. All of these directories are located in the `SAS-configuration-directory/Lev1` directory on the data tier.

Table 2.3 *UNIX: Recommended Permissions for SAS Financial Management Directories*

Directories	Recommended Permissions
Under `SASApp/SASEnvironment/ FinancialManagement` and `SASApp/ SASEnvironment/SolutionsServices`: `SASCode/Jobs` `SASFormats` `SASMacro`	Permit full access for the sas user ID and the sas user group
`SASApp/Data` and its subdirectories	Permit full access for the sas user ID and the sas user group.

(Optional) Secure Access to MySQL

During the configuration process, several MySQL users are created, and the root user for MySQL is deleted after it is no longer needed.

Note: On Windows, MySQL is installed as a system service by default. Consequently, the service has access to all directories. MySQL can be used only with its own user IDs.

To restrict the IP address that MySQL uses, perform these steps after the configuration has been validated:

1. On the machine where MySQL resides, create a file (grant.sql) with these contents (line breaks are inserted for readability):

```
revoke all privileges, grant option from 'sqladmin'@'%';
GRANT ALL PRIVILEGES ON *.* TO sqladmin@'datatier'
    IDENTIFIED BY 'mysqlpassword'
    WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON *.* TO sqladmin@'midtier'
    IDENTIFIED BY 'mysqlpassword'
    WITH GRANT OPTION;
GRANT ALL PRIVILEGES ON *.* TO sqladmin@'localhost'
    IDENTIFIED BY 'mysqlpassword'
    WITH GRANT OPTION;
flush privileges;
```

In this file, make the following changes:

a. Replace *mysqlpassword* with the password for sqladmin.

b. Replace *datatier* and *midtier* with the fully qualified host names of the data tier and middle tier, respectively.

Save the file.

2. At a command prompt, execute this command (replacing *mysqlpassword* with the password for sqladmin):

```
mysql -usqladmin -pmysqlpassword < grant.sql
```

Follow the same procedure for additional MySQL users.

(Optional) Verify Using Sample Data

Overview

Sample data is provided to help you verify the correct operation of the software and to demonstrate functionality. After verification, you can run batch programs to load the StageFM and SASSDM databases again. These batch programs reset the databases to their default state (their state immediately following a new installation and configuration).

CAUTION:
If this installation is the result of a migration, or if you have already stored data that you want to keep, do not load the sample data or reset the databases.

Here is an overview of the verification procedure:

1. Load the sample data.

2. Restart the managed servers.

3. Verify the installation.

4. Reset the databases to their default state.

5. Restart the managed servers.

Note: If you are also installing SAS Human Capital Management, see the *SAS Human Capital Management 5.2 Administrator's Guide, Second Edition* for information about installing sample data for that product. That book is available at **support.sas.com/documentation/onlinedoc/hcm/index.html**. Otherwise, do not run the LoadDDSSampleData script. That script loads the DDS database, which is no longer used by SAS Financial Management.

Load Sample Data

Load Sample Data for the SAS Financial Management Staging Area

This step replaces the tables in the StageFM library.

1. Log on to the data-tier server.

2. At a command prompt, change directory to **SAS-config-dir \Lev1\Applications\SASFinancialManagement5.3\SampleData**.

   Note: Logs for these commands are written to a subdirectory of the current working directory.

3. Run one of the following scripts to load the StageFM sample data:

   - Windows: LoadFMSampleData.bat

   - UNIX: LoadFMSampleData.sh

4. When prompted, enter the following connection information:

Prompt	Description
SAS Metadata User ID	User ID of the unrestricted user (for example, sasadm@saspw)
SAS Metadata User Password	User password

Load Sample Data for SAS Financial Management Data Mart

Running the LoadSASSDMSampleData script populates tables in the SAS Financial Management Data Mart (SASSDM).

1. At a command prompt, change directory to **SAS-config-dir \Lev1\Applications\SASSolutionsServices5.3\SampleData**.

2. Run one of the following scripts to load the sample data:

   - Windows: LoadSASSDMSampleData.bat

   - UNIX: LoadSASSDMSampleData.sh

3. When prompted, enter the following connection information:

Prompt	Description
MySQL Host Name	Name of the host machine where MySQL is running.
MySQL DB Name	**sassdm**
MySQL Port (default:3306)	MySQL port number. 3306 is the default port number.

Prompt	Description
MySQL DB User ID	User ID for accessing the SASSDM database.
MySQL DB User Password	Password for accessing the SASSDM database.

If you installed the MySQL server on a different machine, the batch file prompts you to run the scripts manually. Follow these steps:

1. Copy the following files to the machine where MySQL is running:

    • sassdm.sql

    • fmsdata.sql

    These scripts are located in the **SAS-install-dir \SASSolutionsServicesDataTier\5.3\SampleData\sassdm** directory on the data tier.

2. On the MySQL machine, run the commands as you were instructed in the batch file output.

    Note: Replace the source directory name in the commands with the new source directory on the MySQL machine.

Update Sample Data

After loading the sample data, review SAS Note 45897 to see if you need to update the sas_etl_group_actions and sas_etl_user_actions tables. See **support.sas.com/kb/ 45/897.html**.

Verify the Installation

About SAS Financial Management Verification

To verify the Web application components of SAS Financial Management, follow the steps in the Instructions.html file on the middle tier, logging on as the SAS Demo User.

SAS Financial Management Studio

Note: For installation instructions, see the *Installation Instructions for SAS Financial Management Release 5.3*.

Follow these steps:

1. Restart SASServer3 and the other SAS Financial Management managed servers.

    Otherwise, your tests will fail.

2. Log on to SAS Financial Management Studio as the SAS Demo User (sasdemo).

3. Examine the dimensions in the Dimensions workspace.

4. Expand the Account dimension and view these sample hierarchies:

    • ACCOUNT_MR

    • ACCOUNT_Sal

    • ACCOUNT_TH

5. Examine the default model in the Models workspace.

SAS Financial Management Add-In for Microsoft Excel

Note: For installation instructions, see the *Installation Instructions for SAS Financial Management Release 5.3*.

To verify the operation of the SAS Financial Management Add-In for Microsoft Excel, follow these steps:

1. Open a workbook in Microsoft Excel.

2. From the SAS Financial Management menu, select **Log On** and log on to the middle tier as the SAS Demo User (sasdemo).

3. Insert a read-only table:

   a. Select **SAS Solutions ⇨ Insert ⇨ Read-only table**.

   b. Select the **Default_Model**.

   c. Use the arrows to make the following selections:

      • **Rows**: Account

      • **Columns**: Time

      • **Slicers**: Analysis, Organization, Frequency, and Product

   d. Make additional selections in the rest of the wizard or click **Finish** to accept the defaults.

Reset the Databases

On the data tier, follow these steps to reset the StageFM and SASSDM databases to their default state:

1. At a command prompt, change directory to ***SAS-config-dir* `\Lev1\Applications\SASSolutionsServices5.3\SampleData`**.

2. You must update some variables in the restore script before executing it. The scripts are as follows:

   • Windows: restore_original_install_data_fm.bat

   • UNIX: restore_original_install_data_fm.sh

   Make a backup copy and then edit the appropriate script, modifying the following variables as necessary for your site configuration.

   Note: When you modify the script, replace *SAS-config-dir* and **`!sasroot`** with the appropriate file system path. Replace **`SASApp`** and **`Lev1`** as necessary for your site. In the SET statements, do not use spaces around the equal sign.

Variable	Description
SAS_EXE	Full path (including filename) to the sas executable, located in the **`!sasroot`** directory.
	For an explanation of the **`!sasroot`** directory, see the *SAS Companion for Windows* or the *SAS Companion for UNIX Environments* at **`support.sas.com/ documentation`**.
SAS_DIR (UNIX)	Path to the **`!sasroot`** directory. In UNIX scripts, this variable is used in the definition of SAS_EXE.

Variable	Description
SAS_CONFIG (Windows) or SOL_CFG (UNIX)	Full path (including filename) to the SASV9.cfg file that is located in the **SAS-config-dir\Lev1\SASApp** directory.
REG_STAGEDDS_DDS	Full path (including filename) to the register_stagedds_and_dds.sas file, which is located in the **SAS-config-dir\Lev1\Applications \SASSolutionsServices5.3\SASCode** directory.
REG_COMMON	Full path (including filename) to the register_common.sas file, which is located in the **SAS-config-dir \Lev1\Applications \SASSolutionsServices5.3\SASCode** directory.
REG_ODCS	Full path (including filename) to the register_odcs.sas file, which is located in the **SAS-config-dir \Lev1\Applications \SASSolutionsServices5.3\SASCode** directory.
REG_FM	Full path (including filename) to the register_fm.sas file, which is located in the **SAS-config-dir \Lev1\Applications \SASFinancialManagement5.3\SASCode** directory.
REG_OP	Full path (including filename) to the register_opplan.sas file, which is located in the **SAS-config-dir \Lev1\Applications \SASFinancialManagement5.3\SASCode** directory.
REG_STAGEFM	Full path (including filename) to the register_fm.sas file, which is located in the **SAS-config-dir \Lev1\Applications \SASFinancialManagement5.3\SASCode** directory.

3. Run the script:

   • Windows: restore_original_install_data_fm.bat

   • UNIX: restore_original_install_data_fm.sh

   When prompted, enter the following connection information:

   Note: The UNIX script omits "MySQL" in the prompts.

Prompt	Description
MySQL Host Name	Name of the host machine where MySQL is running
MySQL DB PORT	Port number for MySQL (default: 3306)
MySQL DB NAME	Database name (default SASSDM)
MySQL DB User ID	User ID for accessing the SASSDM database
MySQL DB Password	Password for accessing the SASSDM database

Prompt	Description
META USER ID	User ID of the unrestricted user (for example, sasadm@saspw)
META USER PASSWORD	Metadata user password

The batch program runs several SAS registration programs. The logs from these programs are in the **`SAS-config-dir`\Lev1\Logs\Configure** directory.

If you installed the MySQL Server on a different machine, you are prompted to run the scripts manually. Follow these steps:

1. On the machine where MySQL is installed, run the command to drop and then create the SASSDM database, as you were prompted by the batch file output.

2. On the data-tier machine, run the following .sas commands as you were prompted by the batch file output.

   - register_common.sas

   - register_odcs.sas

   - register_fm.sas

   - register_opplan.sas

After you run the scripts, restart SASServer3 and the other SAS Financial Management managed servers.

Create the Site's Users and Groups

Register users at the site and assign them to groups and roles. For instructions, see Chapter 3, "Assigning Groups and Roles," on page 23.

Load Production Data

For instructions about loading production data, see the *SAS Financial Management: Data Administrator's Guide*.

Chapter 3

Assigning Groups and Roles

Overview of Users, Groups, Roles, and Capabilities

This chapter describes the default users that are defined when you install SAS Financial Management, as well as the groups, roles, and capabilities that are required to use SAS Financial Management.

Additional information:

- For detailed information about authentication and authorization, see the *SAS Intelligence Platform: Security Administration Guide*.

- The SAS Intelligence Platform configures a default set of users, groups, and roles during the deployment process. For information about those identities, see "Understanding the State of Your System" in the *SAS Intelligence Platform: System Administration Guide*.

These books are available at **http://support.sas.com/93administration**.

Default User Identities

SAS Financial Management has two default user identities: Solutions Host User and SAS Solutions Administrator.

Identity	Description
Solutions Host User	This identity has JDBC access to the MySQL databases, depending on group membership and operating system permissions. It is an external identity.
SAS Solutions Administrator	The SAS Solutions Administrator is an internal identity that is used for cases in which a user must perform a query as a part of a larger process, but the query requires a role that the user does not generally need. Rather than requiring the user to be assigned to that role, the application recognizes the SAS Solutions Administrator as a user with the proper role to successfully complete the process.
	This identity performs special tasks within SAS Financial Management and should have ReadMetadata, Read, WriteMetadata, Write, and Create permissions for SAS Financial Management content. Do not use this identity as a login identity.
	In a system that was migrated from SAS 9.1.3, this user is called Solutions Role Administrator.

Groups

SAS Financial Management uses the following predefined groups: SASSDM MySQL Users and Administrators. In addition, sites typically create their own groups to meet their needs (for example, for assigning roles or access permissions to folders).

Group	Description
SASSDM MySQL Users	This group has a login to the SASSDM database. Users need to belong to this group only if they run stored processes and jobs that directly write to MySQL tables in the SASSDM database.
	The following administrative stored processes require membership in the SASSDM MySQL Users group:
	• Import Users and Groups
	• ETL Job Status
	The following SAS Data Integration Studio jobs require membership in the SASSDM MySQL Users group:
	• solnsvc_1300_load_users
	• solnsvc_1400_load_groups
	• solnsvc_1500_load_user_x_group
	If you create a stored process from a SAS Data Integration Studio Job and include a reference to etlstatus.sas, then any user who runs the stored process must belong to the SASSDM MySQL Users group.

Group	Description
Administrators	The Administrators group confers superuser status on its members. Users who belong to the Administrators group are not subject to security restrictions on objects such as cycles, models, and dimensions. These users still need the appropriate capabilities, and member security still applies.
	Use caution in assigning users to this group.

Note: SAS Financial Management 5.3 does not have a predefined Solutions Users group.

Capabilities and Roles

About Capabilities and Roles

Role membership determines the actions that a user can perform in an application, as well as the menus and links that are displayed. Each role is assigned one or more capabilities, such as the ability to manage dimensions or approve forms.

Capabilities do not function as data security. Use role assignments to enable the actions that a user can perform and the menu items that are available to a user. Use data security, such as metadata permissions, to restrict the data that a user can access.

For more information about roles and capabilities, see the "Security Overview" and "User Administration" chapters of the *SAS Intelligence Platform: Security Administration Guide*. In particular, see the section entitled "Role-Based Access to Application Features."

SAS Financial Management Capabilities

Description of Capabilities
The following capabilities are available for SAS Financial Management. Each capability confers the ability to perform specific actions, such as creating and managing cycles and periods.

Table 3.1 *SAS Financial Management Capabilities*

Category	Capability	Description
Administrative	Cycle Administration	Perform all actions in the Periods workspace and Cycle Manager of SAS Financial Management Studio.
	Dimension Administration	Perform all actions in the Dimensions workspace of SAS Financial Management Studio.
	Form Administration	Perform all actions in the Forms workspace and Form Manager of SAS Financial Management Studio.
		Create and update form templates in the SAS Financial Management Add-In for Microsoft Excel.
		Create reports in the SAS Financial Management Add-In for Microsoft Excel.
	Model Administration	Perform all actions in the Models workspace of SAS Financial Management Studio.
	Rate Administration	Perform all actions in the Rates workspace of SAS Financial Management Studio.
	Security Administration	Assign security permissions for objects (such as cycles and models) and dimension members in SAS Financial Management Studio.
		Access the Administration workspace of SAS Financial Management on the Web.
Planning	Approve Financial Forms	Approve financial forms.
		Create reports in the SAS Financial Management Add-In for Microsoft Excel.
	Submit Financial Forms	Submit and enter data into financial forms.
		Create reports in the SAS Financial Management Add-In for Microsoft Excel.
	Approve And Submit Operational Forms	Approve, submit, and enter data into operational forms.
Reporting	Create and Edit Reports	Create reports in the SAS Financial Management Add-In for Microsoft Excel.
		With this capability, users can access the Reports workspace of SAS Financial Management.
	View Reports	View SAS Financial Management reports.
		In Microsoft Excel, this capability enables users to change the view of a table. However, it does not enable them to copy the table, create a new table, or publish a report or information map.
		With this capability, users can access the Reports workspace of SAS Financial Management.

Capabilities are additive. They grant the ability to perform one or more actions. They do not deny the ability to perform an action.

Having a capability does not confer access to data. For example, a user might have the Model Administration capability but be denied access to specific models because of object security. A user might have the Submit Financial Forms capability but not be able to view the contents of all the cells in a form because of member-level security.

Capabilities for SAS Financial Management Studio

To log on to SAS Financial Management Studio, users must have at least one of the following capabilities: Cycle Administration, Dimension Administration, Form Administration, Model Administration, or Rate Administration. Users have full access to the workspaces that are associated with their capabilities, and they have Read access to the other workspaces.

Capabilities for SAS Data Integration Studio Jobs

The Administrative capabilities also enable users to execute SAS Data Integration Studio jobs. The following capabilities are required:

Table 3.2   Capabilities for SAS Data Integration Studio Jobs

Job	Required Capabilities
Load Facts	Cycle Administration
Export Facts	Model Administration
Load Cell Protection Rules	
Load Member Selection Rules	
Load Supplemental Schedule Details	Cycle Administration
Load Supplemental Schedule Facts	
Load Exchange Rates	Rate Administration
Load Driver Rates	

Default Roles

SAS Financial Management has a set of default roles, each with preassigned capabilities. Each site can create additional roles with the capabilities that are required at that site.

Table 3.3   SAS Financial Management: Default Roles

Role Name	Display Name	Capabilities
Finance Process Administrator	Financial Management: Process Administrator	all SAS Financial Management capabilities
Finance Adjuster	Financial Management: Model Administrator	Model Administration Excel Reporting View Reports
Form Approver	Financial Management: Form Approver	Approve Financial Forms View Reports

Role Name	Display Name	Capabilities
Form Submitter	Financial Management: Form Submitter	Submit Financial Forms View Reports
Planning Data Entry User	Financial Management: Planning Data Entry User	Approve and Submit Operational Forms

Registering Users

About Registering Users

For information about registering users, see the *SAS Intelligence Platform: Security Administration Guide*.

When you define a user, be sure to include the user's e-mail address. E-mail notifications are often sent to users. For the successful processing of some functions, you must define an e-mail address for every user.

Note: On Windows, in order for users to access a standard workspace server using credential-based host authentication, they need the local **Log on as a batch job** right on that machine. For more information, see "Windows Privileges" in the *SAS Intelligence Platform: Security Administration Guide*.

Synchronizing Users and Groups

About Synchronizing Users and Groups

Information about users and groups is stored in database tables that must be kept in synchronization with the metadata. As a part of best practices, it is recommended that you set up a SAS Data Integration Studio job as a scheduled process to synchronize data tables.

Running a Batch Job

The typical way to update user and group information is by using a batch job:

1. In SAS Data Integration Studio, create a batch job that includes these three jobs:

   • solnsvc_1300_load_users

   • solnsvc_1400_load_groups

   • solnsvc_1500_load_user_x_group

2. Schedule this job to be performed on a regular basis.

For more information about these jobs, see the *SAS Financial Management: Data Administrator's Guide*. For information about creating batch jobs and about scheduling jobs, see the *SAS Intelligence Platform: System Administration Guide*.

Running the Import Users and Groups Stored Process

In some cases, changes to users and groups might need to be reflected in the database as soon as those changes are made in the metadata. In that case, you can run the jobs manually, rather than waiting for the scheduled process to run. If you need to synchronize on demand, follow these steps:

1. In a Web browser, log on to the SAS Information Delivery Portal.

    Note: You must be a member of the SASSDM MySQL Users metadata group.

2. If the **Import Users and Groups** stored process is not already available, add it to a portlet such as the Collection portlet.

    This stored process is located in the **Products** ⇨ **SAS Financial Management** ⇨ **5.3 Standard Reports** folder.

3. Run the stored process.

4. Check the stored process log to make sure that the stored process ran successfully. (You might not see an error message in your Web browser.) The stored process log files are located at **SAS-config-dir\Lev1\SASApp\StoredProcessServer \Logs**.

For changes in roles, you might also need to flush the user cache, as described in the next topic.

Clear User Cache

For performance reasons, roles and capabilities are cached when the Web application server is started. Use this option if there have been changes to capabilities or roles and the Web application server has not been restarted. This cache affects users' eligibility to edit or review forms, or to be assigned as form authors or reviewers.

To flush the cache, follow these steps:

1. From a Web browser, log on to SAS Financial Management as a user with the Security Administration capability.

2. Select the **Administration** workspace.

3. From the categories menu on the left, select either **Users** or **Groups**.

4. Click the Clear the user cache button �֍.

Chapter 4

Performance Improvement via Multiple Query Processors and Secondary ODCS Servers

Overview

Depending on your site's configuration and needs, the following performance improvements are possible in SAS Financial Management:

- **Multiple query processors**: If you have a site in which many users are executing queries at the same time, you can add query processors to an ODCS managed server. This option does not improve the processing time for a single query. However, it increases the number of queries that can execute simultaneously.

  See "Multiple Query Processors for Multi-User Environments" on page 32.

- **Secondary ODCS servers**: To make operational planning more efficient, you can add secondary ODCS servers to your configuration. When a form set is published, its forms are distributed to the available secondary servers.

  See "Secondary ODCS Servers for Operational Planning" on page 34.

For information about other types of performance improvement, see the *SAS Financial Management: Performance Guide*.

Multiple Query Processors for Multi-User Environments

Overview

SAS Financial Management supports an optional configuration in which multiple query processors can be attached to an ODCS managed server. With this configuration, ODCS jobs are automatically routed to various machines where the ODCS query processor is running.

The benefit of this configuration is that it maximizes query processing concurrency in multi-user environments. It is most effective in situations where there are many concurrent users executing small simultaneous queries. Although it does not improve the performance of individual queries, it does allow more queries to execute simultaneously and thus improves overall throughput.

Configure ODCS Target Machines

In a configuration with multiple query processors, each target machine must be network-accessible from the ODCS server, and each target machine must have installed the Java Development Kit (JDK). (For the correct JDK version, see the system requirements document.)

To configure a target machine:

1. Create a directory on the target machine.

2. Copy each JAR file from the top-level directory of the exploded ODCS application to the target directory.

   The exploded EAR files are in the **SAS-config-dir\Lev1\Web\Staging \exploded** directory.

3. On the target machine, run the query processor:

   ```
   java -classpath jarfiles -Xms800m -Xmx800m -Xss256k _
   -Dlog4j.configuration= _
   file:///C:/SAS/Config/Lev1/Web/Common/LogConfig/SASODCS-log4j.xml _
   -Dodcs.dispatcher.host=host QueryProcessor
   ```

   Note: Line breaks ("_") added for readability.

   • *jarfiles* refers to the JAR files that you copied in the previous step. If you list each file, separate the JAR filenames with semicolons.

     Alternatively, you can use a wildcard to include all the JAR files in a particular directory; for example:

     ```
     java -classpath "D:\ODCS_QP\*" ...
     ```

   • **-Xms** and **-Xmx** determine the initial and maximum Java heap size. For best results, these values should be the same. The optimal heap size varies from customer to customer and from site to site.

   • **-Xss** represents the thread stack size. In a 32-bit environment, the recommended value is 256K. In a 64-bit environment, 512K.

- The **-Dlog4j.configuration** option specifies the path to the log4j configuration file.

  You can use a configuration file similar to that used by ODCS. Take care to set the appender output to a file that is designated for that query processor's logging. We recommend using different appender files for ODCS and for each query processor.

- The **-Dodcs.dispatcher.host** option specifies the name of the machine on which the ODCS application is running.

  You can include additional options, in the form **-Doption=value**, as described in the table below.

Table 4.1   *Query Processor Options*

Option	Description and Default Value
odcs.dispatcher.port	The TCP/IP port on which the in-process RMI registry is hosted by ODCS and through which the query processors make the bootstrap contact. The default is **9876**. If you change this port number, you must also add or modify the **odcs.dispatcher.port** JVM option for the ODCS managed server (SASServer4).
odcs.dispatcher.passkey	The password key handshake between the query processor and the dispatcher. If the passkey does not match, the query processor cannot connect to the dispatcher to run queries. The passkey must be specified by both the dispatcher and the query processor. The default value is **passkey**.
odcs.queryprocessor.maxthreads	The number of CPUs that are available on the machine that hosts the query processor. Because the algorithms are CPU-bound, adding more threads than physical CPUs causes context switching and degrades performance. The default is *number of available processors*.
odcs.queryprocessor.reattach	If this value is set to **false** (the default), then the query processor shuts down when the ODCS dispatcher stops running. If the value is **true**, then the query processor waits for the dispatcher to start again and reattaches to the dispatcher immediately. In a production environment, reattaching might be practical. In a development environment, the typical reason for shutting down the ODCS server is to modify the JAR files. As a result, reattaching would result in a ClassCastException.

When a query processor is started, it checks to see whether the ODCS server is running. If so, it attaches to the server and waits for the server to send it jobs to process. Otherwise, the query processor waits until the ODCS server starts and then attaches to the server. When the ODCS server is shut down, all the query processors are shut down as well.

Configure ODCS Server Options

The ODCS server (the managed server on which the ODCS application is running) acts as the dispatcher. When you start the server, you can pass it any of the optional arguments that are listed in the table below. Use the syntax **-D*option=value***.

Table 4.2 *ODCS Server Options*

Option	Description and Default Value
`odcs.dispatcher.port`	The TCP/IP port on which the in-process RMI registry is hosted by ODCS and through which the query processors make the bootstrap contact. The default port number is 9876.
`odcs.dispatcher.passkey`	The password key handshake between the query processor and the dispatcher. If the passkey does not match, the query processor is not allowed to connect to the dispatcher to run queries. The passkey must be specified by both the dispatcher and the query processor—for example, by passing this argument to the command lines of both the ODCS server and the query processor: `-Dodcs.dispatcher.passkey=mysecretpassword` The default is **passkey**.
`odcs.dispatcher.ipfilter`	A comma-separated list of Internet addresses of machines that are allowed to connect. If you specify such a list and a query processor tries to connect to an IP address that is not in the list, the connection is rejected. There is no default.
`odcs.dispatcher.use.internal.qp`	If **true** (the default), the dispatcher uses the built-in internal query processor, in addition to any external query processors that might be available. There are benefits to running queries locally. In-process queries do not require the data to be serialized to them. Moreover, if you configure only a few external query processors, then the ODCS server might be better used to share the query load, in addition to the data and dispatch. If this argument is **false**, the dispatcher does not process any queries locally, so that it is always available to route queries to external query processors. This mode is useful if you have a large number of query processors.

Secondary ODCS Servers for Operational Planning

How Secondary Servers Are Used

In SAS Financial Management, operational planning is supported by a configuration that includes a primary ODCS server and, as an option, one or more secondary ODCS

servers. The main benefit of secondary servers is to distribute the load for the planning process.

Note: Secondary ODCS servers are not used by financial planning.

When a form set is published, the forms are distributed based on the ODCS servers that are available. In a configuration with two secondary ODCS servers, the forms are divided between the servers, as in this simplified diagram:

Figure 4.1   *Form Set Distribution with Two Secondary ODCS Servers*
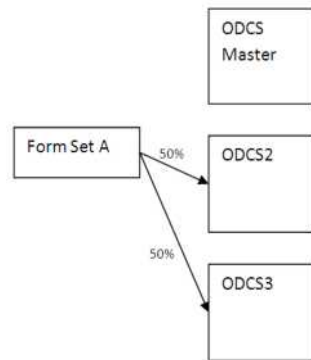
By default, the primary (master) ODCS server is not a target of the distribution, although this default can be changed. (See "Configure the Primary ODCS Server to Share the Load" on page 37.)

Install an Additional Secondary ODCS Server

During installation and configuration, secondary ODCS servers are established, based on your plan file. Typically, each ODCS EAR runs on a different host, to take full advantage of the additional memory and processing power, and each ODCS EAR references its own local SAS_FACT table. Each ODCS EAR also has an ODCSDS data source that points to the SASSDM database on the primary server.

Post-installation, you can create additional secondary ODCS servers. Follow these steps:

Note: If you are replacing an existing server, see "Replace a Secondary ODCS Server" on page 37.

1. Make sure that users have logged off.

   Via the SAS Web Administration Console, you can send an e-mail message to users, notifying them of your intention to shut down the managed servers. You can also force users to log off. For more information, see "Administering the SAS Web Infrastructure Platform" in the *SAS Intelligence Platform: Web Application Administration Guide*.

2. Shut down the ODCS managed servers and the managed server to which you deployed SAS Financial Management (by default, SASSERVER3).

3. In the SASSDM database for the primary ODCS server, view the records in the SAS_ODCS_SERVER table. Note the host names shown in the SERVER_HOSTNAME column. If the new secondary server's host name matches a name that is already in this column, change the host name of the new server before proceeding. (Do not delete or edit the existing database record.)

4. Install the secondary server on the target machine.

Use your original plan file and follow the instructions under "Deployment of the SAS OLAP Data and Compute Server (ODCS) Secondary Mid-Tier" in the installation guide for SAS Financial Management.

Note: SAS Financial Management Web Data Entry is also deployed to the secondary ODCS Server.

5. If you configured any JVM options for your primary ODCS server, configure the same options for your new secondary server.

6. Start the managed servers that you shut down.

Effects of Installing an Additional Secondary Server

It is important to understand the effect of installing a new secondary ODCS server. On the host machine for the secondary server, the installation script creates a database called SASSDM, containing a single SAS_FACT table. It also creates at least two data sources:

• FACTDS, pointing to the local SASSDM database

• ODCSDS, pointing to the primary ODCS server's full SASSDM database

The new server automatically registers itself with the primary ODCS server. As a result, the new server is visible to SAS Financial Management when it distributes forms during the next publish operation for a form set. Data for a form set that is already published continues to reside with its original secondary server (the work load is not rebalanced). However, the new secondary server is included in the list for future publish operations.

This diagram shows an additional secondary server, ODCS4. The forms that are part of form set A, which was published before ODCS4 was added, continue to reside on ODCS2 and ODCS3. However, form set B was published after ODCS4 was installed. As a result, one-third of its forms are distributed to each of the three secondary ODCS servers.

Figure 4.2   *Form Set Distribution after Addition of a Secondary ODCS Server*
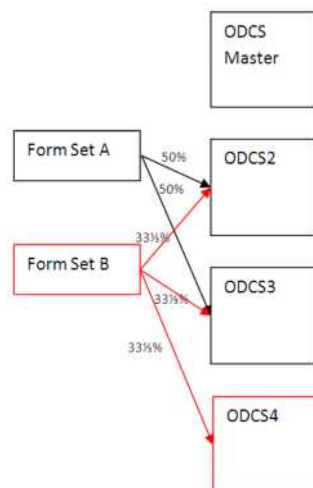
If you republish a form set and select the option to delete existing data, then the republish operation can also take advantage of a new secondary server.

Configure the Primary ODCS Server to Share the Load

By default, in an installation with both primary and secondary ODCS servers, the primary ODCS server is used for financial planning, and operational planning is performed on the secondary ODCS servers.

If the site does very little financial planning, you might want the primary ODCS server to share in the load, so that it can also be a target for published forms. To configure this behavior:

1. Add the following option to the JVM options for the managed server for SAS Financial Management (by default, SASServer3):

   ```
   -Dop.forms.useMaster=true
   ```

2. Restart the managed server for SAS Financial Management (by default, SASSERVER3) and all ODCS managed servers.

Note: In a configuration with no secondary servers, all forms are published to the primary ODCS server.

Replace a Secondary ODCS Server

It is possible for a new secondary server to replace an existing secondary server and take over its current form sets. Follow these steps:

1. Make sure that users have logged off.

   Via the SAS Web Administration Console, you can send an e-mail message to users, notifying them of your intention to shut down the managed servers. You can also force users to log off. For more information, see "Administering the SAS Web Infrastructure Platform" in the *SAS Intelligence Platform: Web Application Administration Guide*.

2. Shut down the ODCS managed servers and the managed server to which you deployed SAS Financial Management (by default, SASSERVER3).

3. In the SASSDM database for the primary ODCS server, view the records in the SAS_ODCS_SERVER table.

   Note the host names shown in the SERVER_HOSTNAME column. If the new (replacement) secondary server's host name matches a name that is already in this column, change the host name of the new server before proceeding. (Do not delete or edit the existing database record.)

4. Install the secondary server on the target machine.

   Use your original plan file and follow the instructions under "Deployment of the SAS OLAP Data and Compute Server (ODCS) Secondary Mid-Tier" in the installation guide for SAS Financial Management.

   Note: SAS Financial Management Web Data Entry is also deployed to the secondary ODCS Server.

5. If you configured any JVM options for your primary ODCS server, configure the same options for your new secondary server.

6. Copy the existing server's SASSDM.SAS_FACT table to the replacement server's SASSDM database:

   a. At a command prompt on the source machine, dump the SAS_FACT table:

```
cmd> mysqldump sassdm sas_fact -hhost_name1
     -uusername -ppassword > sas_fact.sql
```

b. At a command prompt on the target machine, drop the SAS_FACT table (which should be empty) and replace it with the table from the source machine:

```
cmd> mysql -hhost_name2 -uusername -ppassword
     mysql>  drop table sassdm.sas_fact;
     mysql> exit;
cmd> mysql -hhost_name2 -uusername -ppassword sassdm < sas_fact.sql
```

7. On the primary ODCS host, edit the SASSDM.SAS_ODCS_SERVER table.

In the record for the old server, replace the SERVER_HOSTNAME and SERVER_PORT values with the host name and port number of the new secondary server. Be sure to enter the full host name (including the domain). Do not use aliases, shortened names, or apparently equivalent names.

Do not change the SERVER_ID value (which is referenced by other tables) or the SERVER_TYPE value. (A SERVER_TYPE of **0** signifies a secondary server; a SERVER_TYPE of **1**, a primary server.)

8. Save your changes.

9. Start the managed servers that you shut down.

Note: Only the primary ODCS server should have a SASSDM.SAS_ODCS_SERVER table.

Referring to Figure 4.1 on page 35, if you replaced server ODCS3 with a server named ODCS4, the result would be as follows:

Figure 4.3 *Result of Replacing a Secondary ODCS Server*
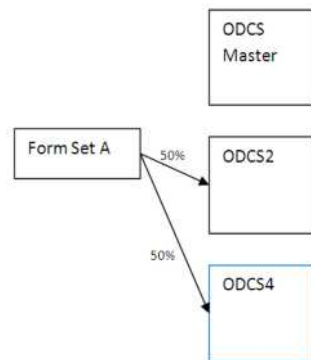
Server ODCS4 would continue to process the forms that were originally on the ODCS3 server.

Chapter 5
Log Files

Log File Locations and Configuration Files

The log files for SAS Financial Management and other applications are configured using log4j.xml files in the **SAS-config-dir\Lev1\Web\Common\LogConfig** directory. The default log file location is **SAS-config-dir\Lev1\Web\Logs**.

The log files for the Web application servers have the following default locations:.

Table 5.1 *Web Application Server Log Files*

Web Application Server	Log File Information
Oracle WebLogic servers	By default, the log files for the Oracle WebLogic servers are located in subdirectories under **SAS-config-dir\Lev1\Web\SASDomain \servers\server-name\logs** directory, where *server-name* is the name of the managed server. Output to the WebLogic console is written to these log files.
IBM WebSphere servers	By default, the log files for the IBM WebSphere servers are located in the **profiles/profile-name/logs/server-name** directory where you installed WebSphere.
JBoss servers	By default, the log files for the JBoss servers are located in the **jboss-as\server\server-name\log** directory where you installed JBoss.

Configure a Log File for the SAS Financial Management Reports

On the data tier, you can configure a log file to be written when a stored process uses the **Javaobj** interface. **Javaobj** is a mechanism that is similar to Java Native Interface (JNI) for instantiating Java classes and accessing their methods and fields. This mechanism is used by the standard reports that are shipped with SAS Financial Management.

You can also use this log file to capture output from SAS Data Integration Studio jobs that invoke SAS code.

To log this output, follow these steps:

1. Create a log4j.properties file that is similar to the following:

```
# Hierarchy: DEBUG < INFO < WARN < ERROR < FATAL
log4j.appender.A1=org.apache.log4j.FileAppender
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.layout.ConversionPattern=%d{MM-dd HH:mm:ss,SSS} [%t][%-5p%c{1}] - %m%n

# Modify the path to the log file as needed
log4j.appender.A1.File=c:/tmp/logs/javaapi.log
log4j.appender.A1.Append=False
log4j.rootLogger=INFO, A1
log4j.rootCategory=INFO, A1

# for FM API logging, uncomment this line
# log4j.category.com.sas.solutions.finance.api=DEBUG

# for SAS Data Integration Studio logging, uncomment this line
# log4j.category.com.sas.solutions=DEBUG
log4j.category.com.sas.solutions.finance=DEBUG
```

2. Add the following line to the JREOPTIONS of the sasv9_usermods.cfg file:

   ```
   -Dlog4j.configuration=file:/c:/log4j.properties
   ```

   The configuration file is located in the **SAS-config-dir\Lev1\SASApp** directory.

   Be sure to add the option within the parentheses that specify the JRE options. Modify the path to the log4j.properties file as needed.

Log Files for Client Applications

SAS Financial Management Studio writes log messages to a file called client.log, in a location that is relative to the Java system property **user.home**. This property represents the user's home directory (in this case, the user who is logged on to Windows).

If you are running Windows Vista and logging is not working correctly, edit the application's INI file and set **user.home** explicitly, like this:

```
-Duser.home=C:\Users\userid
-data
@user.home\SAS\FinancialManagementStudio\5.3
```

userid represents the Windows user.

For information about logging from SAS Data Integration Studio, see "Configure a Log File for the SAS Financial Management Reports" on page 40. For information about configuring a log file for the Excel add-in, see "The SAS Financial Management Add-In API for Microsoft Excel" in the *SAS Financial Management: Customization Guide*.

Temporary Files for the Excel Add-In

The SAS Financial Management Add-In for Microsoft Excel saves temporary files in the **My SAS Files\FMTempFiles** directory relative to the Excel default save location.

Do not delete temporary files from this directory while Excel is running.

Additional Log Files

- For information about log files for other Web applications, such as SAS Web Report Studio and SAS Web OLAP Viewer, see the *SAS Intelligence Platform: Web Application Administration Guide*.

- For information about log files that are generated by the SAS servers, see "Enabling Server Logging" in the *SAS Intelligence Platform: System Administration Guide*.

Both books are available at **support.sas.com/93administration**.

Chapter 6
Administering MySQL

MySQL Overview

SAS Financial Management stores common data in a MySQL database that is created during the installation process. Support for INNODB tables must be enabled within MySQL to provide transaction support.

For information about installing MySQL, see the *MySQL Database Server Installation Guide for SAS Financial Management 5.3, SAS Human Capital Management 5.21, and SAS Strategic Management 5.3* at **http://support.sas.com/documentation/ onlinedoc/fm**.

Windows: Reconfiguring MySQL

MySQL is configured to read its configuration settings from the **MySQL-install-dir \my.ini** configuration file. If you need to adjust your MySQL configuration, you can modify these configuration settings in the MySQL Administrator, or you can edit the my.ini file directly. Before you make any changes, be sure to make a backup copy of the my.ini file. After making your changes, restart the service.

The MySQL client reads its configuration information from a copy of the my.ini file that is located in the Windows root directory (for example, **C:\WINNT\my.ini**). If you modify the **MySQL-install-dir\my.ini** file, be sure to copy your modified file to the Windows root directory.

UNIX: MySQL Configuration

Excessive I/O in MySQL might be an indicator that sorts are not occurring in memory. In that case, you might consider increasing sort_buffer_size. See the MySQL reference manual for considerations in changing this parameter, which is allocated per thread. As a

test, you might temporarily set sort_buffer_size as high as 32M. However, a lower permanent setting might be more appropriate. To modify a parameter setting, edit the **MySQL-install-dir/my.cnf** file and restart MySQL.

At Solaris sites with heavy data usage, you can improve performance by editing the my.cnf file to set the thread_concurrency value. This value is used in determining the number of threads to be run simultaneously. The recommended value is as follows:

```
number-of-cpus * (2..4)
```

Chapter 7
Configuring the Environment Files

Overview

About the SAS Environment Files

SAS Financial Management uses two environment files:

- **sas-environment.xml** applies to SAS Financial Management Studio.

- **EnvironmentFactory.xml** applies to the SAS Financial Management Add-In for Microsoft Excel.

These files enable desktop client applications to determine the location of required services on the middle tier and to obtain a list of services available in the environment.

When you log on SAS Financial Management Studio, you are asked to select a SAS environment in which the application will operate. The **SAS environment** drop-down box contains a list of available environments. You select an environment and connect to the middle tier for that environment.

When you log on to SAS Financial Management from Microsoft Excel, you are presented with a similar dialog box.

For information about specifying these values during an installation, see "Installing Client Applications" in the installation guide.

See Also

Deploying the SAS Environment Files to an HTTP Server

For the purposes of validating the installation and configuration at a site, use the default sas-environment.xml and EnvironmentFactory.xml locations. After validation, you can deploy those files to a location where they are accessible by all possible clients. The recommended deployment location is an HTTP server.

You must deploy the sas-environment.xml and EnvironmentFactory.xml files to a central location if any of the following situations is true:

- At your site, SAS Financial Management is used in multiple environments, such as development, test, and production environments, and you want your users to be able to select from a list of available environments.

- Secure Sockets Layer (SSL) is configured for SAS Financial Management at your site.

- Web authentication is configured for SAS Financial Management at your site. (In this case, the central location must be an HTTP server.)

Follow these steps:

1. Deploy the sas-environment.xml file to the HTTP server.

   This file is located in the **`SAS-config-dir\Lev1\Web\Common`** directory.

2. Deploy the EnvironmentFactory.xml file to the HTTP server.

   This file is located in the **`SAS-config-dir\Lev1\Web\Applications \SASSolutionsServices5.3`** directory.

3. On each client machine, update the URLs to the environment files:

   - **SAS Financial Management Add-In for Microsoft Excel:** Change the URL so that it points to the HTTP server. For example:

     ```
     [Environment Factory]http://myhttpserver:port/EnvironmentFactory.xml
     ```

   - **SAS Financial Management Studio:** By default, the location of the sas-environment.xml file is specified in the **`!SASHOME\sassw.config`** file on the client machine. This setting might also apply to other SAS desktop applications.

     If you want the new location to apply only to SAS Financial Management Studio, add the **`-Denv.definition.location`** option to the INI file. For example:

     ```
     -Denv.definition.location=http://myhttpserver:port/sas-environment.xml
     ```

     If you want the new location to apply to all SAS desktop applications that use sas-environment.xml, update the sassw.config file instead. For example:

     ```
     SASENVIRONMENTSURL=http://myhttpserver:port/sas-environment.xml
     ```

   Note: The INI files are located in the **`5.3`** directory under the directory in which you installed each client application.

4. If your end users perform their own client installations, make them aware of the URLs to these files.

Defining Additional Environments

sas-environment.xml

The SAS environment file (sas-environment.xml) applies to SAS Financial Management Studio.

If you want your end users to select from multiple SAS environments, modify the sas-environment.xml file that you deployed to an HTTP server. Include information about each of the available environments.

For more information, see "Configuring the SAS Environment File" in the *SAS Intelligence Platform: Middle-Tier Administration Guide*.

EnvironmentFactory.xml

The EnvironmentFactory.xml applies to the SAS Financial Management Add-In for Microsoft Excel. By default, the EnvironmentFactory.xml file defines a single environment that represents the installation in which it resides. The file has the following structure:

```
<environments>
  <environment name="environment-name">
    <object>
      ...
      </object>
      ... [additional object definitions]
  </environment>
  ... [additional environment definitions]
</environments>
```

The *environment-name* is an identifier that appears in the selection list when a user logs on to the middle tier from Microsoft Excel.

To customize the EnvironmentFactory.xml file to support more than one configuration of the SAS Financial Management, follow these steps:

1. Open the EnvironmentFactory.xml file that you deployed to an HTTP server.

2. Copy the default environment definition (from **<environment name="default">** through **</environment>**) and paste it directly after the **</environment>** tag of the first entry.

3. Give this second environment definition a new environment-name, such as **test**.

   Names must be valid as XML attributes.

4. Modify the server name (and port number, if necessary) of each entry for the **test** environment definition.

   In a WebLogic configuration, the result might resemble the following:

Example Code 7.1 *EnvironmentFactory.xml Example with Two Environments*

```
<environments>
  <environment name="default">
    <object>
      <name>default</name>
      <java.naming.factory.initial>
          weblogic.jndi.WLInitialContextFactory
      </java.naming.factory.initial>
      <java.naming.provider.url>
          t3://server1:7201
      </java.naming.provider.url>
    </object>
...
    </environment>

    <environment name="test">
      <object>
      <name>default</name>
      <java.naming.factory.initial>
          weblogic.jndi.WLInitialContextFactory
      </java.naming.factory.initial>
      <java.naming.provider.url>
          t3://server2:7201
      </java.naming.provider.url>
    </object>
  ...
    </environment>
</environments>
```

Note: Change only the environment name and URLs. Notice that
`<name>default</name>` is the same in both environments.

5. Follow steps 2–4 (with a different environment name and different server names) to create an environment definition for the production environment.

6. Save your changes.

Note: The EnvironmentFactory.xml file should be synchronized with the sas-environment.xml file. That is, if you define multiple environments for SAS Financial Management Studio, you should define comparable environments for the add-ins.

Appendix 1

Port Usage

The servers in the SAS Intelligence Platform communicate with clients and other servers using TCP/IP. Each server listens on a particular port or ports for incoming requests. During installation, the SAS Deployment Wizard enables you to either accept the default ports or to specify different port numbers for some servers.

For a list of default ports, see the "Default SAS Ports" appendix of the *SAS Intelligence Platform: System Administration Guide*. In addition, the following default port numbers apply to SAS Financial Management:

Table A1.1   *Default Port Numbers and Descriptions*

Port Number	Description
25	SMTP mail: Port used by mailhost or Simple Mail Transfer Protocol (SMTP). Used to send administrative e-mail notices and end-user alert notifications.
3306	Database server port. All JDBC access from the managed servers goes through this port to the MySQL server. SAS/ACCESS to MySQL also uses this port.

Your site might use different port numbers than the defaults. For a complete list, see the pre-install checklist for your site.

Index