



SAS[®] Federation Server Manager 4.2: User's Guide, Second Edition

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2017. *SAS® Federation Server Manager 4.2: User's Guide, Second Edition*. Cary, NC: SAS Institute Inc.

SAS® Federation Server Manager 4.2: User's Guide, Second Edition

Copyright © 2017, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

February 2017

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.

P1:fedsrvmgrug

Contents

<i>What's New in SAS Federation Server Manager</i>	<i>v</i>
PART 1 Introduction to SAS Federation Server Manager	1
Chapter 1 • Overview of SAS Federation Server Manager	3
About SAS Federation Server Manager	3
Chapter 2 • SAS Federation Server Security Features	5
About SAS Federation Server Security Features	5
About Authorizations	6
Understanding DSNs and Permissions	8
Table, Column, and Row-Level Security	10
Overview	11
PART 2 SAS Federation Server Manager Interface	13
Chapter 3 • Accessing SAS Federation Server Manager	15
Log On to SAS Management Console	15
SAS Federation Server Manager Home Page	16
Specifying Preferences	17
Chapter 4 • Navigating SAS Federation Server Manager	19
Layout of the Interface	19
Federation Server Objects and Views	19
PART 3 Configuring SAS Federation Server Manager	25
Chapter 5 • Post-Installation Configuration	27
Post-Installation Configuration	27
Chapter 6 • Configuring SAS Federation Server	35
Configuring a Federation Server	35
User and Group Authorizations	39
Granting Permissions for Federation Server and Associated Objects	41
Chapter 7 • Configuring Access to Data Sources	45
Working with Data Services	45
Creating a Data Service	46
Using the Generic Data Service Template	68
Editing a Data Service	69
Working with Data Source Names (DSNs)	69
Establishing DSN Permissions	71

PART 4 Working with Federated Data 73

Chapter 8 • Working with Catalogs and Schemas	75
Catalogs and Schemas	75
Chapter 9 • Working with FedSQL Views	81
Working with FedSQL Views	81
Chapter 10 • Caching Data	85
Caching Views	85
Refreshing Cached Data	90
Chapter 11 • Configuring Row-Level Security	97
Column and Row-Level Security	97
Defining User Functions for Row-Level Security	101

PART 5 Advanced Topics 105

Chapter 12 • Working with the Console	107
SAS Federation Server Manager Console	107
Working with Information Views	109
Chapter 13 • Working with DS2 Dialect	111
Overview of DS2 on SAS Federation Server	111
Create a DS2 DSN	111
DSN and DS2 Object Permissions	112
Chapter 14 • Data Quality and Cleansing Functions	113
About Data Quality on SAS Federation Server	113
Data Quality Functions in SAS Federation Server Manager	120
Chapter 15 • Data Masking	123
Overview	123
Displaying Data Masking Functions in the Console	125
Data Masking in a FedSQL View	125
Data Masking Examples	126
Chapter 16 • SQL Logging	131
About SQL Logging	131
Enable SQL Logging	131
Viewing the SQL Log	132
SQL Logging Transactions	137

What's New in SAS Federation Server Manager

Overview

The following new features are available in this version of SAS Federation Server and SAS Federation Server Manager:

- new SAS Federation Server driver that allows access to SAS Scalable Performance Data Server.
- replacement of DataFlux Authentication Server by SAS Metadata Server for authentication and other permission-based functions
- support for SAS DS2 model scoring code in your database
- enhanced data masking and encryption support
- cache enhancements that include cache refresh for data held in memory (MDS)
- embedded data quality and cleansing functions in data views
- Read/Write access to Hadoop (HIVE) using the SAS Federation Server Driver for Apache Hive
- new SAS Federation Server driver that allows shared data sources across multiple SAS Federation Servers.

SAS Federation Server Driver for SPD Server

SAS SPD Server tables can be accessed for reading, writing, and update by SAS Federation Server with the SAS Federation Server Driver for SPD Server (Driver for SPD Server). The Driver for SPD Server provides connectivity from a SAS Federation Server on any host to an SPD server running anywhere. The Driver for SPD Server integrates with UNIX hosts as well as Windows. See [“SPDS Data Service”](#) in “Working with Data Services”.

SAS Metadata Server

SAS Metadata Server replaces DataFlux Authentication Server as the authentication provider. SAS Metadata Server provides access for user and group objects, as well as

other permission-based functions such as shared logins and trusted users. SAS Federation Server objects are populated from objects created in SAS Metadata Server. There is no longer a need to create federation server objects in SAS Federation Server Manager. See the *SAS Federation Server: Administrator's Guide* for information about how SAS Federation Server works with SAS Metadata Server.

SAS DS2 Language Support

DS2 is a SAS proprietary programming language that is used for advanced data manipulation. DS2 provides capabilities not available through SQL, such as scoring models. You can also use DS2 code to run data quality functions using SAS Federation Server Manager. See [“Data Quality Functions in SAS Federation Server Manager”](#) for additional information.

New Data Masking Rules

New data masking features include TRANC, which transliterates characters from the input string to characters in the output string. A series of random data masking rules are also available. See the [Overview of Data Masking](#) for information about each of these new rules and their functionality in SAS Federation Server Manager.

Enhanced Cache Operations

Federation Server now has the capability of refreshing cached data in MDS after a server restart. In previous releases, cached data that was held in memory was deleted if the server was restarted or shut down. Now the cache held in memory is retained and refreshed at server restart by default. See [“Caching a FedSQL View”](#) for information about configuring this option.

Data Quality and Cleansing

You can now use QKB with SAS Federation Server. The data quality methods are installed with your SAS Federation Server and are available in SAS Federation Server Manager through the Console. See [“About Data Quality on SAS Federation Server”](#).

SAS Federation Server Driver for Apache Hive

A new data service for the SAS Federation Server Driver for Apache Hive is available in SAS Federation Server Manager. Connection options are outlined in the “Driver Reference” of the *SAS Federation Server: Administrator's Guide*. To configure the

driver in SAS Federation Server Manager, see [“SAS Federation Server Data Service”](#) in [“Working with Data Services”](#).

SAS Federation Server Driver

A new data service for the SAS Federation Server Driver (FEDSVR) is available in SAS Federation Server Manager. The SAS Federation Server driver enables you to define a connection from one SAS Federation Server to another SAS Federation Server. Connection options are outlined in the “Driver Reference” of the *SAS Federation Server: Administrator’s Guide*. To configure the driver in SAS Federation Server Manager, see [“SAS Federation Server Data Service”](#).in [“Working with Data Services”](#).

Part 1

Introduction to SAS Federation Server Manager

Chapter 1

Overview of SAS Federation Server Manager 3

Chapter 2

SAS Federation Server Security Features 5

Chapter 1

Overview of SAS Federation Server Manager

About SAS Federation Server Manager	3
Introduction	3
About SAS Federation Server Manager	3

About SAS Federation Server Manager

Introduction

SAS Federation Server Manager is a web application that enables administrators to configure and secure access to SAS Federation Server and its associated data sources. With SAS Federation Server Manager, you can create data services and data source names (DSNs) for connecting to data sources supported by SAS Federation Server.

Most of the functions performed with SAS Federation Server Manager can be accomplished with administration DDL statements. The *SAS Federation Server: Administrator's Guide* contains information about administration DDL.

Note: The *SAS Federation Server: Administrator's Guide* provides the concepts that are needed to complete tasks in SAS Federation Server Manager.

About SAS Federation Server Manager

SAS Federation Server Manager is the editor used to set up and manage SAS Federation Server. With SAS Federation Server Manager, you can also configure and secure access to a SAS Federation Server. Using SAS Federation Server Manager, an administrator can create data services and DSNs that connect to data sources supported by any SAS Federation Server.

You can access SAS Federation Server Manager through a browser session. SAS recommends that you work within only one browser session of SAS Federation Server Manager. Do not access SAS Federation Server Manager from multiple browser sessions. Operations that take place in one session might not be reflected in subsequent sessions.

Chapter 2

SAS Federation Server Security Features

About SAS Federation Server Security Features	5
Overview	5
Levels of Security Access (Permissions)	6
About Authorizations	6
Overview	6
About the Authorizations Tab	7
Understanding DSNs and Permissions	8
CONNECT Permission	8
Federation Server SQL Authorization Enforcement	9
About the Data Source Names Tab	9
Table, Column, and Row-Level Security	10
Overview	10
Column and Table Privileges	10
Overview	11

About SAS Federation Server Security Features

Overview

Properly configured security for SAS Federation Server ensures that both the server and its data are secure. Data is protected against unauthorized access, and can be guaranteed secure transmission for transferring data. SAS Federation Server security configuration and maintenance are easily managed with SAS Federation Server Manager. You can assign security for these server objects:

- Federation Server
- Connections: Data Service and DSN
- DS2 package functions and threads
- Catalogs and schemas
- Tables and columns
- Views

Levels of Security Access (Permissions)

For each user or group, you can grant or deny these permissions:

- SELECT
- UPDATE
- INSERT
- REFERENCES
- DELETE
- ALTER TABLE or VIEW
- DROP TABLE or VIEW
- CREATE TABLE or VIEW
- CREATE, ALTER CACHE
- CREATE TABLESPACE
- CREATE DSN
- CONNECT
- EXECUTE

By default, users are not granted any permissions. The SYSTEM user or a SAS Federation Server administrator must grant privileges so that users can perform actions and gain access to data. Group permissions are granted and denied in the same manner as individual users. Users who are members of a group inherit the permissions from the group unless explicitly denied in the individual user account. At a minimum, any user who accesses any objects on SAS Federation Server must have CONNECT permissions. See (DSN Permissions) for information. See [“Granting Permissions for Federation Server and Associated Objects”](#) for additional information.

About Authorizations

Overview

Authorization determines what privileges a user or group object contains in order to gain access to resources and associated data sources. An object that might require security has an associated **Authorizations** tab. For example, a federation server object has an **Authorizations** tab that displays permissions for the users and groups associated with a particular SAS Federation Server.

Figure 2.1 SAS Federation Server Manager Authorizations Tab

Permission	Setting	Securable	Grantee	Grantor
SELECT	Grant	FEDERATION SERVER	user1	ADMINISTRATOR
UPDATE	Grant	FEDERATION SERVER	user1	ADMINISTRATOR
INSERT	Grant	FEDERATION SERVER	user1	ADMINISTRATOR
REFERENCES	Deny	FEDERATION SERVER		
DELETE	Deny	FEDERATION SERVER		
ALTER TABLE	Deny	FEDERATION SERVER		
ALTER VIEW	Deny	FEDERATION SERVER		
DROP TABLE	Deny	FEDERATION SERVER		
DROP VIEW	Deny	FEDERATION SERVER		
CREATE TABLE	Deny	FEDERATION SERVER		
CREATE VIEW	Deny	FEDERATION SERVER		
CREATE CACHE	Deny	FEDERATION SERVER		
ALTER CACHE	Deny	FEDERATION SERVER		
CREATE TABLESPACE	Deny	FEDERATION SERVER		
ADMINISTER	Deny	FEDERATION SERVER		
TRACE	Deny	FEDERATION SERVER		
CREATE DSN	Deny	FEDERATION SERVER		
CONNECT	Grant	FEDERATION SERVER	user1	ADMINISTRATOR
EXECUTE	Grant	FEDERATION SERVER	user1	ADMINISTRATOR

About the Authorizations Tab

Overview

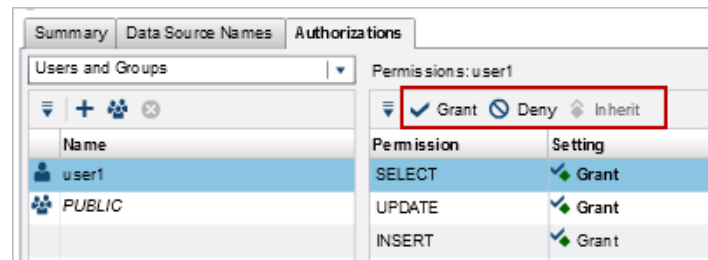
You can grant user or group permissions on the **Authorizations** tab for a selected object. The **Authorizations** tab contains information for users and groups and their associated permissions. You can add users, view group memberships, and delete users.

Users and Groups

On the left side of the **Authorizations** tab, users and groups are displayed for the selected server object. From here, you can add users and groups. You can also remove users and groups that you have added through a search. However, removing a user does not delete the user object or retract permissions. This action only removes the user from the view and has no impact on permissions. Use the drop-down menu to filter the view to users, groups, or users and groups together.

Permissions

On the right side of the **Authorizations** tab, the permissions view lists the permissions associated with the selected user or group. All possible permissions for the selected user or group object are displayed. From here, you can grant, deny, or edit inherited permissions. When you select a permission for a user, the active permissions are dimmed in the toolbar. In the following example, user1 has an 'inherited grant' for the SELECT permission. An inherited grant is a permission inherited from a group, or an object in the hierarchy. The **Securable** column reflects the object where the granted permission occurred. When Inherit is the active permission, the Grant and Deny permissions are available and Inherit is dimmed.

Figure 2.2 User Permissions – Inherited Grant

About Inherited Permissions

SAS Federation Server contains an inherent hierarchy of objects, in the following order:

- Server
- Data Service
- DSN and Catalog
- Schema (from catalog)
- Table or View (from schema)
- Column (from table or view)

A data service inherits privileges from the server. The privileges on the data service are inherited by the DSN and catalog. Privileges on the catalog are inherited by the schema, passed to the table or view, and finally, the column. This inheritance hierarchy allows for general security settings on higher level objects, and specific exceptions on the subordinate objects. See the *SAS Federation Server: Administrator's Guide* for a detailed description of permissions and inheritance.

Understanding DSNs and Permissions

CONNECT Permission

A user must have CONNECT permission to establish connection to a DSN. This permission is effective from the user object, inherited through the hierarchy, or acquired through group permissions. For a standard DSN, the CONNECT permission must be on the following (in order of inheritance):

- the DSN
- the parent data service of the DSN
- the federation server object

For a federated DSN, the CONNECT permission must be on the following (in order of inheritance):

- the DSN
- the federation server object

Permissions granted on a federated DSN override any permissions that exist for child DSNs that are contained within the federated DSN. If a user has CONNECT permission on a federated DSN, permissions on any of the child DSNs contained within (standard or

federated) are ignored, even if the user is explicitly denied CONNECT on any of the child DSNs.

Federation Server SQL Authorization Enforcement

Overview

When Federation Server SQL Authorization Enforcement is enabled, the FedSQL driver is engaged, and the SQL dialect is automatically set to FedSQL. With FedSQL an additional layer of object-level security is enabled for the connection and SQL statements are secured before processing them. If Federation Server SQL Authorization Enforcement is disabled, object-level security is bypassed and a user is granted all FedSQL privileges regardless of which privileges the user has been granted or denied. If Federation Server SQL Authorization Enforcement is disabled, an administrator can choose the Native SQL dialect associated with the data source. The SQL dialect for BASE data services is always FedSQL.

Using SQL Authorization Enforcement

You can enable SQL Authorization Enforcement when you create a standard or federated DSN.

- **Enabled:** When SQL Authorization Enforcement is enabled, the FedSQL driver is engaged, and the SQL dialect is automatically set to FedSQL. With FedSQL an additional layer of object-level security is enabled for the connection and SQL statements are secured before processing them.
- **Disabled:** If SQL Authorization Enforcement is disabled, object-level security is bypassed and a user is granted all FedSQL privileges regardless of which privileges the user has been granted or denied. If SQL Authorization Enforcement is disabled, an administrator can choose Native SQL dialect. For example, if you are connected to Oracle, then native dialect is SQL supported by Oracle.

About the Data Source Names Tab

Overview

The **Data Source Names** tab shows the DSNs that are associated with the selected federation server object. This list also includes system-generated DSNs such as ADMIN, BASE, SQL_LOG, and SYSPROC. See the *SAS Federation Server: Administrator's Guide* for more information about the system-generated DSNs.

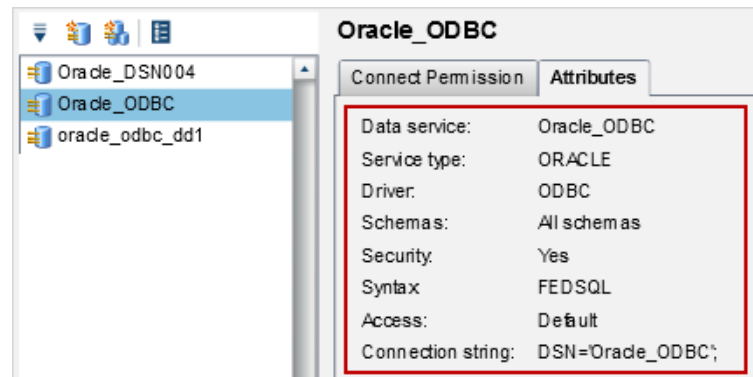
DSNs are accompanied by the **Connect Permission** and **Attributes** tabs where you can view information about a selected DSN. Federated DSNs display an additional **Members** tab that lists the DSNs belonging to the federated DSN.

Connect Permission Tab

The **Connect Permission** tab shows the list of users and groups that have been granted or denied CONNECT permission for the selected DSN. You can also add users and groups in the **Connect Permission** tab. After adding these objects to the list, they must be granted or denied permissions in order to persist beyond the session.

Attributes Tab

The **Attributes** tab displays the name of the data service for the DSN that is selected in the tree. Also listed are the service type, driver, catalogs, schemas, security option, syntax, and connection string.



Table, Column, and Row-Level Security

Overview

Table, column, and row-level security provide an additional layer of security for data when there is a need to control access at a granular level. Security is provided at the table, column, or row level in the database so that users have access to only the data that they require. For additional details related to table, column and row-level security see the *SAS Federation Server: Administrator's Guide*.

Column and Table Privileges

By default, SQL requires a column privilege to override a privilege placed on the table. In other words, a column GRANT overrides a DENY on the table. In the event that column privileges override table privileges, the privileges displayed on the table reflect the user's overall permissions by taking the column privileges into account.

A DENY placed on a table where GRANT is set at the column level results in one of the following conditions:

- If the table is not inheriting privileges from a column because the privilege was directly granted at the table level, the securable column changes from the name of the table (to columns), because the table is now inheriting GRANT permission from the column. Any other columns in the table for which explicit privilege is not granted display an inherited DENY.
- If the table is inheriting GRANT privilege from one or more columns, the explicit DENY does not result in a privilege change. If any columns that explicitly GRANT permission are set to INHERIT, the explicit DENY is reflected on the table.

Performing the inherit action on the table cascades to the column level. GRANT or DENY privileges are revoked for all columns that have explicit GRANT or DENY set as the inherited privilege. The columns inherit the permission level that is in force for the table.

Overview

Data Masking in SAS Federation Server is a series of FedSQL functions that are accessed through the Console using the SYSCAT.DM.MASK function in a SELECT statement. Here is a brief description of each of the data masking functions.

ENCRYPT

Encrypt masks the values in a column by encrypting a single value using symmetric key encryption. Encrypted values cannot be decrypted if a KEY argument is not specified and the ENCRYPT_KEY package configuration option is not set.

```
SYSCAT.DM.mask('ENCRYPT', "value"
/*[,
'ALG', 'AES/FIPS|AES|SAS002|BASE64|SAS004|SAS003|SAS001',
'KEY', 'encrypt_key',
'DETERMINISTIC', YES|TRUE|ON|1|NO|FALSE|OFF|0,
'EXPAND_PREC', YES|TRUE|ON|1|NO|FALSE|OFF|0,
'CASE', 'U|L',
'STRIP', 'BLANK|UNICODESP|UNICODESPACE|ANY|ALL|WS'
]*/ )
```

Note: The encrypt function preserves the data type of the original column if the data type is that of character (for example, char, nchar, varchar). If the column is not a character data type, the output produces a binary data result.

DECRYPT

Decrypt unmask the values in a column by decrypting a previously encrypted value using symmetric key encryption. The DECRYPT rule returns NULL if a KEY argument is not specified and the ENCRYPT_KEY package configuration option is not set.

```
SYSCAT.DM.mask('DECRYPT', "value"
/*[, 'ALG', 'AES/FIPS|AES|SAS002|BASE64|SAS004|SAS003|SAS001',]*/ )
```

HASH

HASH masks the values in a column by hashing a single value into a fixed-length hash digest or HMAC string. HASH is not reversible.

```
SYSCAT.DM.mask('HASH', "value"
/*[, 'ALG', 'MD5|SHA256',
'CASE', 'U|L',
'KEY', 'encrypt_key']*/ )
```

TRANC

TRANC masks the values in a column by transliterating characters from an input string to characters in an output string. Ensure that the mapped result is “lossy” (many instances of mapping multiple input character values to a single output character value) to prevent inference of the original value.

```
SYSCAT.DM.mask('TRANC',
/*Expression*/
)
```

RANDOM

RANDOM masks the values in a numeric column by replacing them uniformly distributed pseudo-random numbers. RANDOM is not reversible.

```
SYSCAT.DM.mask('RANDOM',
```

```
/*Expression*/
)
```

RANDATE

RANDATE masks the values in a date column by replacing them with pseudo-random date values.

```
SYSCAT.DM.mask( 'RANDATE',
/*Expression*/
)
```

RANSTR

RANSTR masks the values in a column by replacing the values with random strings. Strings are generated by an algorithm that uses characters from the source string in the generation process, adding padding characters if necessary. Padding is placed to the left of the string unless RIGHT is specified.

```
SYSCAT.DM.mask( 'RANSTR',
/*Expression*/
)
```

RANDIG

RANDIG masks the numeric values in a column by replacing digits with strings of random digits. Strings are generated by an algorithm that uses digits derived from the base number system of the source value, adding padding digits if necessary. Padding is always to the left of digits.

```
SYSCAT.DM.mask( 'RANDIG',
/*Expression*/
)
```

Part 2

SAS Federation Server Manager Interface

Chapter 3

Accessing SAS Federation Server Manager 15

Chapter 4

Navigating SAS Federation Server Manager 19

Chapter 3

Accessing SAS Federation Server Manager

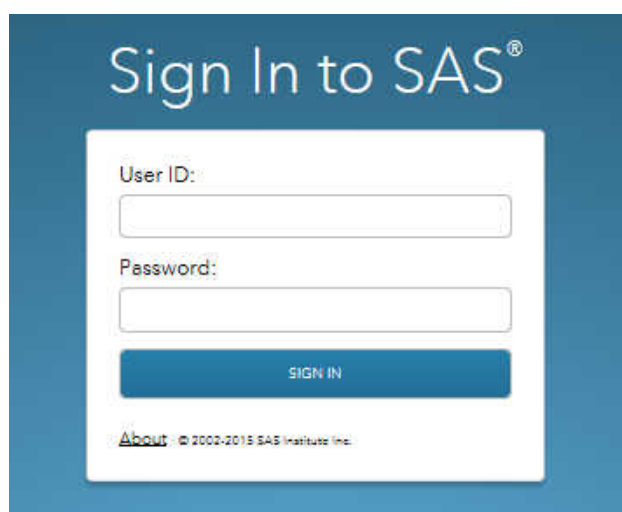
Log On to SAS Management Console	15
SAS Federation Server Manager Home Page	16
Specifying Preferences	17
User Preferences	17
Server Preferences	17

Log On to SAS Management Console

SAS Federation Server Manager is accessed through the standard logon window for SAS applications. Logging on from this window opens SAS Data Management Console from which you can launch SAS Federation Server Manager. To log on to SAS Data Management Console:

1. Click the URL that is supplied in Instructions.html, or paste it into the address field of your browser to display the SAS logon window:

Figure 3.1 Logon Window for SAS Data Management Console



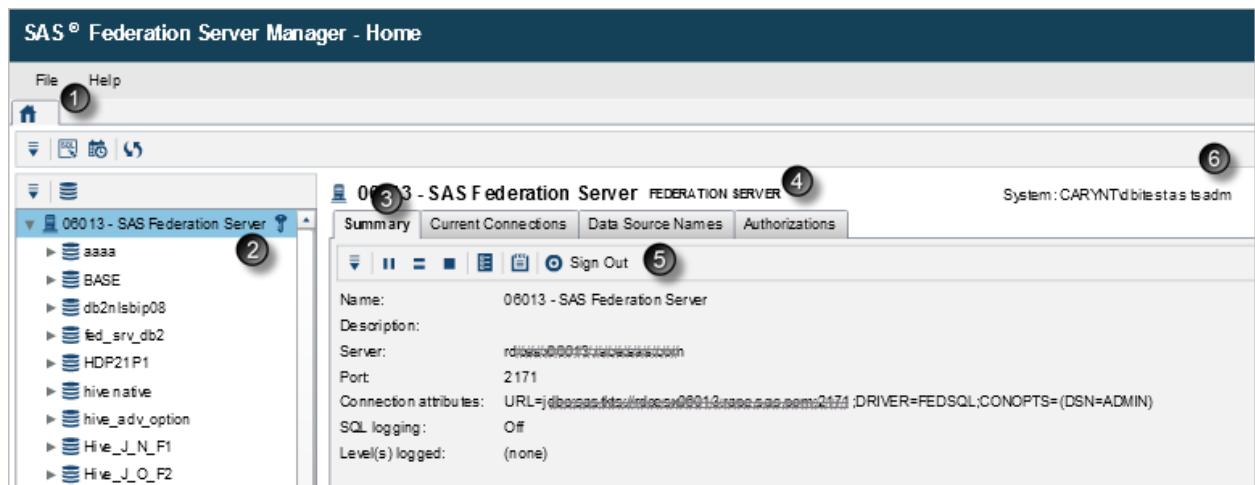
2. In the **User ID** field, enter your user ID.
3. In the **Password** field, enter the password for your user ID.

Note: If you log on to SAS Federation Server Manager in one browser tab, and then log on to SAS Federation Server Manager in another browser tab, the same credentials are used automatically for subsequent authentication attempts.

SAS Federation Server Manager Home Page

After logging in to SAS Federation Server Manager, you are presented with the Home page. The home page displays a home tab and a list of registered federation servers in a tree configuration.

Figure 3.2 SAS Federation Server Manager – Home



- 1 Home tab: The home tab is always displayed first and cannot be closed. The home tab displays all of the federation servers and associated objects in the tree.
- 2 The tree: The SAS Federation Server Manager contains all registered servers and their associated objects. You can expand or collapse objects within the tree. Federation server objects expand in the following order: **federation server** ⇒ **data service** ⇒ **catalog** ⇒ **schema** ⇒ **table/view/cache**.
- 3 **Summary** tab: This tab is the first of the information tabs for the selected federation server. The tabs that follow are visible only when logged on to a federation server object. However, the **Summary** tab is always visible. The detail area displays summary information for the selected federation server object, such as server name, address, port, and connection attributes.
- 4 Selection indicator: The selection indicator responds to selection in the tree and shows the type of object that is selected. A **FEDERATION SERVER** is shown in this example. Other possible indicators are **DATA SERVICE**, **CATALOG**, **SCHEMA**, and **TABLE**.
- 5 Action menu and toolbar: This is the location of the action menu and toolbar items for the **Summary** tab. Each tab contains an action menu and toolbar that is related to activity for the selected object.
- 6 Identification: This displays information for the user that is currently logged on to SAS Federation Server Manager. If the user is a system user, the user ID is prefixed with a server role that appears as: **SYSTEM:domain\login**. If a user has

ADMINISTER privilege but is not a system user, the user ID is prefixed with Administrator.

Specifying Preferences

User Preferences

Global Preferences

Global preferences are applied to all SAS web applications that use Adobe Flash Player. These preferences can be set by each user. User locale specifies the geographic region whose language and conventions are used in the applications. This setting might also apply to some SAS web applications that are not displayed with the Adobe Flash Player. Theme specifies the collection of colors, graphics, and fonts that appear in the applications.

Changing the Appearance of the User Interface

You can change the appearance of SAS Federation Manager by modifying the application's theme. To change the theme:

1. In the upper left corner, select **File** ⇒ **Preferences** to open the Preferences dialog box.
2. Select **Global Preferences** and using the drop-down menu for **Theme**, choose another theme.
3. Click **OK** to save your changes.

You can customize certain aspects of the chosen theme by selecting these options:

- **Invert application colors** inverts all of the colors in the application window, including both text and graphical elements. You can also temporarily invert or revert the colors for an individual application session by pressing Ctrl+~.
- **Override settings for focus indicator** enables you to set the attributes of the outline that indicates which user interface component is active. To the right of the controls is a sample area that shows how the changes affect your application.

Server Preferences

SAS Federation Server Manager Preferences

Use the Preferences dialog box to configure server behavior for SAS Federation Server Manager.

- **Maximum rows returned:** The default is 50.
- **SQL statement delimiter characters:** The default is semicolon (;).
- **Refresh connection information:** The default is Never (manual refresh only).

Changing the SQL Statement Character Delimiter

To change the character delimiter for SQL statements:

1. Select **File** ⇒ **Preferences**.
2. In the Preferences dialog box, select **SAS Federation Server**.
3. Change the character in the **SQL Statement delimiter characters** box.
4. Click **OK** when you are finished.

Refresh Server Connection Information

You can set connection information for the federation servers by selecting one of the options below. The setting takes effect at a subsequent logon event for each server.

- Select **Periodically** and enter a value in minutes. For example, enter a value of 60, so the server connections are refreshed every hour.
- Select **Never** so that connections for each federation server are refreshed manually.

Chapter 4

Navigating SAS Federation Server Manager

Layout of the Interface	19
Overview	19
Federation Server Objects and Views	19
Overview	19
Federation Server	20
Tables and Views	21
Columns	23
Action Menus	24

Layout of the Interface


Overview

Navigation in SAS Federation Server Manager consists of a series of tabs and action menus. Federation servers and associated objects are arranged in a tree view in the left panel of the user interface. The tree is populated with federation servers and their associated objects including data services, catalogs, schemas, and tables or views, including cached views.

Federation Server Objects and Views

Overview


Federation servers and associated objects are arranged in a view in the left panel of the user interface. While navigating and configuring contents in the tree, an Action menu is displayed in the upper left corner that contains options for the specific object that you are configuring. For example, when selecting a federation server object in the tree, the

Action menu  contains functions specific to a federation server object. The following objects can appear in the tree:

- Federation server
- Data service
- Catalog

- Schema
- Native Table or Native View
- FedSQL View or Cached FedSQL View
- DS2 Procedure/Package/Method

Federation Server


When a user is logged on to the federation server, an **Online** indicator  appears next to the federation server object in the tree and in the selection indicator.

Each federation server object displays information tabs that summarize configuration and connection properties for the selected server object. The **Summary** tab is displayed at all times. The remaining tabs are displayed after logging on to a federation server. Here are the tabs that are displayed when you are logged in to SAS Federation Server:

Summary

The **Summary** tab reflects the server name, description, server DNS name, port, and connection attributes. The SQL logging options for the selected server are also shown on this tab.

Current Connections

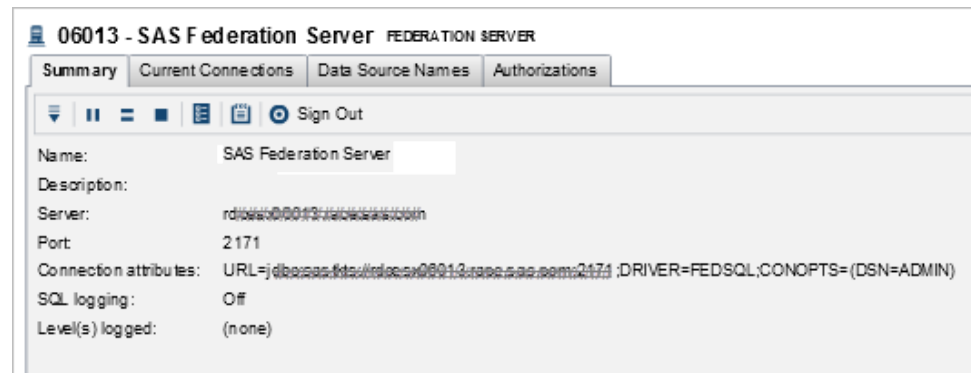
The **Connections** tab lists the current connections for the selected federation server object and also displays current sessions if requested. You must be logged in to a server to see all server connections and sessions. You can toggle between **Show Connections** or **Show Sessions** by using the drop-down menu. Each time you click the **Connections** tab, the displayed content is refreshed. You can also use **Refresh**  to refresh connections and sessions.

Data Source Names

The **Data Source Names** tab shows the DSNs that exist for the selected federation server. This list also includes system-generated DSNs such as ADMIN, BASE, and SQL_LOG. DSNs display a **Connect Permission** and **Attributes** tab. Federated DSNs also display a **Members** tab that lists the DSNs that belong to the federated DSN.

Authorizations


Each federation server object contains an **Authorizations** tab containing the users and groups that are specific to the server. By default, users are not granted any privileges as they are created. The SYSTEM user or federation server administrator must grant permissions and privileges so that users can perform actions and gain access to data. See [“About Authorizations”](#) for additional information.

Figure 4.1 SAS Federation Server Manager: Navigation Tabs

Tables and Views


Native Table

When a table or view object is selected, the name and type of the object is displayed in the right panel, accompanied by a series of tabs that explain the properties of the selected object.

 This is a native table with from which you can create a new FedSQL view from Table, view data, and display authorizations, including row-level authorizations. You can also set row-level security. Other table types can be associated with this icon (for example, a Teradata NOPI table is displayed as a native table)

Note: This icon also depicts a SYSTEM table such as the Memory table used for memory data store (MDS).

FedSQL View

 This is a FedSQL view from which you can cache data. It displays the following tabs:


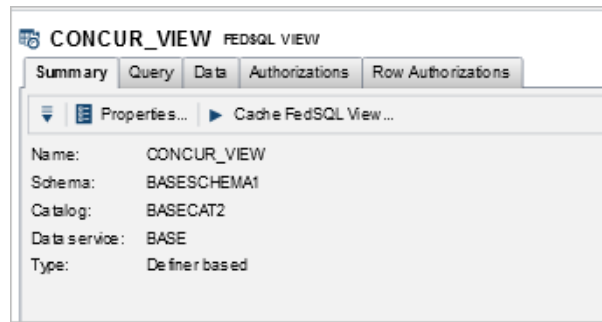
- **Summary:** Shows the name, schema, catalog, and data service associated with the FedSQL view. You can **Cache FedSQL View** from the view.
- **Query:** Displays the query used to create the view.
- **Data:** Displays the data used for the view when you click **View Data**.
- **Authorizations:** Displays user and group authorizations associated with the table.
- **Row Authorizations:** If row-level security is in place, displays existing row filters and the associated user and group authorizations. You can also set row-level security on this tab using **New Filter** .

Figure 4.2 FedSQL View

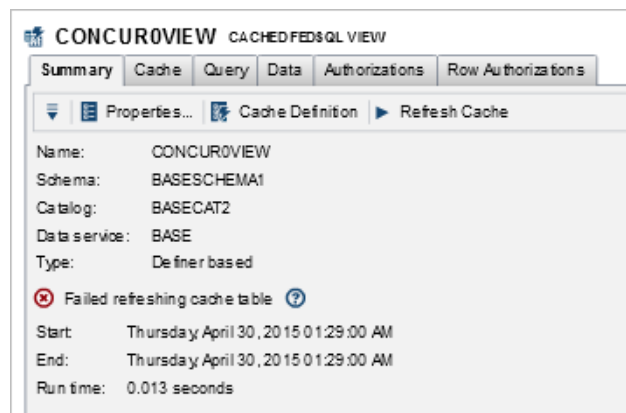


Cached View

A cached view is data cached from a FedSQL view. It displays the following tabs:

- **Summary:** Shows the name, schema, catalog, and data service associated with the cached FedSQL view. You can **Update Cache Table** from the view.
- **Cache:** Shows the start, end, and run time for the cache. From here you can view the **cache definition** or **refresh cache**. This tab also displays the location of the cache table.
- **Query:** Displays the query used to create the view.
- **Data:** Displays the data used for the view when you click **View Data**.
- **Authorizations:** Displays user and group authorizations associated with the table.
- **Row Authorizations:** If row-level security is in place, displays existing row filters and the associated user and group authorizations. You can also set row-level security on this tab using the **New Filter** icon .

Figure 4.3 Cached FedSQL View

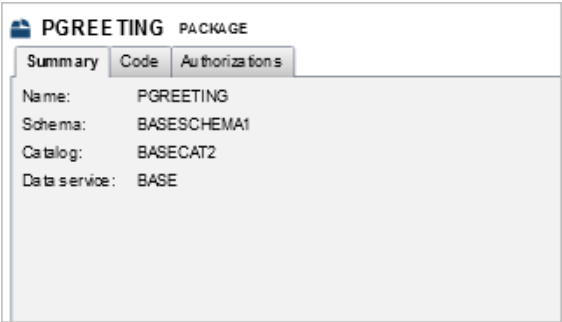


DS2 Package

A DS2 package shows the name, schema, catalog, and data service associated with the DS2 package. It displays the following tabs:

- **Summary**
- **Code**
- **Authorizations**

Figure 4.4 DS2 Package

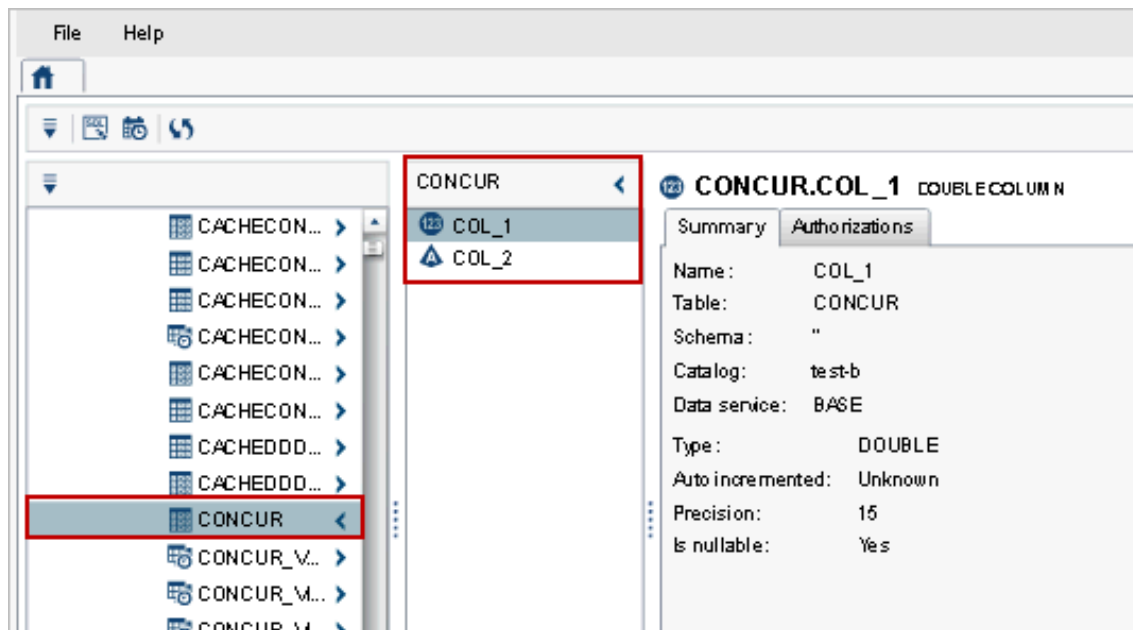


Columns

You can expand table objects to view the columns associated with the table using the arrow icon. When the table is expanded, a column pane opens to the right of the table showing the columns associated with the selected table. When you select a column, a **Summary** tab and **Authorizations** tab are displayed in the right pane. Only one column can be viewed at a time. Each COLUMN_TYPE is associated with an icon. For DS2 Methods, there are two columns of icons, one for COLUMN_TYPE and one for DATA_TYPE. Here is a list of column types with their associated icons:

Table 4.1 SAS Federation Server Manager Column Types

Icon	Column
	Character Type
	Numeric Type
	Date
	Time
	Boolean Data
	Date/Time

Figure 4.5 SAS Federation Server Manager: Columns View

Action Menus


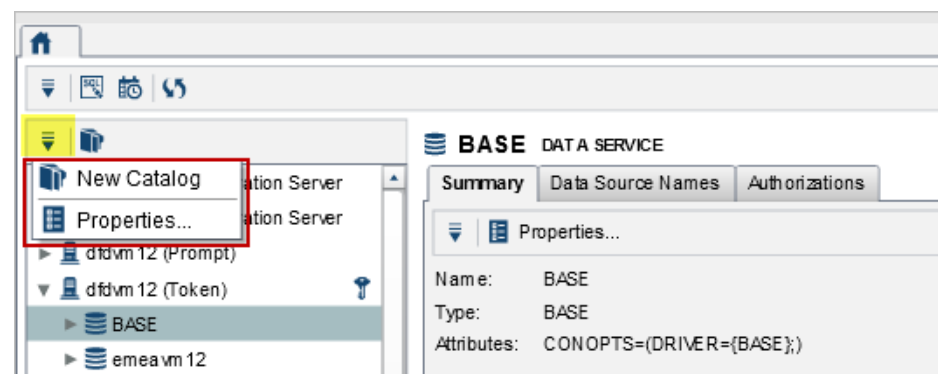
While navigating and configuring contents in the tree, an Action menu  is displayed in the upper left corner that contains options for the specific object that you have selected. As you select objects in the tree, the items displayed on the associated action menu are based on the functionality of the selected object. For example, selecting a data service in the tree, displays **New Catalog** and **Properties** in the **Action** menu.

Figure 4.6 Action Menu Associated with a Data Service

Part 3

Configuring SAS Federation Server Manager

<i>Chapter 5</i>	
Post-Installation Configuration	27
<i>Chapter 6</i>	
Configuring SAS Federation Server	35
<i>Chapter 7</i>	
Configuring Access to Data Sources	45

Chapter 5

Post-Installation Configuration

Post-Installation Configuration	27
Overview	27
SAS Metadata Server	27
Shared Login Accounts	29

Post-Installation Configuration

Overview

After you install SAS Federation Server, you must perform additional configuration steps before you can use SAS Federation Server Manager. At the end of the installation, the SAS Deployment Wizard produces an HTML document named `Instructions.html`. If your server tier and middle tier are hosted on separate machines, there is an `Instructions.html` file for each machine. The `Instructions.html` file is located in **SAS \Config\Lev#\Documents**. Here is an outline of tasks that require attention:

1. Verify that all installation and configuration steps in the `Instructions.html` file have been completed.
2. User security: Create users, groups, and roles.
 - Create a Shared login group.
 - Verify that the trusted user was created at installation.
3. (Optional) Specify an encryption level for SAS Federation Server.
4. If you are upgrading from a previous release of SAS Federation Server Manager, move scheduled jobs to the new version.

SAS Metadata Server

User Requirements and Roles

To access SAS Federation Server Manager, users might require group membership that includes assignment of specific roles.

- A user object requires membership to the SASUSERS group, with the **Federation Server Manager: Operation** role.

- An administrator requires membership to the **SAS Federation Server Administrators** role. The Federation Server Manager: Operation role is assigned by default.
- The SAS Federation Server system user that is created at installation is sasfedadm, and is a member of the **SAS Federation Server Administrators** group.
- The SAS Trusted User that is created at installation is sastrust. This account replaces the trusted user formerly created in DataFlux Authentication Server.
- The SAS Federation Server object is created on SAS Metadata Server at installation. The federation server object is no longer defined in SAS Federation Server Manager.

SAS Federation Server Administrators

A user becomes an administrator when his or her account is added to the SAS Federation Server Administrators group in SAS Metadata Server. There are two ways to make an account administrator:

1. Adding the user account to the **SAS Federation Server Administrators** group. By default, only the Metadata Administrator can perform this action.
2. Issue GRANT ADMINISTER DDL on the SAS Federation Server. Only the system user has the authority to grant or revoke the ADMINISTER privilege through the use of administration DDL. The ADMINISTER permission is available on the server object only.

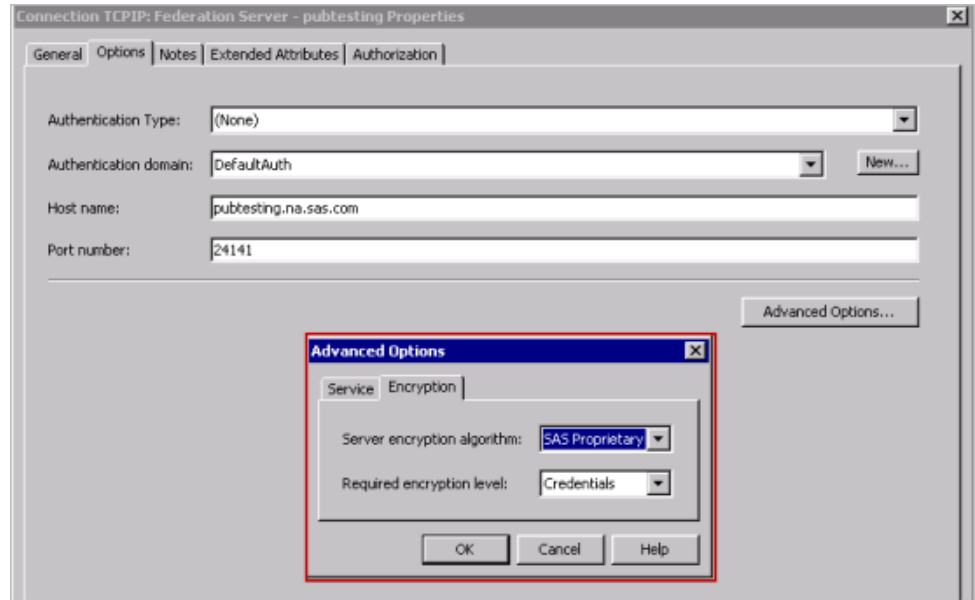
This action grants the ADMINISTER privilege to the user object. Only the SAS Federation Server System user (sasfedadm), or SAS Metadata Administrator (sasadm) can grant the ADMINISTER privilege.

Note: See the *SAS Federation Server: Administrator's Guide* for information about the ADMINISTER privilege and DDL statements.

Specify Server Encryption Level

Use SAS Management Console to specify or change the encryption level for a particular SAS Federation Server:

1. In SAS Management Console, locate your federation server object by expanding **Environment Manager** ⇒ **Server Manager** ⇒ **Federation Server - hostname - logical server**.
2. Expand the logical server entry and select the server definition for which you want to change encryption. The **Connections** tab displays the current connections defined for the selected server.
3. On the **Connections** tab, select a connection and right-click. Select **Properties** from the drop-down menu.
4. Select the **Options** tab, and select **Advanced Options**.
5. Select the **Encryption** tab, and select an option from the **Server encryption algorithm** list menu.

Figure 5.1 Specify Server Encryption in SAS Management Console

6. Click **OK** to exit the Advanced Options dialog box, and click **OK** to close connection properties.
7. Restart SAS Federation Server to update the server encryption algorithm.

Note: The encryption algorithm chosen for SAS Federation Server should always match the encryption algorithm chosen for SAS Metadata Server.

Shared Login Accounts

About Shared Logins

Shared logins consist of a shared login key, the login account, and the users or groups who are members of the (shared) login account. The SAS Federation Server administrator creates and controls the shared logins for SAS Federation Server.

When using a shared login to authenticate to a data source, users do not need to know the credentials of the shared login. The shared login retrieves credentials for the user who is logged on and provides the credentials to SAS Federation Server. In turn, the server connects the user to the database through the appropriate data service or data source name (DSN).

Outline of Shared Login Tasks

The implementation of shared logins has changed in SAS Federation Server 4.2. Here is a summary of the tasks:

- Create a shared login key for SAS Federation Server using administrative DDL or in SAS Federation Server Manager in the properties of a federation server object. The shared login key is case sensitive. The key that is defined in SAS Federation Server must match the key that is part of the shared login definition in the SAS Metadata Server.
- Create a shared login account (group) in SAS Metadata Server using SAS Management Console. The shared login account includes the login to be shared and its domain.

- Add consumers of the shared login as members of the shared login account. Consumers are SAS Federation Server user accounts or groups. You should never use the actual shared login group as a consumer group in a DSN.
- Create a data service for the applicable data source. In the DSN, specify that the data will be accessed with a shared login.

About the Authentication Domain

When establishing connection to the SAS Federation Server, the following logic is used to find the proper login:

- If connecting with a DSN configured to use a personal or group login, SAS Federation Server uses the authentication domain associated with the data service to look up a login for the user.
- If connecting with a DSN configured to use a shared login, SAS Federation Server uses the authentication domain associated with the data service and appends the domain with a suffix of “@<*shared login key*>” to look up a login for the user.

Creating a Shared Login

The tasks presented in the following topics outline the basic steps to create a shared login for SAS Federation Server:

1. Set a shared login key (SAS Federation Server Manager).
2. Create the shared login account (SAS Management Console).
3. Create a data service and DSN for the data source (SAS Federation Server Manager).

Set a Shared Login Key

The shared login key is used when configuring an authentication domain in SAS Metadata Server. The shared login key is case sensitive. The following steps show how to set a shared login key with SAS Federation Server Manager:

1. Locate the federation server object in the tree, and log on to the server if prompted to do so.
2. Select **Action Menu** ⇒ **Properties** in the upper left corner.
3. Click the **Security** tab and enter the shared login key.
4. Click **OK** to exit the properties dialog box.

TIP You can also use administration DDL to set a shared login key: **ALTER SERVER {OPTIONS (SHAREDLOGINKEY *name-of-key*) }**

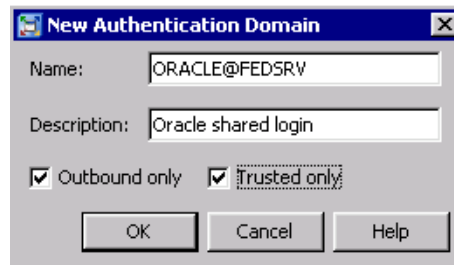
Create a Shared Login Account

The shared login account is actually a group that serves as the shared login account, so the name of the group should reflect that. See step 4a below.

1. Log on to SAS Management Console
2. On the **Plug-ins** tab, select **User Manager**.
3. Right-click and select **New** ⇒ **Group**.
4. In the New Group Properties dialog box:
 - a. On the **General** tab, enter a name for the shared login (for example, Oracle Shared Login for FedServer).
 - b. On the **Members** tab, add users and groups who will use the shared login.

- c. On the **Accounts** tab, add the account and password.
- d. Select **New** for Authentication Domain.
 - Enter an Authentication Domain name using this format:
`<data_service_domain>@<shared_login_key>`
 For example, if the domain for the data service is OracleAuth and the shared login key is **FSKey1**, then the shared login domain must be **OracleAuth@FSKey1**. The shared login key is case sensitive and must match the shared login key that was set in SAS Federation Server Manager.
 - Select **Outbound only** and **Trusted only** for the domain.

Figure 5.2 New Authentication Domain Dialog Box



Outbound only: An outbound domain is used only to provide SAS applications with access to external resources, such as a third-party vendor database.

Trusted only: The trusted user is a privileged service identity that can act on behalf of all other users. A login in a trusted domain can be accessed only by a trusted user.

5. On the **Authorizations** tab, ensure that the SAS Administrators group has these permissions:
 - ManageMemberMetadata
 - ManageCredentialsMetadata
 - ReadMetadata
 - WriteMetadata

Create a Data Service and DSN

When you create a data service, a DSN with the same name is automatically created for you. Use SAS Federation Server Manager to perform the following task.


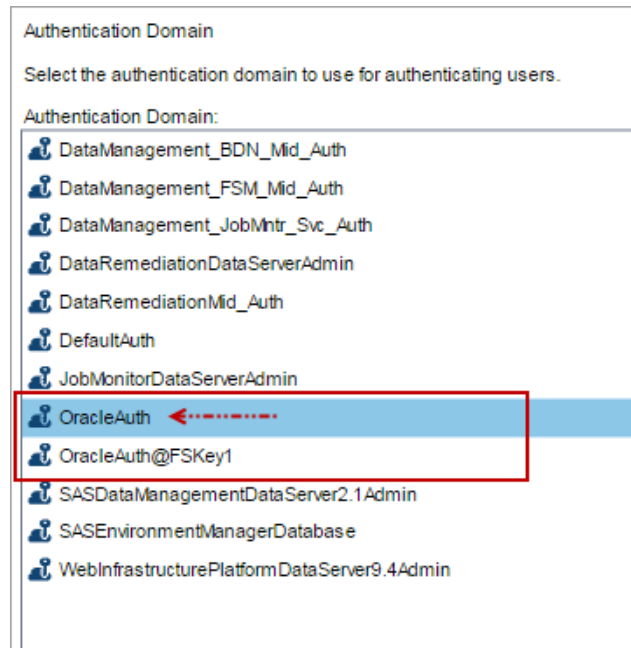
1. Select a federation server object in the tree, and log on to the server if you are prompted
2. Select **Action** ⇒ **New Data Service**, or click the New Data Service icon  on the toolbar.
3. In the Identification dialog box, enter the name of the data service and click **Next** to continue.
4. In the Authentication Domain dialog box, select an Authentication Domain from the list of available domains and click **Next** to continue.

Figure 5.3 Defining the Data Service Authentication Domain**CAUTION:**

Select a stand-alone data source domain. Do not select the domain with the shared login key that was created in SAS Metadata Server. When the DSN is set to use a shared login, SAS Federation Server appends the selected domain with @ and the shared login key and verifies that **data source@<shared login key>** exists in SAS Metadata as a valid authentication domain that includes user and password account information.

5. In the Summary dialog box, verify the settings and click **Finish**.

Set the Shared Login Indicator in the DSN


1. Select the **Data Source Names** tab affiliated with the Oracle data service that you just created. You should see a DSN that is named for the new data service.
2. Select the Action menu , select **Properties**, and click **Next** until you reach the Access dialog box.
3. In the **Specify the type of login required to access this DSN** field, select the **Shared login** check box.

Figure 5.4 Shared Login Specification for DSN Access

The screenshot shows a configuration window titled "Access". It contains the following elements:

- A heading "Access" followed by the instruction "Specify the type of login required to access this DSN." with a help icon.
- Two radio button options: "Personal login" (unchecked) and "Shared login" (checked).
- A label "Consumer group:" followed by a drop-down menu showing "Data Management Business Users".
- A label "Access Order:" followed by two radio button options: "Try personal login first" (selected) and "Try shared login first" (unselected).

4. From the **Consumer group** drop-down list, select a group if necessary.

Note: The Consumer group identifies which shared login should be used if a conflict occurs for a user. The Consumer group should be a group that is directly or indirectly a member of the shared login.

5. Click **Next**, **Next**, **Next**, and **Finish**.

Chapter 6

Configuring SAS Federation Server

Configuring a Federation Server	35
Prerequisite	35
Federation Server Properties	35
Define Security Objects	37
Enable Logging	37
Enable Connection Pooling	37
Purge Cache Tables	38
User and Group Authorizations	39
Overview	39
Adding Users and Groups	39
Removing Users and Groups	39
Granting Permissions for Federation Server and Associated Objects	41
Overview	41
Granting Permissions	41

Configuring a Federation Server

Prerequisite

Federation servers must be added in SAS Metadata Server before they are visible in SAS Federation Server Manager. See “[Post-Installation Configuration](#)” for a list of configuration tasks for SAS Metadata Server.

Federation Server Properties

Overview


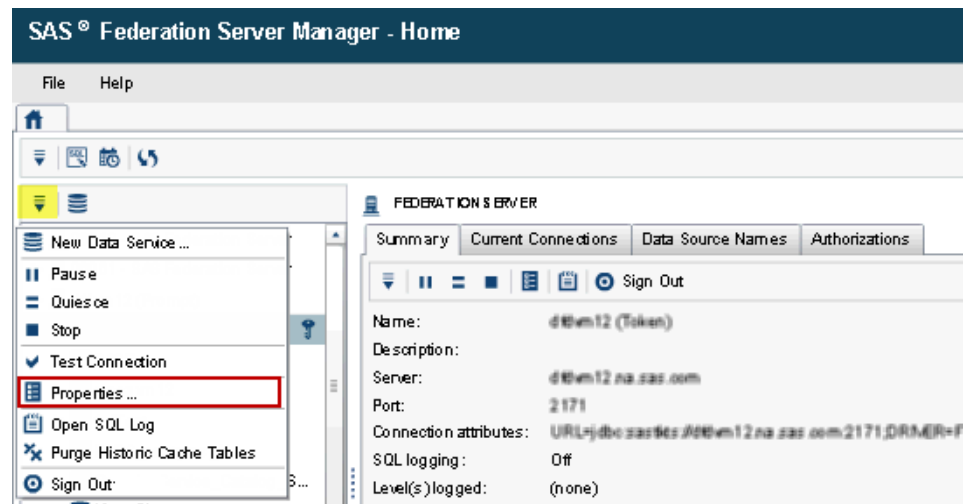
You can access the properties for a federation server object using the action menu  on the toolbar.

Figure 6.1 Action Menu – Federation Server Object

The federation server properties dialog box contains the following tabs:

General

Displays the name and description of the selected federation server. This dialog box also contains the server's DNS name and port number that define the SAS Federation Server in the network. The default port number is 24141. These fields are populated with configuration information from SAS Metadata Server. If you need to edit any of these properties, use SAS Management Console to update the information in SAS Metadata Server. You can test the server's connection using **Test Connection**.

Security

Defines the Shared Login Key and Data Masking encryption value.

Logging

Enables SQL Logging and specific logging options.

Schedule

Shows the schedule to purge unused historic cache tables. You can configure cache settings here.

Configuration

This tab is read only. It contains configuration information for SAS Federation Server such as DNS name, port number, server state, loggers, and federation server cache tables.

Connection Pooling

Enables connection pooling where you can set additional connection options.

Other


Contains configuration settings for Trace File Path, Shutdown Timeout, FedSQL settings, and Driver transcoding failure behavior. You can also set advanced options for the server using key/value pairs.

Note: Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Define Security Objects

Specify a Shared Login Key


To specify a shared login key if one has not already been specified:

1. Select a federation server object in the tree.
2. Open the  in the upper left corner and select **Properties**.
3. Click the **Security** tab on the Federation Server Properties dialog box.
4. Enter the **Shared Login Key** and click **OK**.

Note: The shared login key is case sensitive.


Data Masking Encryption Key

You can specify an encryption key for data masking, also referred to as RANDOM SEED. If the encryption value is set using the ALTER SERVER DDL statement, the key is reflected here. To specify a data masking encryption key in federation server properties:

1. Select a federation server object in the tree.
2. Open the  in the upper left corner and select **Properties**.
3. Click the **Security** tab on the Federation Server Properties dialog box.
4. Enter the **Data Masking Encryption** key and click **OK**.

Enable Logging


When you enable Logging, any parameters set for logging during the session revert to the default configuration settings upon restart of the server. The information captured here does not affect regular server logging, which is set within the `dfs_log.xml` configuration file on SAS Federation Server. To enable logging for session events:

1. Select a federation server object in the tree.
2. Open the  in the upper left corner and select **Properties**.
3. At the Federation Server Properties dialog box, select the **Log** tab
4. Click to select **On. Log SESSION** and select the events that you want to log and click **OK**.

See “[SQL Logging Transactions](#)” for a brief description of the behavior of each of these transactions.

Enable Connection Pooling

Connection pooling is a reserve of database connections that are maintained so that the connections can be reused as future requests to the database are required. To enable connection pooling:

1. Select a federation server object in the tree, and select  in the upper left corner.
2. Select **Properties** from the drop-down menu.

3. Click the **Connection Pooling** tab on the Federation Server Properties dialog box.
4. Select **Enable connection pooling**.
5. Accept the **Maximum Used Connections** default of 50, or select **Specify** and enter the number of maximum connections allowed.
6. To set a time-out, select **After** and specify the *number of minutes* for time-out of unused connections.
7. Click **OK** when you are finished configuring Connection Pooling.

Purge Cache Tables

To schedule cache purge on the **Schedule** tab in federation server properties:


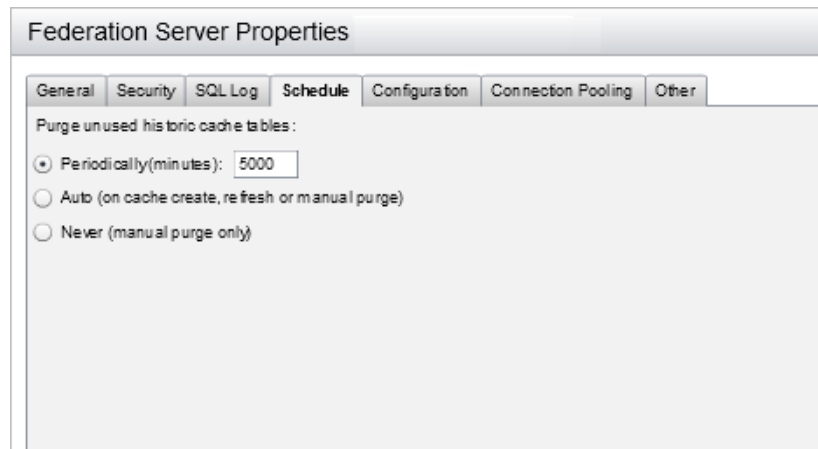
1. Select a federation server in the tree and click  in the upper left corner.
2. Select **Properties** from the Action Menu.
3. Select the **Schedule** tab in Federation Server Properties.
4. Select an option for purging unused cache tables:
 - Periodically (default): Enter a timeframe, in minutes.
 - Automatic: Old cache tables are removed after a CREATE CACHE, REFRESH CACHE, or PURGE CACHE command is issued.
 - Never: Manual purge only.

Figure 6.2 Federation Server Properties – Schedule Tab



5. Click **OK** when you are finished.

Note: You can also issue a **Purge Historic Cache** command using the **Action** menu of the selected federation server.

User and Group Authorizations

Overview


By default, users are not granted any permissions. The SAS Federation Server administrator must grant privileges so that users can perform actions and gain access to data. Group permissions are granted and denied in the same manner as individual users. Users who are members of a group inherit the permissions from the group unless explicitly denied in the individual user account.

The basic permissions that users need are SELECT and CONNECT. When users log on to SAS Federation Server Manager, they can see only the permissions that have been assigned to them. All other permissions reflect a status of DENY.

Adding Users and Groups

Add Users to a Server Object

To add users or groups to a federation server object:

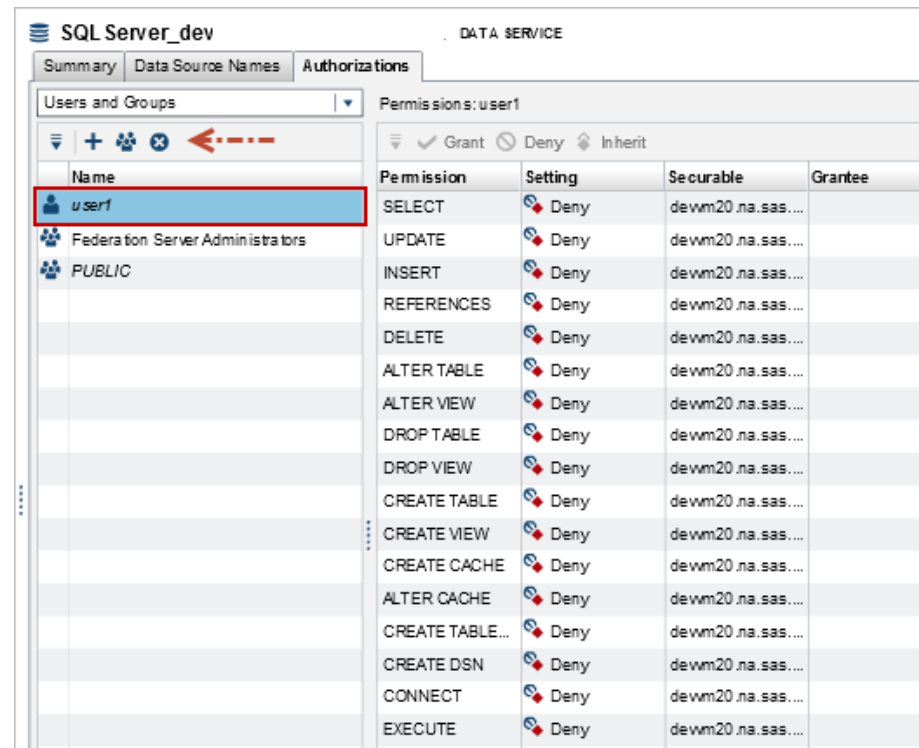
1. Select a federation server object in the tree.
2. Click the **Authorizations** tab in the right pane.
3. Select the **Add Users and Groups** icon .
4. In the Add Users and Groups dialog box, enter a user or group name in the search box and click **Search**, or scroll down and select a user object.
5. Select a user or group and click **Add**. The selected object is added to the Users and Groups list.
6. Click **Close** to return to Authorizations.

Removing Users and Groups

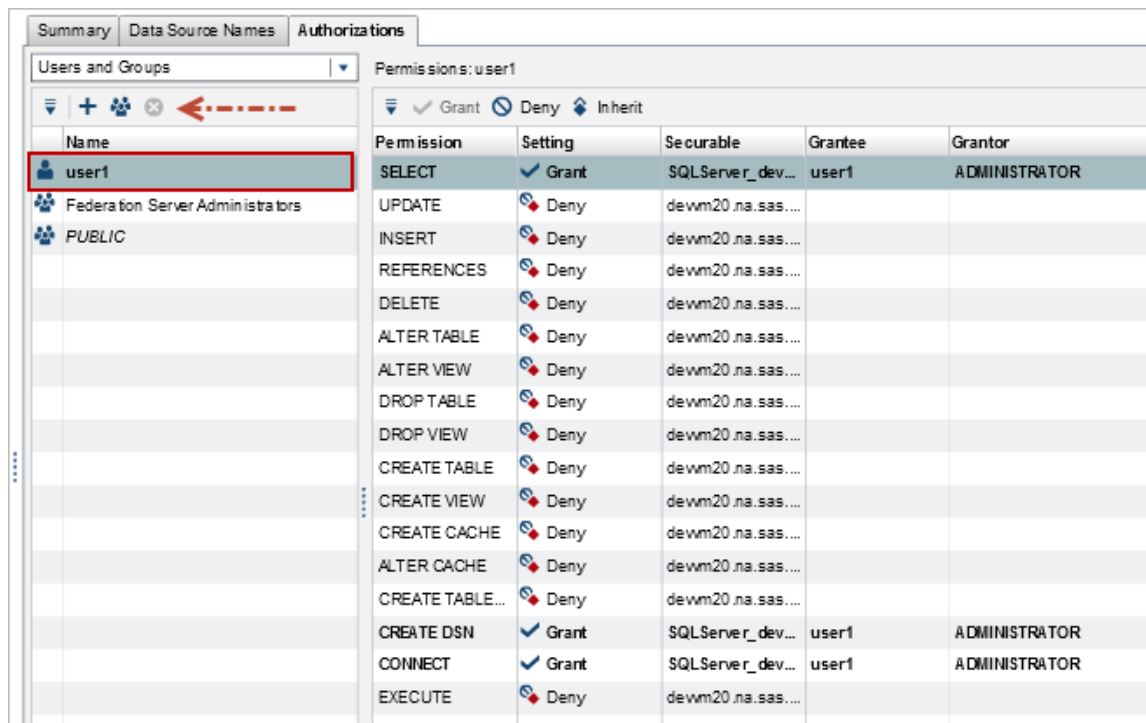
You cannot delete a user or group in SAS Federation Server Manager. These objects are managed through SAS Metadata Server. However, you can remove a user or group from a federation server object if the following conditions are met:

- The object was added through the add or search feature in SAS Federation Server Manager.
- Permissions for the object have not changed.

If you add a user and grant permissions, those permissions must be reversed, or set back to original status, before you can remove the user from the federation server object. In the following example, **user1** was added to the SQL Server data service. Permissions are set to 'inherited deny' and have not been altered. Notice that the delete button is active:

Figure 6.3 Adding a User on the Authorizations Tab

Once you change permissions for the user object, the delete button is dimmed:

Figure 6.4 Adding a User and Altering Permissions

To remove user1, you must reverse the altered permissions to inherited deny. Once the permissions are reversed, the delete button becomes active and you can remove the user from the **Authorizations** tab.

Granting Permissions for Federation Server and Associated Objects

Overview


You can configure permissions for SAS Federation Server and associated objects such as data services, DSNs, and catalogs. Permissions set at the server object level are inherited throughout the object hierarchy as follows:

- Schema inherits from catalog.
- Catalog inherits from data service.
- DSN inherits from data service.
- Data service inherits from federation server.

Granting Permissions

Federation Server

To grant permissions for a federation server object:

1. Select a federation server in the tree and select the associated **Authorizations** tab.
2. Select .
3. Enter a user or group name in the search box and click **Search**.


Note: The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.

4. Select the user (or users) and click **Add**. The user is added to the user list for the object.
5. Click **Close** to return to the **Authorizations** tab.
6. Select the user and set permissions by selecting one or more items in the **Permission** list.
7. Select **Grant**, **Deny**, or **Inherit** to assign permissions.

Note: Permissions that are currently assigned to a user object are dimmed on the **Permissions** toolbar and cannot be selected.

Data Service

Select a **federation server** ⇔ **data service** and use the following task to grant permissions:

1. Click the **Authorizations** tab associated with the selected object.
2. Under **Identities**, select .

3. Enter a user or group name in the search box and click **Search**. To return a list of available user names and groups, leave the search box empty and click **Search**.

Note: The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.

4. Select the user (or users) and click **Add**. The user object is added to the **Identities** list.
5. Select the user from the Identities list, and set permissions by selecting one or more items in the **Permission** list.

Note: Any permissions that are currently assigned to a user object are dimmed on the **Permissions** toolbar and cannot be selected.


6. Select **Grant**, **Deny**, or **Inherit** to assign permissions.

DSN

See “[Establishing DSN Permissions](#)” to configure authorizations for a DSN.


Catalog

Select a **federation server** ⇒ **data service** ⇒ **catalog** and use the following task to grant permissions:

1. Click the **Authorizations** tab associated with the selected object.
 2. Under **Identities**, select .
 3. Enter a user or group name in the search box and click **Search**. To return a list of available user names and groups, leave the search box empty and click **Search**.
- Note:* The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.
4. Select the user (or users) and click **Add**. The user object is added to the **Identities** list.
 5. Select the user from the Identities list, and set permissions by selecting one or more items in the **Permission** list.
- Note:* Any permissions that are currently assigned to a user object are dimmed on the **Permissions** toolbar and cannot be selected.
6. Select **Grant**, **Deny**, or **Inherit** to assign permissions.

Schema

Select a **federation server** ⇒ **data service** ⇒ **catalog** ⇒ **schema** and use the following task to grant permissions:

1. Click the **Authorizations** tab associated with the selected object.
2. Under **Identities**, select .
3. Enter a user or group name in the search box and click **Search**. To return a list of available user names and groups, leave the search box empty and click **Search**.

Note: The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.


4. Select the user (or users) and click **Add**. The user object is added to the **Identities** list.
5. Select the user from the Identities list, and set permissions by selecting one or more items in the **Permission** list.

Note: Any permissions that are currently assigned to a user object are dimmed on the **Permissions** toolbar and cannot be selected.

6. Select **Grant**, **Deny**, or **Inherit** to assign permissions.

Tables and Columns

Select a **federation server** ⇒ **data service** ⇒ **catalog** ⇒ **schema** ⇒ **table (column)** and use the following task to grant permissions:

1. Click the **Authorizations** tab associated with the selected object.
2. Under **Identities**, select .
3. Enter a user or group name in the search box and click **Search**. To return a list of available user names and groups, leave the search box empty and click **Search**.

Note: The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.

4. Select the user (or users) and click **Add**. The user object is added to the **Identities** list.
5. Select the user from the Identities list, and set permissions by selecting one or more items in the **Permission** list.


Note: Any permissions that are currently assigned to a user object are dimmed on the **Permissions** toolbar and cannot be selected.

6. Select **Grant**, **Deny**, or **Inherit** to assign permissions.

Note: You can also configure authorizations on the rows in a table. See ‘Row-Level Security’ for more information.

Views

Select a **federation server** ⇒ **data service** ⇒ **catalog** ⇒ **schema** ⇒ **view** and use the following task to grant permissions:

1. Click the **Authorizations** tab associated with the selected object.
2. Under **Identities**, select .
3. Enter a user or group name in the search box and click **Search**. To return a list of available user names and groups, leave the search box empty and click **Search**.

Note: The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.

4. Select the user (or users) and click **Add**. The user object is added to the **Identities** list.
5. Select the user from the Identities list, and set permissions by selecting one or more items in the **Permission** list.

Note: Any permissions that are currently assigned to a user object are dimmed on the **Permissions** toolbar and cannot be selected.

6. Select **Grant**, **Deny**, or **Inherit** to assign permissions.

Note: You can also configure authorizations on the rows in a view. See ‘Row-Level Security’ for more information.

Chapter 7

Configuring Access to Data Sources

Working with Data Services	45
Creating a Data Service	46
Defining a New Data Service	46
Apache HIVE Data Service	46
DB2 Data Service	48
Greenplum Data Service	50
MDS (Memory Data Store) Data Service	51
Netezza Data Service	52
ODBC Data Service	53
Oracle Data Service	55
PostgreSQL	56
SAP Data Service	58
SAP HANA Data Service	59
SAS Federation Server Data Service	61
SASHDAT Data Service	62
SPDS Data Service	64
SQL Server Data Service	65
Teradata Data Service	66
Using the Generic Data Service Template	68
Editing a Data Service	69
Working with Data Source Names (DSNs)	69
Overview	69
DSN Types	69
Creating a DSN	70
Establishing DSN Permissions	71
Overview	71
Establishing Connect Permissions	72

Working with Data Services

Data services control information that identifies the location of data source tables. If a data source does not support native catalogs, you can use SAS Federation Server Manager to define a logical catalog name that serves as an SQL identifier. This enables you to identify each data source uniquely when performing heterogeneous operations.

Data services that require logins must be associated with a domain in SAS Metadata Server. When users connect to the data service through a data source name (DSN), the domain name is used to retrieve the user credentials associated with that data service. The credentials are then passed along to the back-end database. User credentials are stored in SAS Metadata Server.

Note: To manage data services in SAS Federation Server Manager, an administrator must have personal credentials to the associated databases. SAS Federation Server Manager connects to a data service behind the scenes using a credential search order of PERSONAL (CSO=PERSONAL). This means that all users attempting to connect to the data service must have their own database credentials.


The administrator can assign privileges to allow users access to the data source. In order to connect, a user must be granted CONNECT privilege on either the SAS Federation Server, a specific data service, or a specific DSN.

Creating a Data Service

Defining a New Data Service


Data services contain connection information and driver specifics to connect with data sources. When a data service is created, SAS Federation Server automatically generates a corresponding DSN that matches the name of the new data service. If the data service name conflicts with the name of an existing DSN, a DSN will not be generated for the data service.

When you create a data service, the configuration options that are presented in the dialog boxes depend on the data source, or service type that you are configuring. Use the following procedure to create and configure a data service in SAS Federation Server Manager.

1. Select a federation server object in the tree and log on if you have not already done so.
2. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
3. At the Identification dialog box, enter the **Name** of the data service and click **Next** to continue.
4. Select a data source from the service type list and follow the instructions that apply to your data source. If your data source is not listed below, use the [Generic data service template](#) to configure your data service.

Apache HIVE Data Service

Use the following procedure to create a data service for Apache Hive.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. For the service type, select **Apache HIVE** and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers.

- a. Select **Configure a native Apache HIVE driver** and update the fields shown below. The required fields are marked with a red asterisk in the dialog box.
 - Enter a server name.
 - Enter a port number.
 - Enter a schema name.
 - Enter a path to the configuration file where the Hadoop JAR files are located.
 - Enter the subprotocol type.
 - Select the mode of authentication.

Note: See the "SAS Federation Server Driver for Apache Hive" in the *SAS Federation Server Administrator's Guide* for details about these connection options.

- b. Select **Configure an ODBC driver** and update the following configurations.
 - Enter a DSN name or a driver connection string.
 - Enter the name for the default schema.

Figure 7.1 Apache Hive Data Source Configuration

The screenshot shows a dialog box titled "Properties - Hive_N_J6". It has two main sections. The first section, "Data Source", contains the instruction "Configure one or more drivers to use to connect to the data source." and a checked checkbox "Configure a native Apache HIVE driver". Below this are several fields: "Server:" with the value "cdh53d1", "Port:" with the value "10000", "Schema:" with the value "engine", "Configuration file:" (empty), "Sub protocol:" (empty), and "Authentication mode:" with a dropdown menu set to "default". The second section contains an unchecked checkbox "Configure an ODBC driver" with a help icon. Below it are fields for "ODBC DSN:" (empty), "Driver connection string:" (empty), and "Default schema:" (empty).

- c. If you have configured more than one driver, select a **Default driver**.
- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.

- Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see "SAS Federation Server Driver for Apache Hive" in the *SAS Federation Server Administrator's Guide*.

- Click **OK** when you are finished.
- Click **Next** to continue.

5. Specify a **Catalog** name and click **Next** to continue.
6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. At the Summary, verify the settings and click **Finish**.

DB2 Data Service

Use the following procedure to create a data service for DB2.


1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **DB2** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers. The required fields are marked with a red asterisk in the dialog box.
 - a. Select **Configure a native DB2 driver** and enter the *Database name*.
 - b. Select **Configure an ODBC driver**, and specify either a DSN name or a driver connection string.

Figure 7.2 DB2 Data Source Configuration

Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native DB2 driver

Database: DB2_Database

☒ Configure an ODBC driver ?

ODBC DSN: DB2_DSN

Driver connection string:

Default driver:

☒ DB2

☐ ODBC

Advanced Driver Options

- c. If you have configured more than one driver, select a **Default driver**.
 - d. Click **Advanced Driver Options** if additional configuration is required for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see the "SAS Federation Driver for DB2" topic in the *SAS Federation Server Administrator's Guide*.

 - Click **OK** when you are finished.
 - Click **Next** to continue.
 5. Specify a **Catalog** name and click **Next** to continue.
 6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.
- Note:*
1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
 2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. At the Summary, verify the settings and click **Finish**.

Greenplum Data Service

Use the following procedure to create a data service for Greenplum.


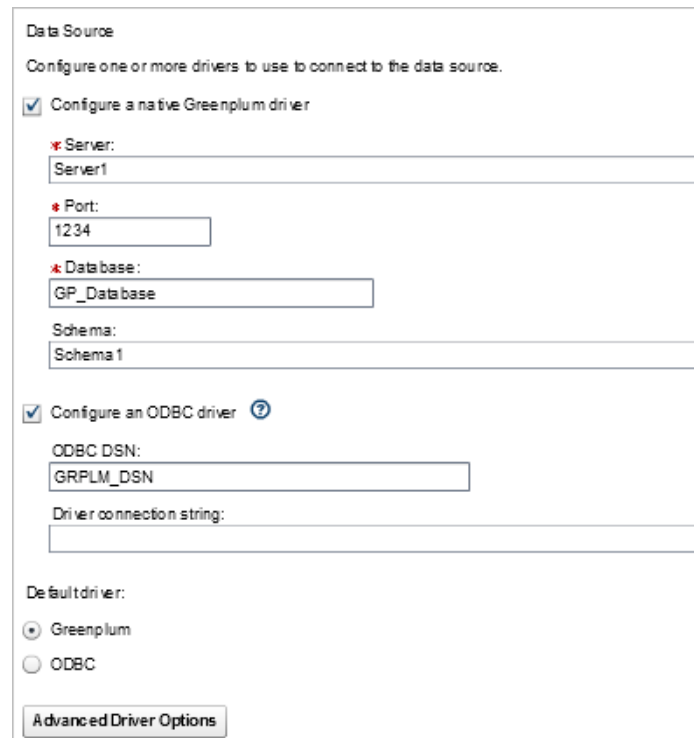
1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **Greenplum** as the service type and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers.
 - a. Select **Configure a native Greenplum driver** and update the following configurations. The required fields are marked with a red asterisk in the dialog box.
 - Enter the name of the database server.
 - Enter the port number for the server.
 - Enter the name of the database.
 - Enter a schema name.
 - b. Select **Configure an ODBC driver**, and specify either a DSN name or a driver connection string.

Figure 7.3 Greenplum Data Source Configuration



Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native Greenplum driver

* Server:
Server1

* Port:
1234

* Database:
GP_Database

Schema:
Schema1

☒ Configure an ODBC driver ?

ODBC DSN:
GRPLM_DSN

Driver connection string:

Default driver:

☒ Greenplum

☐ ODBC

Advanced Driver Options

- c. If you have configured more than one driver, select a **Default driver**.
- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see the "SAS Federation Server Driver for Greenplum" in the *SAS Federation Server Administrator's Guide*.


- Click **OK** when you are finished.
 - Click **Next** to continue.
5. Specify a **Catalog** name and click **Next** to continue.
 6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. At the Summary dialog box, verify the settings and click **Finish**.

MDS (Memory Data Store) Data Service

Use the following procedure to create a data service for MDS.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **MDS (Memory Data Store)** as the service type and click **Next** to continue.
4. At the Data Source dialog box, configure the values below. The required fields are marked with a red asterisk in the dialog box.
 - a. Enter a value, in bytes, that represents the maximum memory size for the database.

Note: A blank or zero value indicates that no memory limit is set for the database.
 - b. Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.
 - Click **OK** when you are finished, and click **Next** to continue.
5. Specify a *catalog name* that will contain all schemas belonging to the MDS data service.
6. At the Summary dialog box, verify the settings and click **Finish**.

Netezza Data Service

Use the following procedure to create a data service for a Netezza data source.


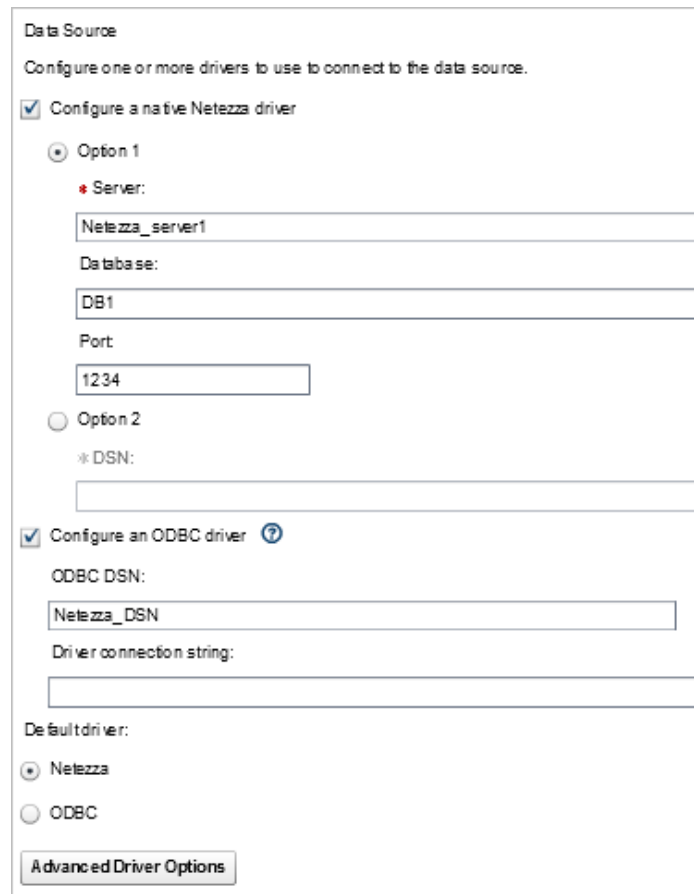
1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **Netezza** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers.
 - a. Select **Configure a native Netezza driver** and update the following configurations. The required fields are marked with a red asterisk in the dialog box.
 - For Option 1, enter the name of the database server.
 - Enter the name of the database.
 - Enter the port number for the server.
 - For Option 2, specify the name of the Netezza DSN.
 - b. Select **Configure an ODBC driver** and enter an ODBC DSN name, or a driver connection string.

Figure 7.4 Netezza Data Source Configuration



Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native Netezza driver

☒ Option 1

* Server:

Netezza_server1

Database:

DB1

Port:

1234

☐ Option 2

* DSN:

☒ Configure an ODBC driver ?

ODBC DSN:

Netezza_DSN

Driver connection string:

Default driver:

☒ Netezza

☐ ODBC

Advanced Driver Options

- c. If you have configured more than one driver, select a **Default driver**.

- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.

- Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see "Netezza Driver Reference" in the *SAS Federation Server Administrator's Guide*.

- Click **OK** when you are finished.
- Click **Next** to continue.

5. Specify a **Catalog** name and click **Next** to continue.
6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. At the Summary, verify the settings and click **Finish**.

ODBC Data Service

You can create an ODBC data service 'with native catalog support' or 'without native catalog support'. A data service 'without native catalog support' requires that you specify a catalog name. Use the following procedure to create an ODBC data service.


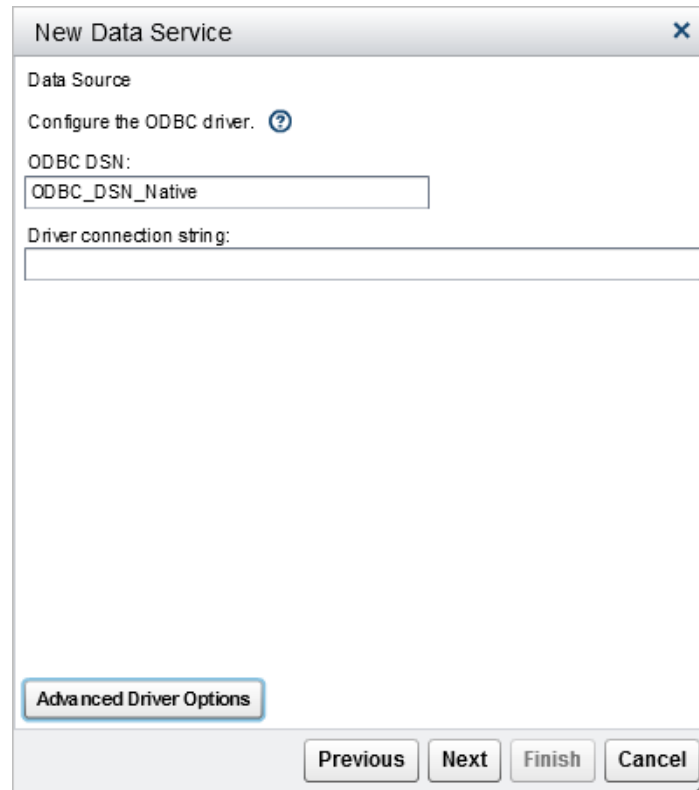
1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Specify the name of the new data service and click **Next**.
3. Select one of the following options from the **Service type** list:
 - ODBC with native catalog support
 - ODBC without native catalog support
4. In the Data Source dialog box, configure an ODBC driver to connect to the data source. The required fields are marked with a red asterisk in the dialog box.
 - a. Specify either a DSN name or a driver connection string.

Figure 7.5 ODBC Data Source Configuration



- b. Select **Advanced Driver Options** to enter additional connection options for the data source.

Note: Advanced options are specified as **<key>=<value>** pairs using a semicolon to separate pairs. For information about advanced options, see the "ODBC Driver Reference" in the *SAS Federation Server Administrator's Guide*.

5. When you are finished configuring the data source, click **Next** to continue.
6. This step is for ODBC without native catalog support only: Specify a *catalog name* and click **Next** to continue.
7. Choose one of the following options at the Authentication Domain dialog box:
 - If authentication is required when using the data service, accept the default **Database supports authentication** and select an associated **domain**. Click **Next** to continue.
 - If authentication is not required for the data service, uncheck **Database supports authentication** and click **Next** to continue.

CAUTION:

When creating a data service with or without authentication support, you cannot change properties of the data service to alter the behavior of authentication after setup has been completed. If the **Database supports authentication** option is selected for the data service, you can later edit the data service to change domains, but you cannot disable the check box for Database supports authentication. The same rule applies to a data service that is configured without authentication support. You cannot alter the properties of the data service later to select the Database supports authentication check box.

8. At the Summary dialog box, review the configuration and click **Finish**.

Oracle Data Service

Use the following procedure to create a data service for an Oracle data source.

1. Select **New Data Service** from the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **Oracle** as the service type and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers.
 - a. Select **Configure a native Oracle driver** and update the following configurations. The required fields are marked with a red asterisk in the dialog box.
 - Specify the path to an Oracle connect identifier (for example, tnsnames.ora).

Note: The path is an Oracle connect identifier as defined in tnsnames.ora or other naming method. A connect identifier can be a net service name or a database service name that resolves to a connect descriptor.
 - b. Select **Configure an ODBC driver**, and specify either a DSN name or a driver connection string.

Figure 7.6 Oracle Data Source Configuration

Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native Oracle driver

* Path: tnsora

☒ Configure an ODBC driver ?

ODBC DSN: ORA_DSN

Driver connection string:

Default driver:

☒ Oracle

☐ ODBC

Advanced Driver Options

- c. If you have configured more than one driver, select a **Default driver**.
- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced driver options, see the "SAS Federation Server Driver for Oracle" topic in the *SAS Federation Server Administrator's Guide*.

- Click **OK** when you are finished.
 - Click **Next** to continue.
5. Specify a **Catalog** name and click **Next** to continue.
 6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. At the Summary dialog box, verify the settings and click **Finish**.

PostgreSQL

Use the following procedure to create a data service for a PostgreSQL data source.


1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **PostgreSQL** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers. The required fields are marked with a red asterisk in the dialog box.
 - a. Select **Configure a native PostgreSQL driver** and configure one of the options below.
 - **Option 1:** Enter the name of the server, database, and associated port number.
 - **Option 2:** Specify the name of the DSN used to access the data source.
 - b. Select **Configure an ODBC driver**, and specify either a DSN name or a driver connection string.
 - c. If you have configured more than one driver, select a **Default driver**.

Figure 7.7 PostgreSQL Data Source Configuration

Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native PostgreSQL driver

☒ Option 1

* Server:

PostgreSQL_Server1

Database:

Postgres

Port:

1234

☐ Option 2

* DSN:

☒ Configure an ODBC driver ?

ODBC DSN:

Postgres_DSN

Driver connection string:

Default driver:

☒ PostgreSQL

☐ ODBC

Advanced Driver Options

- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.

- Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: See "SAS Federation Server Driver for PostgreSQL" in the *SAS Federation Server Administrator's Guide* for a list of available connection options.

- Click **OK** when you are finished.
- Click **Next** to continue.

5. Specify a **Catalog** name and click **Next** to continue.

6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.

7. At the Summary dialog box, verify the settings and click **Finish**.

SAP Data Service

Use the following procedure to create a data service for SAP.


1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **SAP** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure one of the items shown below. The required fields are marked with a red asterisk in the dialog box.
 - **SAPGUI logical name:** Specify an SAP Logon ID to use on a Windows 32-bit system.
 - **Application Server:** Specify the host name and system number of the application server. Use this option if load balancing is not in use.
 - **SAPRFC.INI logical name:** Specify the destination used in the SAPRFC.INI file, or the destination in the SAPNWRFC.INI file, if working with the NetWeaver RFC library and a sapnwrfc.ini file.
 - **Message Server:** Specify the message server host, the name of the R3 system, and logon group that applies to the application servers used for load balancing.

Figure 7.8 SAP Data Source Configuration



Data Source

Specify how to connect to the data source.

☐ SAPGUI logical name:

* SAP Logon ID:

☒ Application server:

* Host:

* System number:

☐ SAPRFC.INI logical name:

* Destination:

☐ Message server:

* Host:


* R3 name:

* Logon group:

Advanced Driver Options

- Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see "SAS Federation Server Driver for SAP" in the *SAS Federation Server Administrator's Guide*.

- Click **OK** when you are finished.
 - Click **Next** to continue.
5. Specify a **Catalog** name and click **Next** to continue.
 6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.
- Note:* If you want to change the Authentication Domain, select **Previous**, **Next** and select a new domain. Click **refresh**  after changing the domain.
7. At the Summary dialog box, verify the settings and click **Finish**.

SAP HANA Data Service

Use the following procedure to create a data service for SAP HANA.


1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **SAP HANA** from the service type list, and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers.
 - a. Select **Configure a native SAP HANA driver** and update the following configurations. The required fields are marked with a red asterisk in the dialog box.
 - **Option 1:** Enter the name of the server, server instance, and associated port number.
 - **Option 2:** Specify the name of the DSN used to access the data source.
 - b. Select **Configure an ODBC driver**, and specify either a DSN name or a driver connection string.

Figure 7.9 SAP HANA Data Source Configuration

Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native SAP HANA driver

☒ Option 1

* Server:

SAP_HANA_server1

Instance:

1

Port

1234

☐ Option 2

* DSN:

☒ Configure an ODBC driver ?

ODBC DSN:

SAP_HANA_DSN

Driver connection string:

Default driver:

☒ SAP HANA

☐ ODBC

Advanced Driver Options

- c. If you have configured more than one driver, select a **Default driver**.
- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.

- Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see "SAS Federation Server Driver for SAP HANA" in the *SAS Federation Server Administrator's Guide*.

- Click **OK** when you are finished.
 - Click **Next** to continue.
5. Specify a **Catalog** name and click **Next** to continue.
 6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.


Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.

7. At the Summary dialog box, verify the settings and click **Finish**.

SAS Federation Server Data Service

Use the following procedure to create a data service that allows connectivity between federation servers.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **SAS Federation Server** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure the items listed below. The required fields are marked with a red asterisk in the dialog box.
 - **Server**: Specify the name of the federation server.
 - **Port**: Specify the port number for the federation server.
 - **DSN**: Specify a DSN for the federation server.
 - Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see the "SAS Federation Server Driver Reference" in the *SAS Federation Server Administrator's Guide*.

 - Click **OK** when you are finished.
 - Click **Next** to continue.

Figure 7.10 Federation Server Driver Data Source Configuration

5. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.


Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
6. At the Summary dialog box, verify the settings and click **Finish**.

SASHDAT Data Service

The SAS Federation Server Driver for SASHDAT (Driver for SASHDAT) is a write-only driver designed for use with Hadoop on a grid host, such as the SAS LASR Analytic Server. SAS LASR Analytic Server integrates with Hadoop by storing SAS data in the Hadoop Distributed File system (HDFS). See the ‘SAS Federation Server Driver for SASHDAT’ in the *SAS Federation Server: Administrator’s Guide* for additional information.

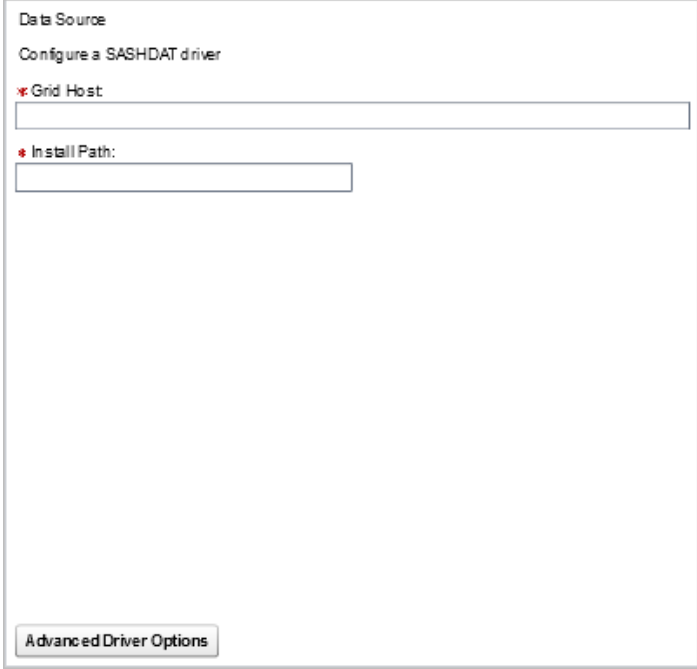
Use the following procedure to create a data service for SASHDAT.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **SASHDAT** as the service type and click **Next** to continue.
4. At the Data Source dialog box, configure the items below. The required fields are marked with a red asterisk in the dialog box.
 - **Grid Host:** specify the name of the grid host that has a running Hadoop NameNode.
 - **Install Path:** specify the path to the TKGrid installation on the grid host.
 - Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see "SAS Federation Server Driver for SASHDAT" in the *SAS Federation Server Administrator's Guide*.

 - Click **OK** when you are finished.
 - Click **Next** to continue.

Figure 7.11 SASHDAT Data Source Configuration



Data Source

Configure a SASHDAT driver

* Grid Host:

* Install Path:

Advanced Driver Options

5. Specify a **Catalog** name and click **Next** to continue.
6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an

@ *shared login key* to the domain, when the associated DSN is configured to use a shared login.

2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. At the Summary dialog box, verify the settings and click **Finish**.

SPDS Data Service

To create a data service for SPDS, use the **Generic without catalog support** data service template.


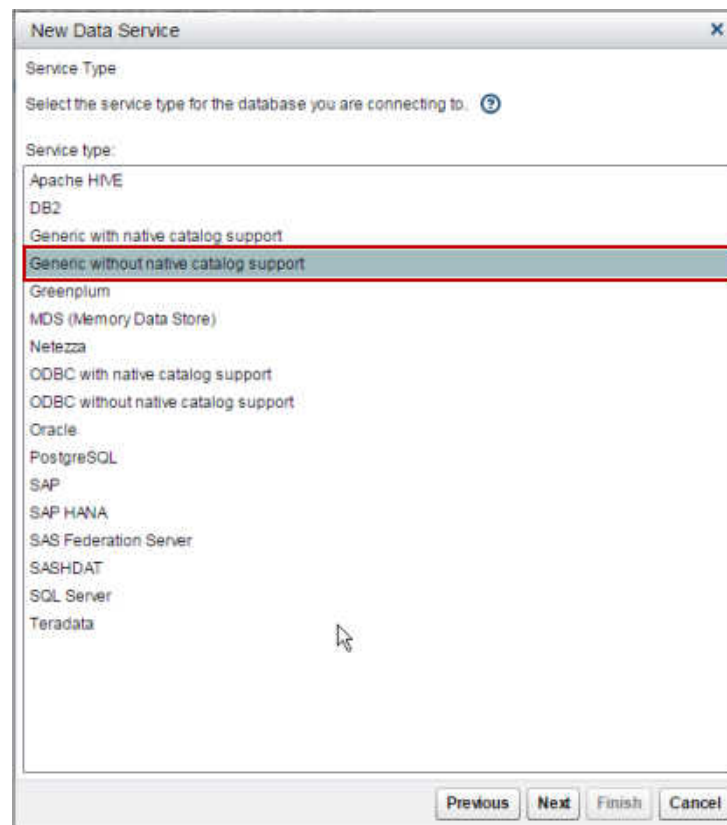
1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Specify the name of the new data service and click **Next**.
3. Select **Generic without native catalog support** from the **Service type** list.

Figure 7.12 Data Service Type for SPDS



4. In the Data Source dialog box, specify **SPDS** for the driver type name, and enter a connection string.

Figure 7.13 Driver Type and Connection String

New Data Service

Data Source

Configure the driver to use to connect to the data source.

* Driver Type Name:

SPDS

Driver connection string:

HOST=lax94d01;LOCALE=en_us;SERV=14512;SCHEMA=(NAME='PUBLIC';DBQ='PUBLIC')


Note: the connection string should be specified as <key>=<value> pairs using a semicolon (;) to separate pairs from each other.

Previous Next Finish Cancel

5. When you are finished configuring the data source, click **Next** to continue.
6. Specify a *catalog name* and click **Next** to continue.
7. If authentication is required when using the data service, accept the default **Database supports authentication**, and select an associated **domain**. Click **Next** to continue.
8. Review the summary and click **Finish**.

SQL Server Data Service

Use the following procedure to create a data service for SQL Server.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **SQL Server** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure the following items:
 - Specify either a DSN name or a driver connection string.
 - Click **Advanced Driver Options** if additional configuration is needed for the data service.
 - Enter advanced options with <key>=<value> pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.

Note: For information about advanced options, see "SAS Federation Server Driver for SQL Server" in the *SAS Federation Server Administrator's Guide*.

 - Click **OK** when you are finished.

- Click **Next** to continue.

Figure 7.14 SQL Server Data Source Configuration


5. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an *@ shared login key* to the domain, when the associated DSN is configured to use a shared login.
2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
6. Review the summary and click **Finish**.

Teradata Data Service

Use the following procedure to create a data service for a Teradata data source.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Enter the **Name** of the data service and click **Next** to continue.
3. Select **Teradata** from the service type list and click **Next** to continue.
4. At the Data Source dialog box, configure one or more drivers.
 - a. Select **Configure a native Teradata driver** and configure the following items. The required fields are marked with a red asterisk in the dialog box.
 - Specify a server name.

- Specify a database name.
- b. Select **Configure an ODBC driver**, and specify either a DSN name or a driver connection string.

Figure 7.15 Teradata Data Source Configuration

Data Source

Configure one or more drivers to use to connect to the data source.

☒ Configure a native Teradata driver

★ Server: TD_Server1

Database:

☒ Configure an ODBC driver ?

ODBC DSN: TD_DSN

Driver connection string:

Default driver:

☒ Tera data

☐ ODBC

Advanced Driver Options

- c. If you have configured more than one driver, select a **Default driver**.
- d. Click **Advanced Driver Options** if additional configuration is needed for the data service.
- Enter advanced options with **<key>=<value>** pairs using a semicolon to separate pairs. Use ASCII characters only when setting advanced options.
- Note:* For information about advanced options, see the "SAS Federation Driver for Teradata" topic in the *SAS Federation Server Administrator's Guide*.
- Click **OK** when you are finished.
 - Click **Next** to continue.
5. Specify a **Catalog** name and click **Next** to continue.
6. Select an **Authentication Domain** from the list of available domains, and click **Next** to continue.

Note:

1. If you are configuring a data service that uses a shared login, select a stand-alone data source domain. Do not select the domain that is appended with a shared login key as created in SAS Metadata Server. SAS Federation Server appends an

@ *shared login key* to the domain, when the associated DSN is configured to use a shared login.


2. If you recently added a new domain in SAS Metadata Server and do not see it in the list of available authentication domains, click **refresh** in the upper right corner of the dialog box. The new domain should now be listed.
7. Review the summary and click **Finish**.

Using the Generic Data Service Template

If your data source does not appear in the service type list, you can use the Generic data service template to create a data service. There are two types of Generic data services that you can create:

- with native catalog support
- without native catalog support

A data service 'without native catalog support' requires that you specify a catalog name. Use the following procedure to create a data service using the generic data service template.

1. Select **New Data Service** from the **Action** menu list, or click the New Data Service icon  on the toolbar.
2. Specify the name of the new data service and click **Next**.
3. Select one of the following options from the **Service type** list:
 - Generic with native catalog support
 - Generic without native catalog support
4. In the Data Source dialog box, specify a driver name and a connection string.

Note: Options are specified as **<key>=<value>** pairs using a semicolon to separate pairs.
5. When you are finished configuring the data source, click **Next** to continue.
6. This step is for Generic without native catalog support only: Specify a *catalog name* and click **Next** to continue.
7. Choose one of the following options at the Authentication Domain dialog box:
 - If authentication is required when using the data service, accept the default **Database supports authentication** and select an associated **domain**. Click **Next** to continue.
 - If authentication is not required for the data service, uncheck **Database supports authentication** and click **Next** to continue.

CAUTION:

When creating a data service with or without authentication support, you cannot change properties of the data service to alter the behavior of authentication after setup has been completed. If the **Database supports authentication** option is selected for the data service, you can later edit the data service to change domains, but you cannot clear the **Database supports authentication** check box. The same rule applies to a data service that is configured without authentication

support. You cannot alter the properties of the data service later to select the **Database supports authentication** check box.

8. Review the summary and click **Finish**.

Editing a Data Service

Use the following procedure to edit a data service.

1. Select a **federation server** ⇒ **data service** in the tree.
2. Select the **Action** menu on the **Summary** tab located in the right panel.
3. Select the item in the drop-down menu that corresponds to the operation that you want to perform.
 - Select **Test Connection** to test the connection to the data service.
 - Select **Properties** to edit the data service.
 - Select **Delete** to remove the data service.
 - Select **Rename** to rename the data service.

Note: You cannot edit the BASE or SQL_LOG data services because these objects are created and owned by the system.

Working with Data Source Names (DSNs)

Overview

DSNs are displayed only when logged on to a federation server. A DSN references a specific data service through which it connects, and defines how SQL security is enforced. It can be configured so that SAS Federation Server enforces SQL privileges defined for the data service. You can create a standard, single-service DSN or a federated DSN, which is a collection of one or more standard DSNs. For additional information about DSN types and configuration options, see the *SAS Federation Server Administrator's Guide*.

DSN Types

Standard DSN

A standard DSN is a single-service DSN created for a particular data service and is parented to that data service. The scope is limited to one data service and contains connection information, such as server name, port, path, or other connection options specific to a data service.

Federated DSNs


A federated DSN is a collection of one or more standard DSNs. Unlike the standard DSN, which is parented to a data service, the federated DSN is parented to the federation

server itself, even if it contains DSNs from only a single data service. Federated DSNs can contain other federated DSNs.

Creating a DSN

Create a Standard DSN

Use the steps below to create a standard DSN.

1. Select a **federation server** in the tree.
2. Select the **Data Source Names** tab and click the **New Standard Data Source Name** icon  on the toolbar.
3. Specify the *Name* and *Description* of the new DSN and click **Next**.
4. Select the **Data Service** that is associated with the new DSN, and click **Next** to continue.
5. **Enable Federation Server SQL Authorization** is selected by default. Accept the default or uncheck the option to disable it. Click **Next**.
6. Select the type of access for the DSN: **Personal** login or **Shared** login.

- a. If you specify a Shared login, select a consumer group from the drop-down menu.

Note: The Consumer group identifies the authorizations to use should the user exist in more than one group. Consumers are SAS Federation Server user accounts or groups. You should never use the actual shared login group as a consumer group in a DSN.

Note: When shared login is selected on the DSN, SAS Federation Server appends the selected domain with an **@shared login key**, and verifies that **data_source@<shared login key>** exists in SAS Metadata as a valid authentication domain that also includes user account information.

- b. If both Personal and Shared logins are specified, select a credentials search order from the options under **Access Order**.

Note: By default, login credentials are searched in this order: Personal, Group, and Shared Login.

7. At the syntax dialog box, select **FedSQL** or **DS2** dialect and click **Next** to continue.
8. Specify optional parameters for the DSN and click **Next** to continue.
- Note:* If fields are left blank, SAS Federation Server Manager uses the values that are set in the federation server configuration files.
9. Review the summary information and click **Finish**.

Creating a DSN with FedSQL or DS2 Dialects

Use the steps below to create a standard DSN that uses the FedSQL or DS2 dialects.

1. Select a **federation server** in the tree.
2. Select the **Data Source Names** tab and click the **New Standard Data Source Name** icon in the toolbar.
3. The New Standard Data Source Name dialog box appears. Specify the *name* and *Description* of the new DSN and click **Next**.
4. Select a data service that will contain the new DSN and click **Next** to continue.

5. **Enable Federation Server SQL Authorization** is selected by default. You can accept this option or disable it. Click **Next** to continue.

Note: Disabling Federation Server SQL Authorization Enforcement allows for **Native SQL dialect**.


6. At the Syntax dialog box, select **FedSQL dialect** or, **DS2**. If Federation Server SQL Authorization Enforcement is disabled, **Native SQL dialect** is selected by default.
7. Specify optional parameters for the DSN and click **Next** to continue.

Note: If fields are left blank, SAS Federation Server Manager uses the values that are set in the federation server configuration files.

8. Review the summary information and click **Finish**.

Creating a Federated DSN

Since federated DSNs are parented to a federation server, they must be created from a federation server object. You cannot create a federated DSN under a data service. Use the steps below to create a Federated DSN.

1. Select a federation server object in the tree.
2. Select the **Data Source Names** tab and click the **New Federated Data Source** icon  on the toolbar.
3. Specify the *Name* and *Description* of the new data source name and click **Next**.
4. At the New Federated Data Source Name dialog box, select **Add**.
5. The Add Data Source Names dialog box appears. Select the DSNs to add to the Federated DSN and click **OK**.

Note: Ensure that the child DSNs are not pointing to the same catalog, as this might result in a catalog conflict error. You cannot have duplicate catalog names within the connection.

6. Click **Next** at the Members dialog box.
7. **Enable Federation Server SQL Authorization** is selected by default. You can accept this option or disable it. Click **Next** to continue.

Note: Disabling Federation Server SQL Authorization allows for **Native SQL dialect**.


8. At the Syntax dialog box, select **FedSQL dialect** or, **DS2**. If Federation Server SQL Authorization Enforcement is disabled, **Native SQL dialect** is selected by default.
9. Select **Finish** at the Summary dialog box.


Establishing DSN Permissions

Overview

A user must have CONNECT permission to establish connection with a DSN. Use the procedure below to establish connect permissions for users and groups.

Establishing Connect Permissions

1. Select a **federation server** object in the tree.
2. Open the **Data Source Names** tab and select a DSN.
3. Select  on the **Connect Permission tab**, select **Add users and groups**.

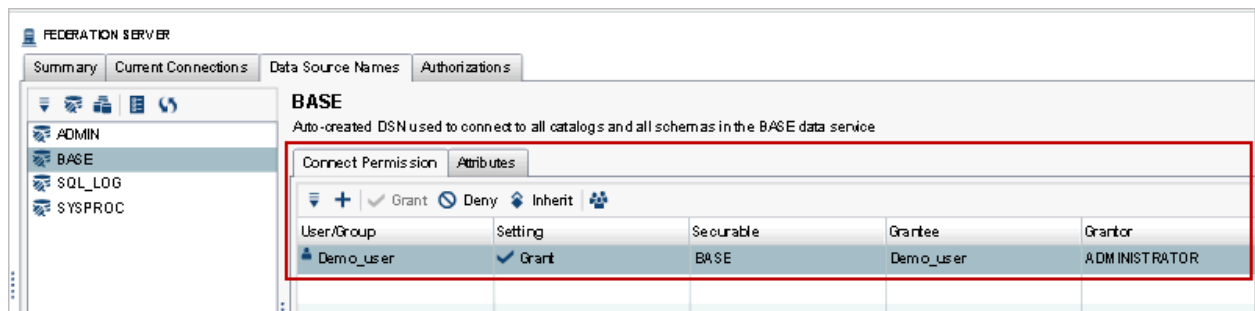
Note: You can also use the Add users and groups icon  on the toolbar.

4. Enter a user or group name in the search box and click **Search**.
5. Select the user (or group) and click **Add**. The user object now appears in the connect permission list.

TIP You can add multiple users by holding down the **Ctrl** key while selecting a user or group object.

6. Click **Close** to exit the Add users and groups dialog box.
7. Select the user object and click **Grant** on the toolbar.

Figure 7.16 Granting Connect Permission for a DSN



Part 4

Working with Federated Data

<i>Chapter 8</i>	
Working with Catalogs and Schemas	75
<i>Chapter 9</i>	
Working with FedSQL Views	81
<i>Chapter 10</i>	
Caching Data	85
<i>Chapter 11</i>	
Configuring Row-Level Security	97

Chapter 8

Working with Catalogs and Schemas

Catalogs and Schemas	75
Overview	75
Working with Catalogs	75
Working with Schemas	77

Catalogs and Schemas

Overview

All data services contain a catalog object. Databases that support catalogs contain native catalogs that are retrieved from the associated data source. If a database does not support catalogs, you can create a logical catalog when the data service is created, or by using the **New Catalog** function in SAS Federation Server Manager. See the *SAS Federation Server Administrator's Guide* for additional information about catalogs and schemas.

Working with Catalogs

Creating a Catalog for Base SAS

Catalogs and schemas are required for the BASE data service to expose data. Use the following procedure to create a new catalog for a Base data service:

1. Select a **BASE data service** in the tree and select the **Action** menu in the upper left corner of the tree.
2. Select **New Catalog** from the drop-down menu.
3. At the New Catalog dialog box, enter a *catalog name*.
4. Click **OK** to save the catalog and exit.

When the new catalog is added to the data service, two tabs are displayed in the right pane: Summary and Authorizations. By selecting the catalog object in the tree, you can use the **Action** menu in the tree, or on the **Summary** tab to perform the following actions:

- Create a new schema.
- Test the connection.
- Rename the catalog.

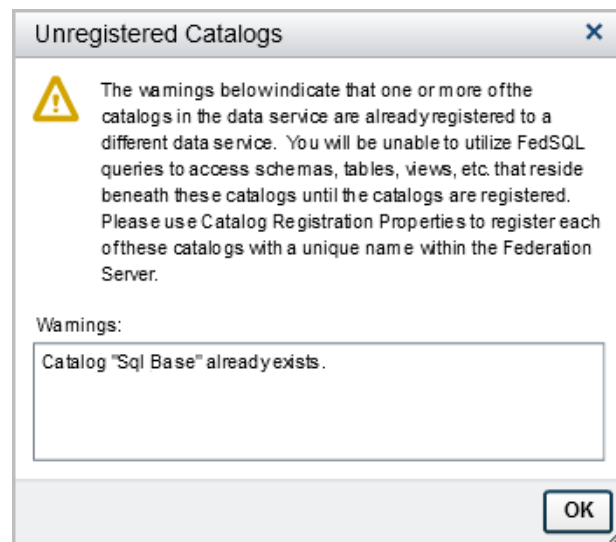
- Delete the catalog.

Use the **Authorizations** tab to grant or deny privileges for the new catalog.

Naming Conflicts and Unregistered Catalogs

Normally catalogs are registered by default when a data service is created. If a catalog is not registered, a warning icon is displayed next to each unregistered catalog in the tree and also in the **Summary** tab for each individual catalog object. Note: This warning is also displayed for SQL Server catalogs that have naming conflicts because they are registered to a different data service. These naming conflicts should be resolved, and the catalogs registered before attempting to set privileges for the table. Unregistered catalogs in the table will not reflect the new privileges. When you expand a data service in the tree, SAS Federation Server Manager checks the registration status of all the catalogs under the selected data service. If catalogs have not been registered or are already registered to a different data service, a warning is displayed.

Figure 8.1 Unregistered Catalogs Warnings

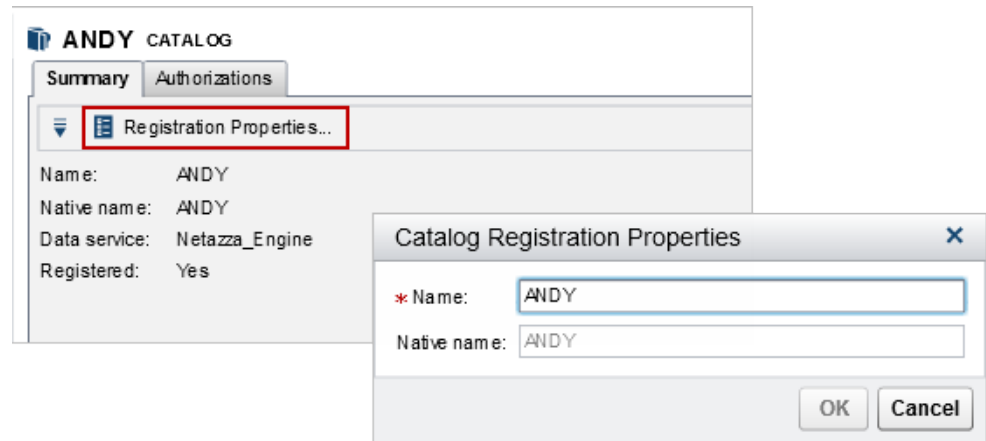


Registering a Catalog

If a data source supports native catalogs, a native catalog name is registered by default when a data service is created. Only SQL Server, Netezza, and SAS Federation Server use catalog registration. Use the following procedure to register a catalog:

1. Select an unregistered catalog in the tree.
2. Select the **Action** menu in the upper left corner and select **Registration Properties** from the drop-down menu.
3. At the Catalog Registration Properties dialog box, enter a *name* for the catalog and click **OK**.

Figure 8.2 Catalog Registration Properties



Working with Schemas

About Schemas

You can create new schemas for the BASE, MDS, and SASHDAT data services only. Other data services retrieve their schemas from the associated data source.

TIP All Base catalogs require a schema. A Base data service cannot establish connection if a schema is not detected for each Base catalog in the tree.

Note: Note that the MDS data service contains a default schema named SYSTEMINFO that is reserved for system use only. Therefore, you must create a new schema for your MDS data service.

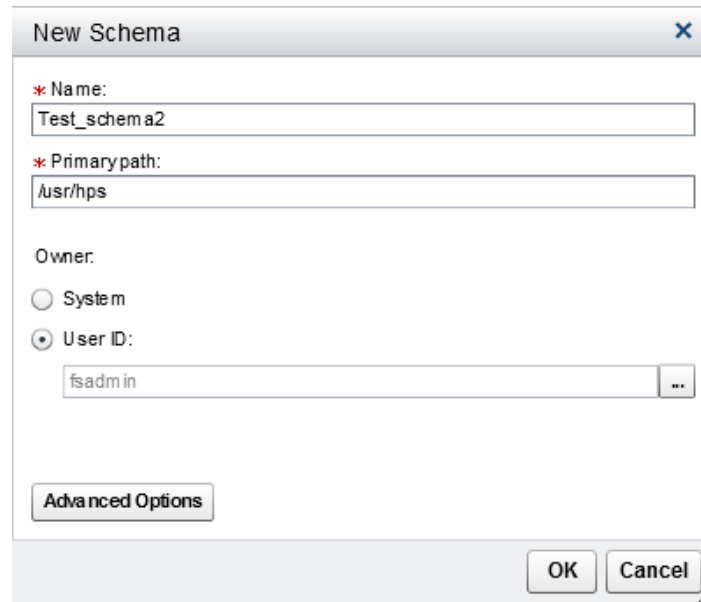
Creating a Schema

Use the following procedure to create a schema for Base SAS, MDS, and SASHDAT data sources.

1. Select a **federation server** ⇒ **data service** ⇒ **catalog** that will contain the new schema.
2. With the catalog selected, click the **Action** menu in the upper left corner of the tree.
3. Select **New Schema** from the drop-down menu.
4. At the New Schema dialog box enter the *schema name* and *primary path* for the schema.

Note: You do not have to specify a primary path when creating a schema for MDS.

5. Under Owner, select **User ID** and click browse to open the Select User dialog box.
6. At the Select User dialog box, select a user name from the list and click **OK**.
7. Base SAS schema only: Select **Advanced Options** and enter additional schema options. Specify advanced options as **key=value** pairs using a semicolon to separate pairs from each other.
8. Click **OK** to return to the New Schema dialog box.
9. Click **OK** to create the schema.

Figure 8.3 New Schema Dialog Box


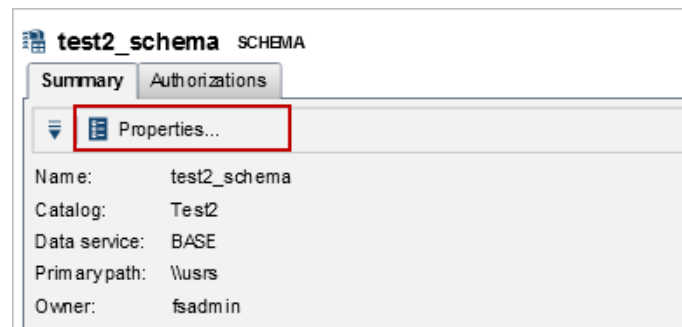
The 'New Schema' dialog box contains the following fields and options:

- Name:** A text field containing 'Test_schema2'.
- Primary path:** A text field containing '/usr/hps'.
- Owner:** Two radio buttons: 'System' (unselected) and 'User ID:' (selected).
- User ID:** A text field containing 'fsadmin' with a browse button to its right.
- Advanced Options:** A button located below the User ID field.
- Buttons:** 'OK' and 'Cancel' buttons at the bottom right.

Assigning Schema Ownership

Use the following procedure to assign or change a schema owner.

1. Select a federation server object that is associated with the data source containing the schema that you are updating.
2. Select the schema object by navigating through **data service** ⇒ **catalog**.
3. Under the **Summary** tab to the right, select **Properties**.

Figure 8.4 Schema Summary Tab


The 'test2_schema' SUMMARY tab displays the following properties:

Name:	test2_schema
Catalog:	Test2
Data service:	BASE
Primary path:	/usr
Owner:	fsadmin

A red box highlights the 'Properties...' button in the left sidebar.

4. In the Schema Properties dialog box, select **User ID** and click browse to open the Select User dialog box.

Figure 8.5 Schema Properties

Schema Properties - test2_schema

* Primary path:
/usr

Owner:
☐ System
☒ User ID: fsadmin

Advanced Options

OK Cancel

Note: If SYSTEM is shown as the owner of the schema, you might encounter problems when creating FedSQL views. It is a best practice to designate a schema owner other than the SYSTEM user.

5. Select a user from the list and click **OK** to return to the Schema Properties dialog box. Notice that the current schema owner is grayed out in the Select User dialog box.

Figure 8.6 Selecting a New Schema Owner

Select User

Users:
Search: Search

Name	Display Name	Type
a'b	a'b	User
brdbdg	brdbdg	User
brdtsg	brdtsg	User
Demo_user	Demo_user	User
fsadmin	fsadmin	User
Guest	Guest	User
sasadm	SAS Administrator	User
sasevs	SAS Environment Manager Se...	User
sasfedadm	SAS Federation Server Syste...	User
sasmdt	SAS Data Remediation Servic...	User

OK Close

6. The User ID field now reflects the new schema owner. Click **OK** to close the Schema Properties dialog box.

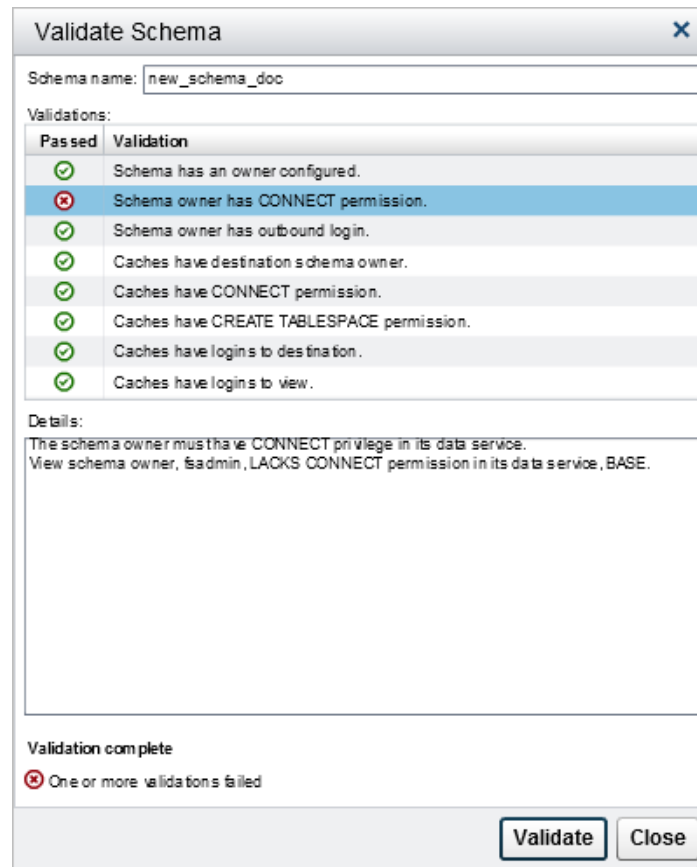
Validating a Schema

Use the following procedure to validate a schema.

1. Select a **federation server** ⇒ **data service** ⇒ **catalog** ⇒ **schema** objects in the tree.
2. With the schema selected, click the **Action** menu on the **Summary** tab.
3. Select **Validate Schema** from the drop-down menu.
4. In the Validate Schema dialog box, click **Validate**.

After validation, SAS Federation Server Manager displays a list of items that have passed or failed the validation test with a summary of failed items listed in the details section. Review each item and take corrective action if necessary. Here is an example of failed validation:

Figure 8.7 Validating Schemas



In this case the schema owner must have CONNECT privilege on the associated data service.

Renaming or Deleting a Schema

Using the **Action** menu on the **Summary** tab, you can also rename or delete a schema. MDS does not allow you to rename or delete a schema if there are any active connections on the database.

Chapter 9

Working with FedSQL Views

Working with FedSQL Views	81
Overview	81
Invoker's and Definer's Rights View	81
Creating a FedSQL View	82
Creating a View from Native Sources	82
Modifying a View	83
Deleting a View	83

Working with FedSQL Views

Overview

A federated data view (FedSQL view) contains the information required to access database sources and is stored separately from the data. By creating a view definition, you are storing only the instructions for where to find the data and how it is formatted, not the actual data. When you need to view information from multiple data sources or other source types, you can create a reusable FedSQL view to deliver data from multiple relational and non-relational data sources. You can also create a materialized view of the data by creating a cache from a FedSQL definer's rights view.

Note: Administration of FedSQL views and caches is not available when using a remote SAS Federation Server (FEDSRV) data service. You must always establish a direct connection to the remote SAS Federation Server to administer these objects.

Invoker's and Definer's Rights View

There are two types of FedSQL views:

Invoker's Rights View


The invoker's rights view is accessed using the current user's authorization, credentials, and login information.

Definer's Rights View

A definer's rights view is accessed using the schema owner's authorization, credentials and login information, and is always associated with a schema owner. A view must be set as definer's rights to cache

Creating a FedSQL View

To create a new FedSQL view, navigate to the schema where the view will reside and use the following procedure.

1. Select the schema where the view will reside.
2. Select **Create FedSQL View** on the toolbar. You can also use the action menu to create a **New FedSQL View**.
3. Enter a name for the view and select a security setting to specify privileges for accessing the view.
Note: If the view will be cached, you should use the definer's rights privileges (schema owner's privilege) to create the view. Only a definer's rights view can be cached.
4. Click **OK** to continue. A new tab with the name of the view opens where you can build the SQL for the query. The CREATE VIEW statement is already created based on the selections made while defining the new view.
5. Using the **Query** tab, select additional data for the CREATE statement and move it to the query panel using the arrow.
6. For the SELECT statement, use Data on the **Query** tab to expand a table expose columns.
7. Select a column and click the arrow  to move columns to the SELECT statement.

TIP You can select multiple columns by holding the **shift** key while selecting columns.

8. After you have built a SELECT statement, add a FROM statement that includes one or more data sources. Position your cursor on a new line below the last-selected column and type FROM. Immediately following, select one or more tables from the available source tables on the **Query** tab.
9. Click **Create View** after you have finished defining the view.

Note: If you receive an error message about duplicate naming, use the **Settings** tab to modify the name of the view.



CAUTION:

When creating a definer's rights view, it is best to assign a schema owner. If the schema owner is not set or the owner is SYSTEM, a warning message indicates that the view might not operate correctly. You can proceed but be sure to set a schema owner when you are finished creating the view.

Creating a View from Native Sources

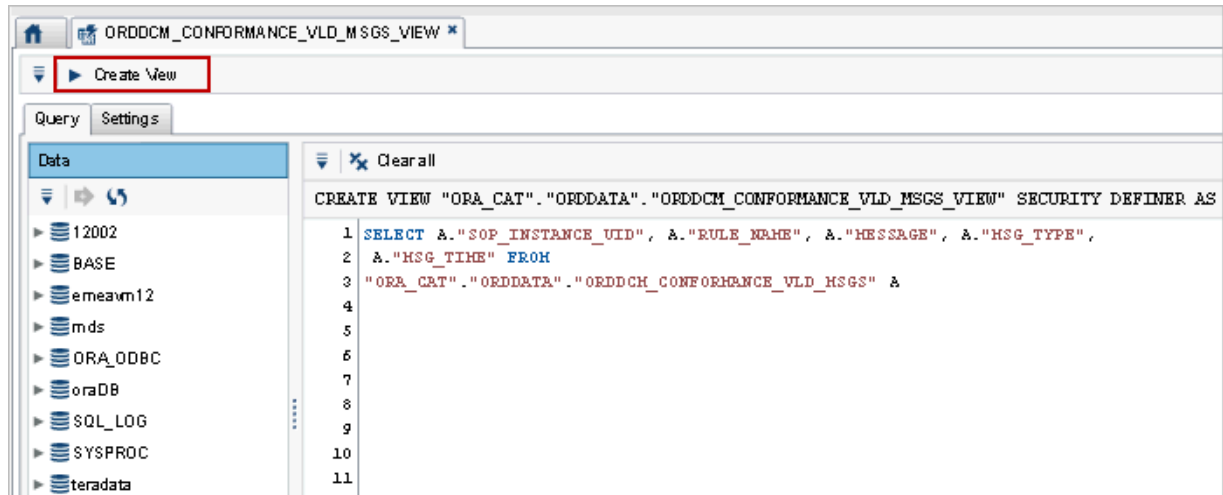
Create a View from a Table

Use the following procedure to create a FedSQL view from a table:

1. Select a **table**  in the tree.
2. Select  in the upper left corner of the tree.
3. Select **New FedSQL View from Table** from the drop-down menu.


4. The New FedSQL View from Table dialog box appears with a default view name. You can change the view name if desired.
5. Choose a security setting for the view and click **OK**.
6. A new table opens, displaying the query, which should include all the columns from the original table. You can make adjustments to the query as needed.
7. Click **Create View**.

Figure 9.1 Creating a FedSQL View



Creating a View from a Native View

Use the procedure below to create a FedSQL view from a native view.

1. Select a **native view**  in the tree.
2. Using the **Action** menu in the upper left corner of the tree, select **New FedSQL View from View**.
3. If necessary, change the view name, and select a security setting (for example **use the definer's privileges when accessed**).
4. Click **OK** to continue. A new tab opens, showing the query and associated with the view. You can use this to edit the query and other settings of the FedSQL view.
5. Make the necessary changes and click **Create View**.

Modifying a View

You can modify a view's security settings from Properties of the **Action** menu. However, you cannot modify the contents of the view (SQL statement or query) after the view is created. You must drop and re-create a new view with modifications.

Deleting a View

You can delete a view using the **Action** menu in the tree or on the **Summary** tab. Deleting a view deletes both the FedSQL view and the cache definition that is associated with the view. Follow these steps to delete a view.

1. Open the **Action** menu and select **Delete FedSQL View** from the list menu.

2. Click **OK** to confirm the deletion.

Chapter 10

Caching Data

Caching Views	85
Overview	85
Prerequisites for Caching Views	85
Caching a FedSQL View	86
Configuring Advanced Properties for Cache	87
Disabling and Enabling Cache Tables	89
Deleting Cache Tables	90
Refreshing Cached Data	90
Overview	90
Manual Refresh	90
Scheduled Refresh	90
CRON Custom Scheduled Refresh	92
Working with the Schedule	94

Caching Views

Overview

SAS Federation Server uses FedSQL to enable users to cache data from a definer's rights view, creating a materialized view of the data. A materialized view is a snapshot of the target view from a specific point in time. Cache implementation requirements and other details related to working with cached data are outlined in the *SAS Federation Server Administrator's Guide*.

Data cache connections use a credential search order (CSO) of PERSONAL, SHARED. See Credentials Search Order for additional information.

Prerequisites for Caching Views

Definer's Rights Views and Cache

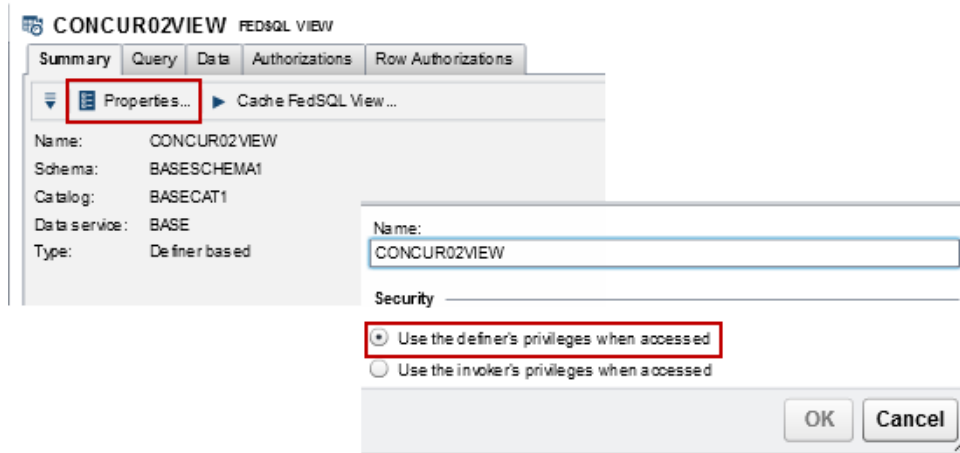
A definer's rights view uses the credentials of the schema owner. When the view is executed, it uses the credentials of the user that created, or defined the view rather than the credentials of the current user. Therefore, only a definer's rights view can be cached. When views are run, they access the catalogs that are referenced using the definer's credentials, even if the user is not currently connected to that catalog. If a definer's rights view is altered to an invoker view, the associated cache is dropped.

Changing a View from Invoker to Definer

If a view is an invoker's rights view, cache operations are not active for the view. Use the following procedure to change the view to a definer's rights view.

1. Open the **Action** menu that is associated with the view, and select **Properties**.
2. At the Properties dialog box, select **Use the definer's privileges when accessed** and click **OK**.

Figure 10.1 Changing FedSQL View Security

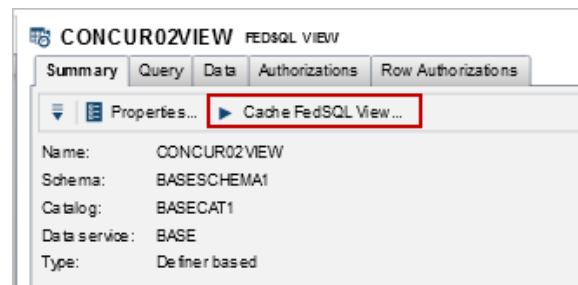


Caching a FedSQL View

Overview

A FedSQL view must be created before you can create a data cache of the view. Once a FedSQL view is created, the **Cache FedSQL View** icon appears in the summary tab of the view.

Figure 10.2 Caching a FedSQL View



Creating a Cache Definition

To create a cache, you must use a definer's rights view. Use the procedure below to create a cache definition.

1. Select a FedSQL view from the tree by selecting **Data Service** ⇒ **Catalog** ⇒ **Schema** ⇒ **View**.

CAUTION:

Creating a cache with a table that is owned by a system user can potentially cause problems and is not recommended.

2. From the **Summary** tab, click **Cache FedSQL View**.
3. Click the browse button to select a schema for the cached view and enter a description for the cache.
4. Select **Advanced Properties** to set other properties for the cache, or you can perform this step after the cache is created.
5. Select **Save Cache Definition**.

Note: You can also select **Create Cache Table**. Creating a cache table automatically saves the cache definition.

After the view is cached, you should receive a message confirming that the cache definition is saved indicating that you are currently using the view since a cache table was not created.

Creating a Cache Table

Use this procedure to create a cache table from a cache definition that was previously saved.

1. Navigate to the view and select it to display the cache definition.
2. Select the **Cache** tab and click **Create Cache Table**.

Delete and Other Cache Actions

Using the **Action** menu from the **Cache** tab, you can Refresh, Disable, or Delete a cache. To remove a cache, select **Delete Cache** from the **Action** menu located on the cache tab. Choosing Disable Cache temporarily disables the cache but does not remove it from the server.

Note: If a cache has a scheduled refresh and it is dropped, you must also delete the scheduled job. See [“Viewing Job History”](#) for additional information.

Configuring Advanced Properties for Cache

Use the **Cache Definition** function to edit or update advanced properties for a cache that has already been created. The cache definition function is a series of tabs that encompass the cache definition editor. Here is an explanation of each of the tabs.

General

The **General** tab shows the name of the view, the location of the cache tables, and a description of the cache. The name field is read only. You can change the location of the cache tables by selecting the **browse** button. There are two additional options on this tab:

- **On creation or population failure, delete all cache tablesFORCE:** When you select this option, the system reverts to the original view when the cache fails to create or populate with data. This option is not selected by default.
- **Refresh cache on server startup:** Select this check box if you would like the cache to persist after restarting the server. This option is selected as the default for all MDS caches.

Table Options

The **Table Options** tab contains options for the processing the data cache. By default, these options are blank.

- **Bulk Load:** Use Bulk Load to process large amounts of data. Enter an option or string in the **Bulk load** options field. The following example shows how to specify options for the log: **BL_LOG="C:/TEMP/bulkload.log"**
BL_LOAD_REPLACE=yes

Other bulk load options are available.

Note: Verify that your data source supports bulk loading because not all data services support this option. For example, BASE data sources do not support bulk load. See the *SAS Federation Server Administrator's Guide* for more information.

- **Database commit level:** sets a limit on the number of modified rows to commit at one time. This action affects transaction logging limits on the back-end database. This option overrides the ERRLIMIT option.
- **Insert buffer size:** sets a limit for the number of rows that can be inserted at one time. This action places a limit on a driver's row array size when inserting data.
- **Error limit:** sets a limit on the number of errors to allow before a statement stops inserting data.
- **CT Preserve:** controls how data types are mapped. The options for CT Preserve are outlined below.

STRICT

Data type mapping is disabled. The requested type must exist in the target database. Therefore, type promotion does not occur. If the type does not exist, an error is returned.

SAFE

Target data types are upscaled only if they do not result in a loss of precision or scale. When character encodings are changed, the new column size is recalculated to ensure all characters can be stored in the new encoding.

FORCE

FORCE is the default for all drivers. The best corresponding target data type is chosen, even if it could potentially result in a loss of precision or scale. When character encodings are changed, the new column size is recalculated to ensure all characters can be stored in the new encoding.

FORCE_COL_SIZE

This option is the same as FORCE, except that the column size for the new encoding is the same as the original encoding. This option can be used to avoid column size creep. However, the resulting column might be too large or too small for the target data.

- **Other table options:** defines additional options for the cache table. Specify table options as **<key>=<value>** pairs using a space to separate pairs, for example, **DBCOMMIT=1000 INSERTBUF=100**. To impose exclusive locks to avert a user from inserting, updating, or deleting a BASE table, select **Specify other table options** and enter **locktable=exclusive**. SAS Federation Server Manager wraps the entry in braces so that it is formatted correctly for FedSQL. Here is an example **{option locktable=exclusive}**.

Table Definition

Select the check box for **Customize table definition** to enable this tab. Once the content is enabled, you can change column types using the list of supported Column Types. Here are the column requirements for a cached view table definition:

- Column names must be identical.
- Columns must be in the same order.

- Columns must contain compatible data types. For example, (n)(var)char in the view must be (n)(var)char in the data cache. This rule applies to all data types.

Column names in the custom table definition are quoted by default. If Native Syntax is used, column names might not be quoted as expected. The information in View's Definition is read only because the cache view must remain consistent with the original FedSQL view.

Note: Using native syntax in the table definition might require adjusting the cache escape to use one of the following: {**CACHE**}, {**CACHE_CATALOG**}, {**CACHE_SCHEMA**}, or {**CACHE_TABLE**}. In addition, the escape should be quoted according to the case sensitivity setting of the destination database.

Before

The BEFORE statement provides a way for a user to specify SQL that will be executed before the view data is cached. Multiple EXEC BEFORE statements are allowed, and will be executed in the order specified. IGNORE RC indicates that an error from this statement will not fail the data cache refresh operation. Use New Statement to create an ordered list of FedSQL statements.

Note: If you change a catalog name after setting a clause, you must manually update the cache.

After

AFTER Statement The AFTER statement provides a way for a user to specify SQL that will be executed after the view data cache is created and populated. Multiple EXEC AFTER clauses are allowed, and will be executed in the order specified. IGNORE RC indicates that an error from this statement will not fail the data cache refresh operation. Use New Statement to create an ordered list of FedSQL statements.

Cleanup

The optional CLEANUP statement provides a way for a user to specify SQL that will be executed when the data cache is removed. This might happen when a refresh operation has failed and the invalid cache must be removed. This would also occur when an old data cache table is removed as a result of a purge_cache operation, or when the cleanup thread times out and does an automatic cleanup. This clause is normally used when USING, EXEC BEFORE, or EXEC AFTER is in use and it generally reverses whatever was done. Multiple EXEC CLEANUP statements are allowed and statements are executed in the order specified. Use New Statement to create an ordered list of FedSQL statements.

Disabling and Enabling Cache Tables

A cache view or table can be temporarily disabled in the event that it has to be taken offline for any reason. Disabling a cache does not drop or delete a cache. The cache is just temporarily suspended while the users are rerouted to the original definer's rights view on which the cache is built. When the cache is enabled, users are directed back to the actual cache.

To disable a cache, select **Disable Cache** from the **Action** menu located in the **Cache** tab.

To enable a cache, select **Enable Cache** from the **Action** menu located in the **Cache** tab.

Deleting Cache Tables

To remove a cache, select **Delete Cache** from the **Action** menu located in the cache tab. Choosing **Disable Cache** temporarily disables the cache but does not remove it from the server.

Refreshing Cached Data

Overview

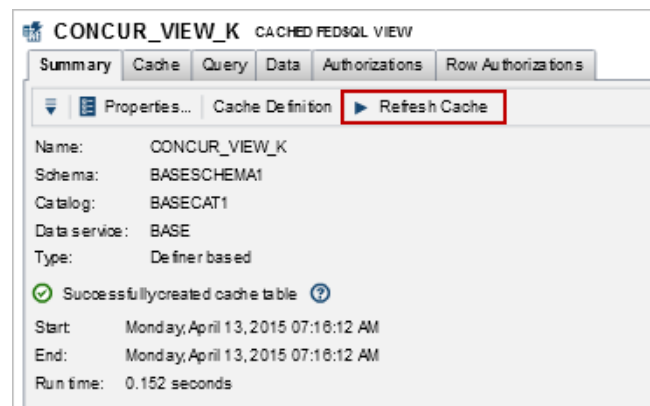
Use one of the methods below to refresh cached data:

- manual refresh
- scheduled refresh
- custom refresh

Manual Refresh

To perform a manual refresh of a data cache, use the **Refresh Cache** icon in the toolbar to refresh data in the selected cache table. This method is useful if you receive a message 'Currently using old cache' upon opening a cached view.

Figure 10.3 Manual Refresh of Cache

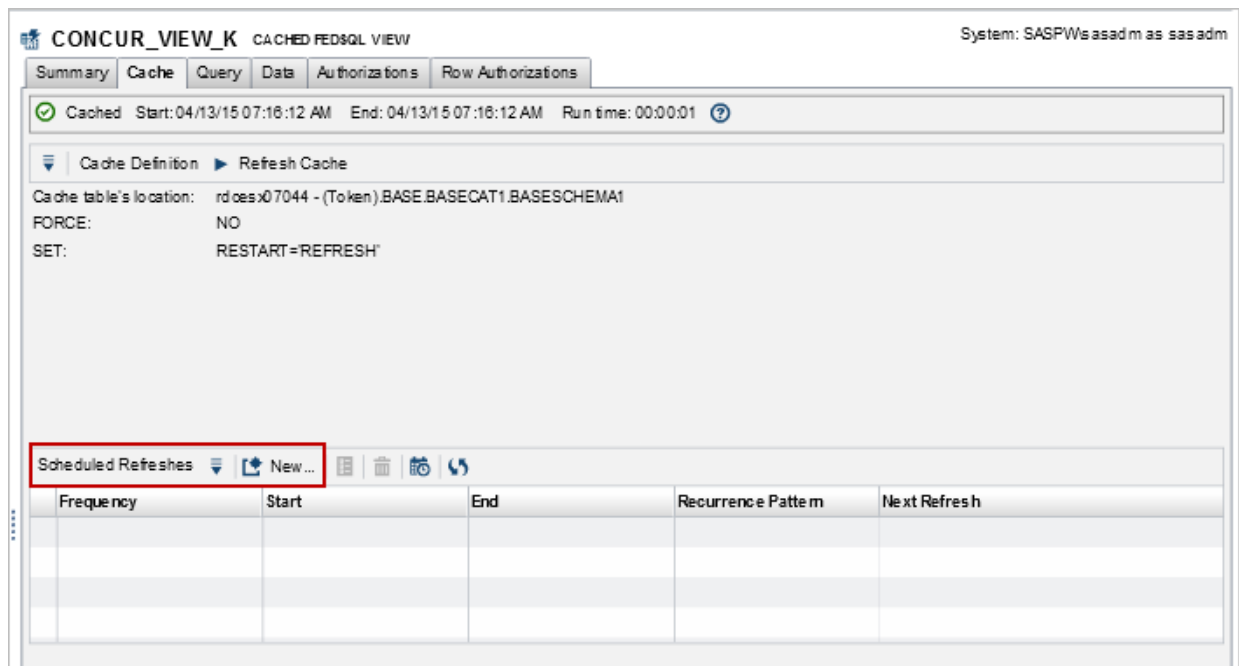


Scheduled Refresh

Configuring a Scheduled Refresh

Cache refreshes are scheduled using triggers that specify a schedule for refreshing the cache. This action can be performed as a one-time refresh or a regularly scheduled refresh. To schedule cache tables to refresh automatically, use **Scheduled Refreshes** located on the **Cache** tab at the bottom of a displayed cache.

Figure 10.4 Scheduled Refresh for Cache



To schedule a refresh of a cache table:

1. Locate the cache table in the tree and click the **Cache** tab.
2. At the bottom of the panel, at **Scheduled Refreshes**, select **New**. The Scheduled Refresh dialog box appears.
3. Specify the frequency of the job using the drop-down menu. Possible options are monthly, weekly, daily, hourly, minutes, once, or custom. The options in the Scheduled Refresh dialog box vary according to the frequency type selected.
4. To schedule an hourly refresh with a start and stop time, perform the following steps:
 - a. At frequency, select **Hourly** from the drop-down menu.
 - b. Specify the interval in hours and specify a start time and start date.
 - c. Specify an end time and date, or select **No end time and date**.
 - d. Click **OK** to schedule the job.

Note: A monthly refresh created to occur on the 'last day of the month' is saved as a custom refresh that requires **CRON** for editing.

Figure 10.5 Cache Scheduled Refresh

CRON Custom Scheduled Refresh

Overview

Using cron expressions, you can define a custom schedule to refresh cached data. A cron expression is a string comprised of at least 6 fields separated by white space. You can set any command to run continually or at set intervals by populating fields with a combination of values and special characters. Fields can contain any of the allowed values, along with various combinations of the allowed special characters for that field. Cron expression fields and values are as follows:

Position	Field Name	Required	Allowed Values	Allowed Special Characters
1	Seconds	YES	0-59	, - * /
2	Minutes	YES	0-59	, - * /
3	Hours	YES	0-23	, - * /
4	Day of month	YES	1-31	, - * ? / L W
5	Month	YES	1-12 or JAN-DEC	, - * /
6	Day of week	YES	1-7 or SUN-SAT	, - * ? / L #
7	Year	NO	empty, 1970-2099	, - * /

Here are examples of some common cron expressions and their meanings:

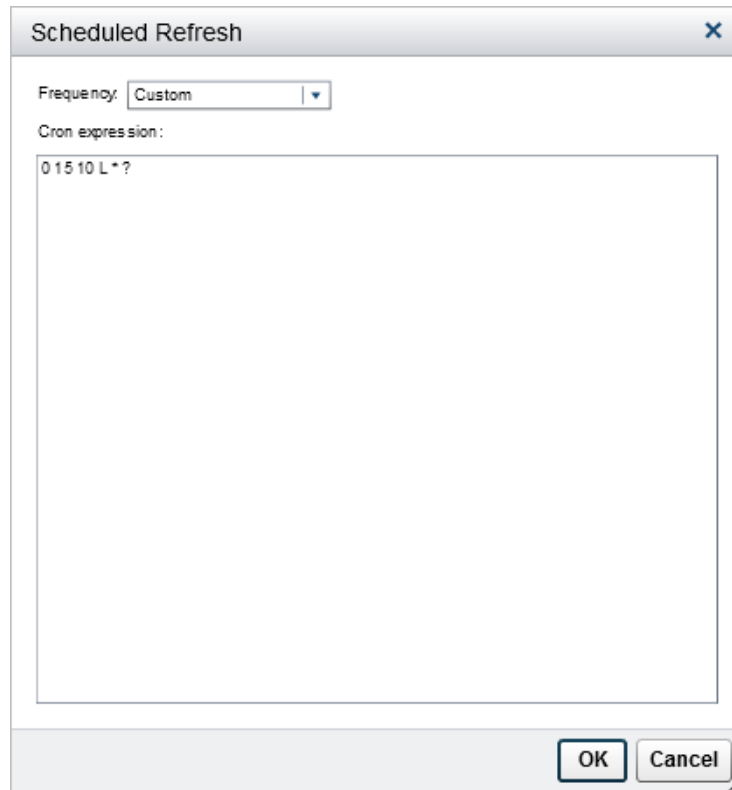
Expression	Meaning
0 0 12 * * ?	Fire at 12:00 PM (noon) every day
0 15 10 ? * *	Fire at 10:15 AM every day
0 15 10 * * ?	Fire at 10:15 AM every day
0 15 10 * * ? *	Fire at 10:15 AM every day
0 15 10 * * ? 2012	Fire at 10:15 AM every day during the year 2012
0 * 14 * * ?	Fire every minute starting at 2:00 PM and ending at 2:59 PM, every day
0 0/5 14 * * ?	Fire every 5 minutes starting at 2:00 PM and ending at 2:55 PM, every day
0 0/5 14,18 * * ?	Fire every 5 minutes starting at 2:00 PM and ending at 2:55 PM, AND fire every 5 minutes starting at 6:00 PM and ending at 6:55 PM, every day
0 0-5 14 * * ?	Fire every minute starting at 2:00 PM and ending at 2:05 PM, every day
0 10,44 14 ? 3 WED	Fire at 2:10 PM and at 2:44 PM every Wednesday in the month of March
0 15 10 ? * MON-FRI	Fire at 10:15 AM every Monday, Tuesday, Wednesday, Thursday, and Friday

For more information and additional examples, see <http://www.quartz-scheduler.org/documentation>

Configure a Custom Refresh Interval

Use the procedure below to schedule a custom refresh interval for cache.

1. Select **New** on the Scheduled Refreshes toolbar.
2. Select **Custom** from the frequency drop-down menu.
3. Enter your cron expression. The cron expression in the following example sets the schedule to fire at 10:15 am on the last day of every month:

Figure 10.6 Custom Scheduled Refresh Using Cron Expression

Scheduled Refresh

Frequency: Custom

Cron expression:

0 15 10 L * ?

OK Cancel

4. Click **OK** to schedule the job.

Working with the Schedule

Viewing Scheduled Jobs

Click **Open Schedule** on the **Federation Server Home** tab to open the schedule and view active jobs.

Figure 10.7 Home Tab – Open Schedule

Figure 10.8 Cache Refresh Jobs Schedule

SAS® Federation Server Manager - Schedule

File Help

Schedule

Job Name	Server URL
d fidvm12 (U&P).rmds.rmds.r1abc_VIEW.Refreshing Job	jdbcsastkts://fidvm12.nasas.com:2171
d fidvm12 (U&P).postgres-rdoesx07044.postgres-rdoesx07044.nlsbicaaa_VIEW.Refreshing Job	jdbcsastkts://fidvm12.nasas.com:2171
d fidvm12 (U&P).rmds.rmds.rmds_view.Refreshing Job	jdbcsastkts://fidvm12.nasas.com:2171
d fidvm12 (U&P).BASEcat.sch.CONCUR_VIEW.Refreshing Job	jdbcsastkts://fidvm12.nasas.com:2171

Scheduled Refreshes

Frequency	Start	End	Recurrence Pattern	Next Refresh
Monthly	10/14/15 03:42:00 AM	No end date	[Day 14 03:42:00 AM]	11/14/15 03:42:00 AM
Weekly	10/14/15 03:42:00 AM	No end date	Every 1 weeks	10/21/15 03:42:00 AM

From the **Schedule** tab, you can view or edit scheduled jobs, delete one or more jobs, and search for jobs.

Viewing Job History

Use the following procedure to view job history.

1. Open the **Action** menu in the Scheduled Refreshes panel
2. Select **Job Information** from the drop-down menu.
3. Select the **History** tab in the Job Information dialog box.

The History tab displays the start and end time, duration, and status of the selected job.

4. Click **Close** when you are finished viewing history.

Deleting Scheduled Jobs

Use the following procedure to delete one or more scheduled jobs:

1. Select one or more jobs in the Schedule.
2. Click **Delete** on the toolbar.

Chapter 11

Configuring Row-Level Security

Column and Row-Level Security	97
Column-Level Security	97
Row-Level Security	98
Defining User Functions for Row-Level Security	101
Overview	101
Using the Clause Builder	101
Working with User Functions	102

Column and Row-Level Security

Column-Level Security

About Column-Level Security

Column security in SAS Federation Server Manager is invoked by expanding the columns associated with a selected table. A new view opens that displays all of the columns. Each column is accompanied by two tabs that outline the details of the column, the **Summary** tab and the **Authorizations** tab.

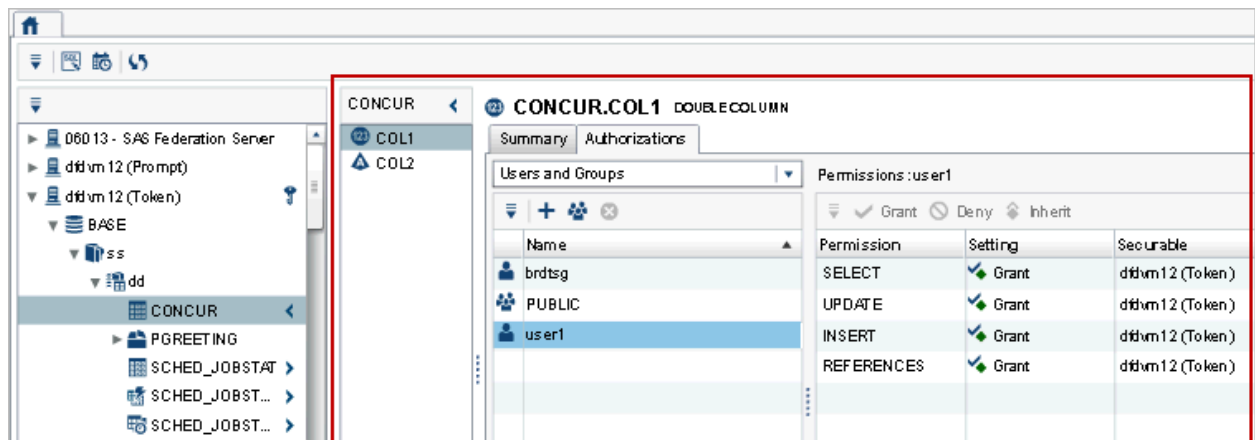
Summary tab

The **Summary** tab displays a summary of the column that shows the identification and details of the column.

Authorizations tab

The **Authorizations** tab displays user and group identities with associated permissions.

Figure 11.1 Column Level Security



Granting Security on a Column

Use the following procedure to grant security for a column:

1. Log on to a SAS Federation Server and navigate to a table by selecting a **data service** ⇒ **catalog** ⇒ **schema**.
2. Expand the table's columns by clicking the arrow ► to the right of the table.
3. Under the **Authorizations** tab, select a column and select a user. You can select multiple users by holding the shift key while selecting user objects.
4. Select a user permission, or multiple permissions, that apply to the column object.
5. Select **Grant** or **Deny** to assign column permissions to the selected user objects.
6. Click **OK**.

Row-Level Security

About Row-Level Security

You can invoke row-level security in SAS Federation Server Manager by expanding a table and defining filters for rows associated with the table. Use the **Authorizations** and **Row Authorizations** tabs to configure row-level security.

Authorizations

The **Authorizations** tab displays user and group identities with associated permissions. This tab also displays permission row filters for users as they are selected.

Row Authorizations

The **Row Authorizations** tab displays all of the row filters that currently apply to the table. Mixed values are shown if different filters are assigned to the selections. The Edit Filter and Export Action menu items are disabled for items marked mixed value. You can also delete filters from this tab. Row filters are available only on tables. When you create a row filter, the new filter replaces the previous filter, only if the filter is applied to the same user or group. Otherwise, you can apply multiple row-level filters to a table for different groups and users.

When working with row filters, the following special conditions might apply if OK is selected when the WHERE clause does not contain text or an otherwise invalid filter.

- OK is equivalent to Cancel if a new filter is being defined. No filter is applied, and Select permissions for selected users or groups are not automatically set to Grant if that had been necessary to apply a filter.
- If an existing filter is being edited, selecting OK is equivalent to deleting the filter. The existing filter is removed, and the user is prompted for setting the Select permissions for the selected users or groups.

Applying Row Filters

Use the procedure below to create and apply row filters to a table.

1. Log on to a federation server object and navigate to a table by selecting **data service** ⇒ **catalog** ⇒ **schema** ⇒ **table**.
2. Select the **Data** tab and click **View Data** to view data for the selected table.
3. Select the **Row Authorizations** tab to create a new filter.
4. Select **New Filter** to open the Filter dialog box.
5. On the Filter dialog box, you can enter a WHERE statement manually, or select the **Clause Builder** or **User Function** to build the WHERE statement.

Note: See “[Working with User Functions](#)” for additional information.


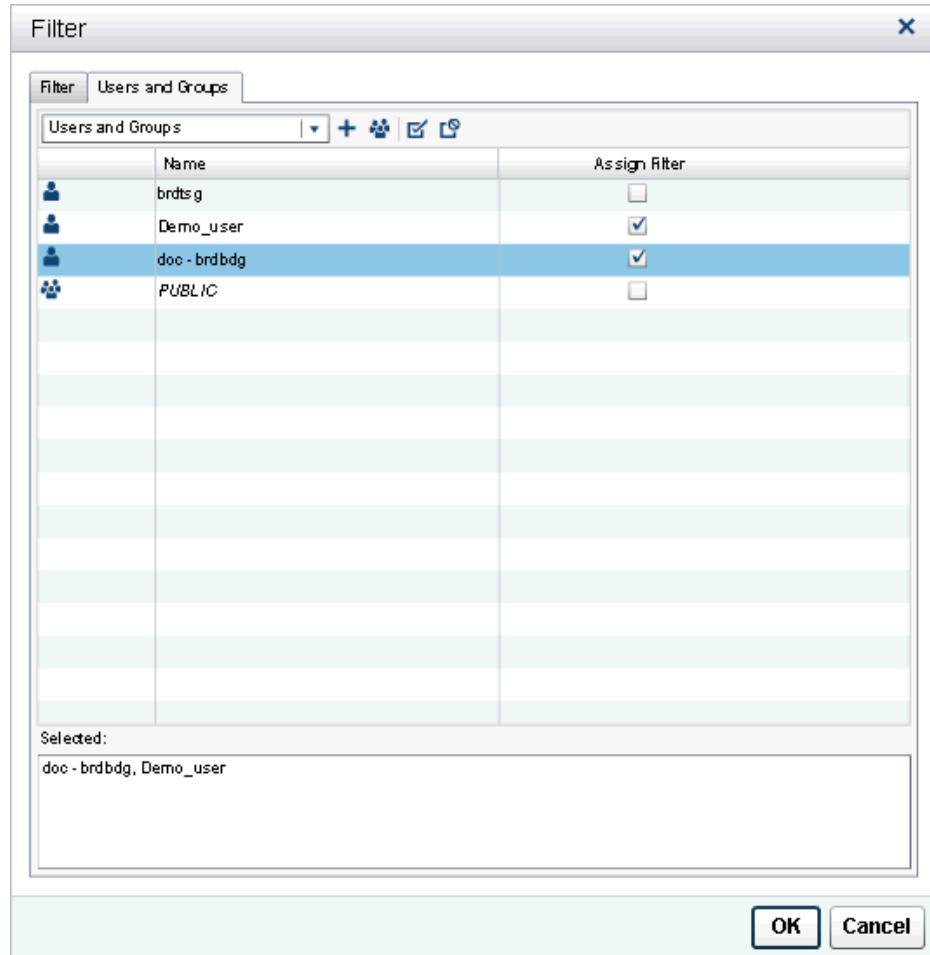
6. Select **New Condition** to add additional rows to the WHERE statement.
7. Select the **Users and Groups** tab and perform the following actions:
 - Select add Users and Groups  to add additional user or group objects to the filter.
 - Select the **Assign Filter** check box for each user affected by the filter.

Figure 11.2 Assigning Filters

8. Click **OK** when you are finished assigning filters.

Editing a Row Filter

Use the Filter Properties dialog box to edit an existing filter.

1. Select the **All Row Filters** tab for the table.
2. Select **Edit Filter Properties**. The Filter Properties dialog box is displayed, listing the existing filters for the table.
3. Use the **Clause Builder** or **User Function** to edit a filter and click **OK**.

Note: If you select a filter and there is an existing filter on the selected user or group, a warning is displayed, indicating that selecting the new filter(s) for these users or groups result in existing filter(s) being overwritten and deleted.

Removing a Filter

Use this procedure to remove a filter.

1. Open the **All Row Filters** tab and click **Delete Filter**.
2. A warning message appears that prompts you to set a new permission for the user or group. Select a new permission and click **OK**.

Note: When selecting OK and the WHERE clause does not contain text or an otherwise invalid filter, the following conditions apply:

- OK is equivalent to Cancel if a new filter is being defined. No filter is applied, and Select permissions for selected users or groups are not automatically set to Grant if that had been necessary to apply a filter.
- If an existing filter is edited, OK is equivalent to deleting the filter. The existing filter is removed, and the user is prompted for setting the Select permissions for the selected users or groups.

Defining User Functions for Row–Level Security

Overview

You can use the “RLS Library and Library Reference” to identify the parameters required for each user function and a description of the value(s) returned by each user function. You can access the User Function dialog boxes directly from the toolbar in the Filter dialog box or from the Clause Builder.

Using the Clause Builder

Using the clause builder, you can build any type of condition from a simple WHERE statement to a more complex user function using RLS functions. The clause builder is accessible while creating a row filter or editing an existing row filter. Use the following procedure to access the Clause Builder and set up filters.

1. Select the **Data** tab and click **View Data** to view data for the selected table.
2. Select the **Row Authorizations** tab.
3. Select **New Filter** to open the Filter dialog box.
4. Using the **Action** list menu, select the **Clause Builder** or **User Function** to build the WHERE statement.
5. Use the drop-down menu under **Column** to select a field from the source table.
6. Use the **Operator** list menu to select a qualifier.
7. In the **Value** field, enter information to qualify the statement. You can also use the list menu to select **Use user function**. See “[About User Functions](#)” for more information.
8. Click **OK** when you are finished.

Figure 11.3 Row-level Security Clause Builder

Clause Builder

+ New Condition

Column: install_failures Operator: = Equals Value: 0

AND

Column: major_version Operator: = Equals Value: 90

OK Cancel

Working with User Functions

About User Functions

SAS Federation Server provides various functions that return information about a user that is currently connected to a data source. These functions are referred to in SAS Federation Server Manager as User Functions. Since the results returned from each function contain information about the current user, User Functions can be valuable when constructing filters for row-level security. For example, if rows in a table should be returned based on the authentication domain of the currently connected user, you could use the DOMAIN() user function to accomplish this. Use the RLS Library and Library Reference to identify the values and formatting associated with each user function.

Insert User Function from Filter

You can select **Insert User Function** from the toolbar while building a WHERE statement. User function is also accessible through the clause builder.

Note: RLS filters with user functions are most valuable when the filter is assigned to one or more groups.

1. Select the User Function icon from the filter dialog box.
2. Use the drop-down menu in the User Function dialog box to select a user function.
3. Using the RLS Library, choose a user function that returns a value that can be compared to values stored in a column from your source table.

User Function Example

Consider a table with a Username column that contains the name of the user that is permitted to view the corresponding row. The User Function that returns the name of the current user is current_user. Creating an RLS filter that compares the contents of the column Username with the value returned by the function current_user shows users the

rows where the Username column contains the name of the current user. This filter can be associated with many users, and it allows each user to view the filtered rows, without having to write a separate filter for each user.

Insert User Function from Clause Builder

Use the following procedure to specify conditions using the Clause Builder.

1. Select **Value** and select **Use user function**. The User Function dialog box appears
2. Use the drop-down menu to select a user function.
3. You can also specify a reference table to look up a user function by selecting Use user function to look up the value in a reference table.
4. Click **OK** when you are finished.

User Function Value from a Reference Table.

The **Use user function to look up the value in a reference table** option is helpful when a table or view exists that associates the value returned from a user function with the values stored in a column of the table to which RLS is to be applied. Here is an example:

You have a table, SALES_LEADS, containing a list of sales leads for your company. Each member of your organization's sales department is responsible for all the leads in one or more regions. Each region has an associated identification code that is stored in the SALES_LEADS table in the REGION_ID column. Members of the sales department should be able to select only rows from SALES_LEADS that correspond with the regions with which the member is associated.

LEAD_NAME	LEAD_ADDRESS	LEAD_PHONE	...	REGION_ID
Lead #1	S
Lead #2	NE
Lead #3	NE
Lead #4	W

Using a reference table that associates the REGION_ID column with each sales department members' user name allows for flexible application of row-level security on the SALES_LEADS table.

REGION_ID	USER_NAME
S	Executive One
W	Executive One
NE	Executive Two

If Executive One changes regions, records need to be updated only in the SALES_LEADS_LOOKUP table in order to maintain appropriate row-level security on the SALES_LEADS table. A link between the reference table (SALES_LEADS_LOOKUP) and the target table (SALES_LEADS), such that RLS will

update as the reference table updates, can be created by selecting the Use user function to look up the value in a reference table check box, and performing the following task.

If the user function is set to look up a value in a reference table, specify the source reference table and the columns to filter:

1. In the reference table field, use the browse button to navigate to the reference table or view. The Select Table dialog box appears and presents a list of data services. Drill down to the source table and select it. In the example, the reference table is SALES_LEADS_LOOKUP.
2. In the Column that the user function value appears in field, specify the column in the reference table that contains values corresponding to those returned by the user function. In the example, the column in the reference table (SALES_LEADS_LOOKUP) that matches the result of the current_user function is USER_NAME.
3. In the Column that the value to filter on appears in field, specify the column in the reference table that contains values mapped to the results of the user function, which will be used to filter the rows in the table to which RLS is being applied. In the example, the column in the reference table (SALES_LEADS_LOOKUP) that contains values to filter on in the target table (SALES_LEADS) is REGION_ID.
4. Click **OK**. The information is set in the Value field of the Clause Builder.
5. Click **OK** to add the values to the row filter.

Part 5

Advanced Topics

<i>Chapter 12</i>	
Working with the Console	107
<i>Chapter 13</i>	
Working with DS2 Dialect	111
<i>Chapter 14</i>	
Data Quality and Cleansing Functions	113
<i>Chapter 15</i>	
Data Masking	123
<i>Chapter 16</i>	
SQL Logging	131

Chapter 12

Working with the Console

SAS Federation Server Manager Console	107
About the Console	107
Working with the Console	107
Dialects	108
Working with Information Views	109

SAS Federation Server Manager Console

About the Console

The SAS Federation Server Manager Console supports FedSQL and DS2 languages. The Console enables administrators to manage SAS Federation Server without navigating all of the user interfaces that SAS Federation Server Manager provides. You can connect to the Console using any DSN that you have created and submit SQL and DDL statements. The type of SQL and DDL statements executed depends on the DSN connection and what you are trying to accomplish. For example, you can use FedSQL information views to call specific information from the SAS Federation Server database. You can also use the Console to execute administration DDL to configure SAS Federation Server objects, and privileges. For a complete list of administration DDL, see the *SAS Federation Server Administrator's Guide*.

Working with the Console

Launching Console

To launch the Console select, select the **Action** menu on the home tab and select **Open Console** from the drop-down menu. You can also use the SQL icon displayed on the toolbar of the **SAS Federation Server** home tab.

Selecting a Federation Server

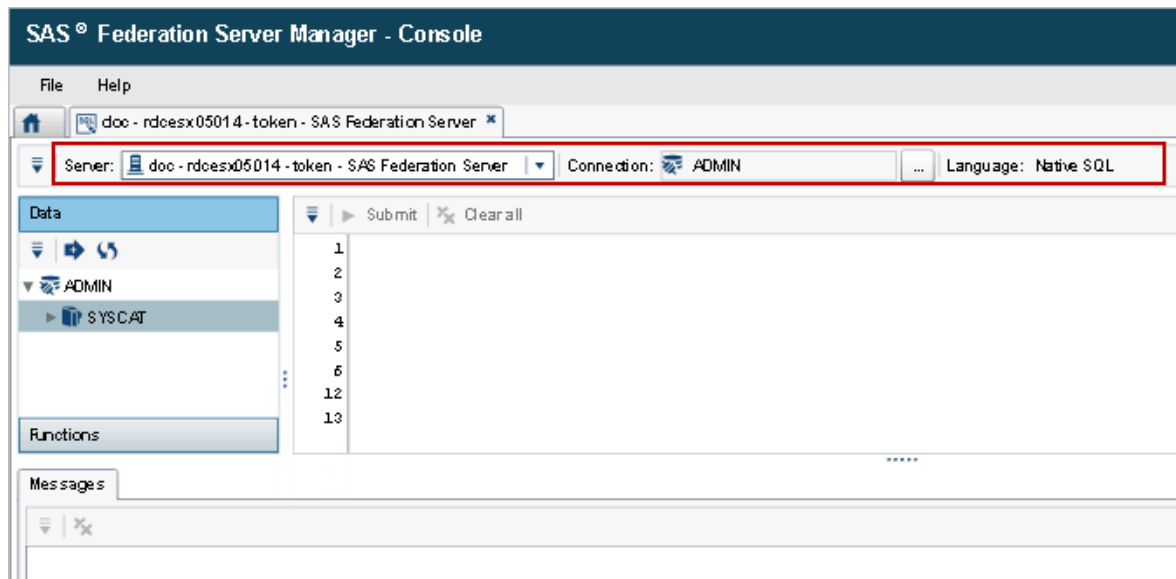
Use the **Server** list menu to select a federation server. You might be prompted to log on to the server if you have not already done so. After establishing connection to the server, the DSNs associated with the selected server will load for the session. The ADMIN DSN is the default connection.

Changing the Connection

You can change the connection by selecting one or more DSNs under **Connection**.

1. Select **Browse** to open the Select DSN(s) dialog box.
2. At the Select Data Source Names dialog box, check each DSN that you want to use.
3. If you are using more than one DSN, use the drop-down menu at **Select connection language** to choose a language for the session. Depending on the DSN, you can choose FedSQL or DS2.
4. Select **OK** when you are finished.

Figure 12.1 SAS Federation Server Manager Console



Using More Than One DSN

If you are using more than one DSN for your console connection, you are creating a federated DSN for the session. When you choose multiple DSNs to include in the federated DSN, any dialect is allowed. However, when you set the dialect on the parent (federated) DSN, the dialects that are associated with the child DSNs are ignored. Because you are using data federation, you can no longer use native SQL for the dialect. The valid options are FedSQL and DS2.

Dialects

When using the Console, the DSN connection controls the dialect that is available for the session. The following table outlines possible DSN and dialect combinations:

Selected DSN	Dialect Type	Driver Type	SQL Functions Displayed
ADMIN DSN	FedSQL	SYSCAT	FedSQL
Standard DSN	FedSQL, DS2, native	ODBC, vendor specific	FedSQL or native, if driver type=ODBC also shows ODBC functions.

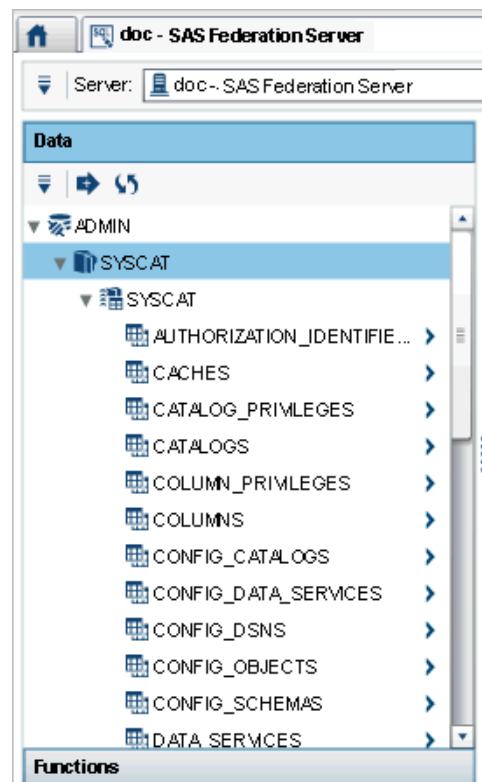
Selected DSN	Dialect Type	Driver Type	SQL Functions Displayed
Federated DSN	FedSQL	ODBC, vendor specific	FedSQL; Federated DSNs always use FedSQL dialect
Multiple DSNs	Various	ODBC, vendor specific	FedSQL, if driver type=ODBC also shows ODBC functions.

Working with Information Views

Using the console, you can access any information view for SAS Federation Server. You must connect to your server with the ADMIN DSN to use the information views. To display a list of information views, expand the ADMIN data service, SYSCAT catalog, and schema in the tree. You can also issue DDL statements against one or more information views.

1. Connect to the Console with the ADMIN DSN.
2. In the left panel, under Data, click to expand **ADMIN data service** ⇒ **SYSCAT catalog** ⇒ **SYSCAT schema** to expose the information views.

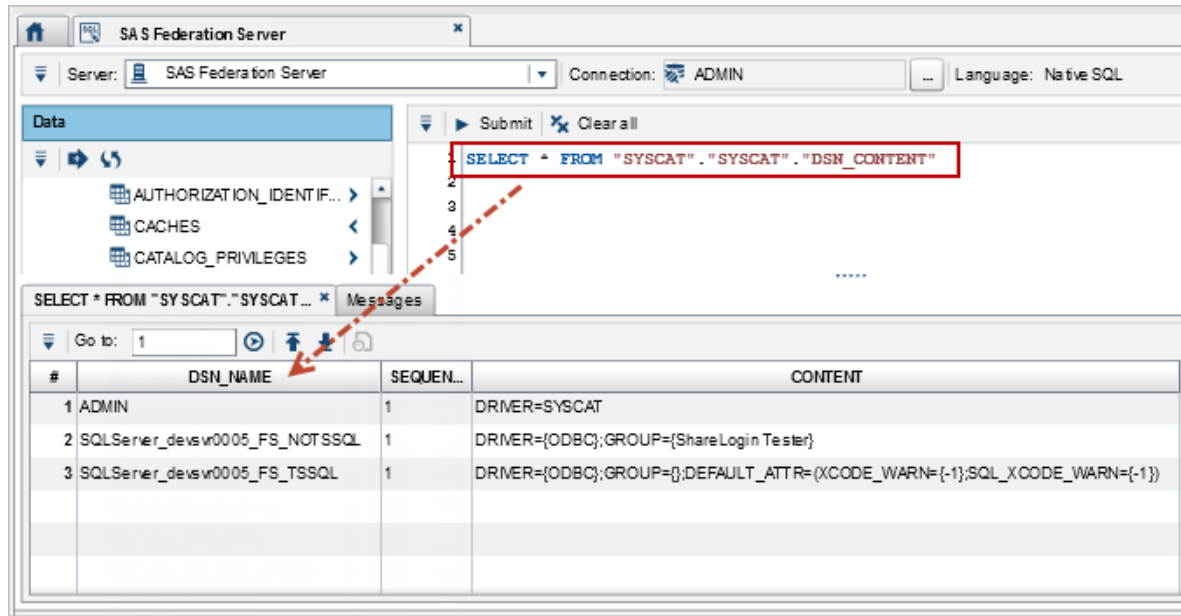
Figure 12.2 SAS Federation Server Information Views



The following example returns a list of DSNs for a federation server object.

1. Log on to a federation server and open the Console.
2. Select the server that you want to query and select the **ADMIN** connection.
3. Enter the following statement in the Console: **SELECT * FROM "SYSCAT"."SYSCAT"."DSN_CONTENT"**.
4. Click **Submit**. A list of DSNs is returned at the bottom of the display:

Figure 12.3 DSN_CONTENT Information View



For additional information, including visibility rules that apply to information views, see the appendix for "Information Views" in the *SAS Federation Server Administrator's Guide*.

Chapter 13

Working with DS2 Dialect

Overview of DS2 on SAS Federation Server	111
Create a DS2 DSN	111
DSN and DS2 Object Permissions	112

Overview of DS2 on SAS Federation Server

DS2 is a SAS proprietary programming language that is used for advanced data manipulation. DS2 provides capabilities not available through SQL, such as scoring models. In addition, you can use DS2 code to run data quality functions on SAS Federation Server. DS2 is included with Base SAS and intersects with the SAS DATA step.

To invoke DS2, you must configure a DSN that uses the DS2 dialect and grant users CONNECT permission to the DSN. In addition, users must have EXECUTE permissions on DS2 objects, such as packages and threads, before any functions can be run against them.


DS2 objects inherit privileges from the server in the following order:

- SERVER
- (DATA) SERVICE
- CATALOG
- SCHEMA
- PACKAGE
- FUNCTION

To view the contents of DS2 programs, use the DESCRIBE PACKAGE or DESCRIBE THREAD commands. For more information about DS2, see the *SAS DS2 Language Reference*.

Create a DS2 DSN


To create a DSN that uses the DS2 dialect:

1. Select a federation server object in the tree and click the **Data Source Names** tab.
2. Click **New Standard Data Source Name**  and enter a name and description for the data source. Click **Next** to continue.
3. Select a data service that is affiliated with the DSN and click **Next**.
4. Select the schemas to make available through the DSN and click **Next** to continue.
5. Security should be enabled by default. Click **Next**.
6. Select **DS2** as the language to use when communicating with this DSN. Click **Next**.
7. Enter additional options and click **Next**.
8. Review the summary and click **Finish**.

DSN and DS2 Object Permissions

Assign the CONNECT privilege to all users that will be using DS2. See [Connect Permissions for DSN](#) for the tasks.

In addition, users require the EXECUTE permission the DS2 objects. To assign permissions for a DS2 object:

1. Select a DS2 object in the tree and click the associated **Authorizations** tab.
2. Under Identities select .
3. Enter a user or group name in the search box and click **Search**. To return a list of available user names and groups, leave the search box empty and click **Search**.

Note: The Add Users and Groups dialog box does not automatically populate with data when it is launched. Use the Search feature to return user names and group names.

4. Select the user (or users) and click **Add**. The user object is added to the **Identities** list.
5. Select the user from the **Identities** list, and set permissions by selecting **EXECUTE Permission** list.

Note: Any permissions that are currently assigned to a user object are dimmed on the Permissions toolbar and cannot be selected.

6. Select **Grant** to assign permissions.

Note: See “[Granting Permissions for Federation Server and Associated Objects](#)” for additional information about object permissions.

Chapter 14

Data Quality and Cleansing Functions

About Data Quality on SAS Federation Server	113
Overview	113
Standardization	114
Matching	115
Pattern Analysis	116
Identification	116
Gender	117
Case	117
Parse	118
Extraction	119
Data Quality Functions in SAS Federation Server Manager	120
Navigation	120
Working with Data Quality Methods	120
Updating the QKB	122

About Data Quality on SAS Federation Server

Overview

Data Quality on SAS Federation Server is implemented through SAS Quality Knowledge Base (QKB) using FedSQL and DS2. The data quality methods use data quality rules from the SAS QKB to cleanse data. The rules, referred to as QKB definitions, are operation- and locale-specific. The data quality functions are exposed through a Memory Data Store (MDS) table with a reserved namespace, SYSPROC.DQ.

Use a SELECT statement to invoke each data quality method. Ensure that locale is always UPPER cased as shown in the following example:

```
SELECT SYSPROC.DQ.DQUALITY.DQSTANDARDIZE (
    state,
    'State/Province (Full Name)',
    'ENUSA' ) AS STANDARD_STATE
FROM employee
```

You can also use fully qualified column names in the SELECT statement as shown in the following example:

```
SELECT SYSPROC.DQ.DQUALITY.DQSTANDARDIZE (
    "HR"."PAYROLL"."employee"."state",
    'State/Province (Full Name)',
```

```
'ENUSA' ) AS STANDARD_STATE
FROM employee
```

TIP When preparing your SQL statement, enclose SQL literals in single quotation marks. SQL literals are any numeric, character, string, date, or Boolean values that are not identifiers. This also includes literal arguments in FedSQL functions. Use double quotation marks when specifying identifiers such as catalog, schema, table, or column parts. Although not required, use double quotation marks if SQL identifier case sensitivity is an issue.

The following topics describe each data quality method that is delivered with SAS Federation Server.

Standardization

DQSTANDARDIZE

Standardization generates a preferred standard representation of data values. Standardization definitions are provided for character content such as dates, names, and postal codes. The available standardization definitions vary from one locale to the next. The return values are provided in the appropriate case, and insignificant blank spaces and punctuation are removed. The order of the elements in the return values might differ from the order of the elements in the input character values.

The value parameter for the **DQSTANDARDIZE** method accepts the following data types:

```
nvarchar(256) | date | timestamp
```

Here are sample SELECT statements for standardization:

```
SELECT SYSPROC.DQ.DQUALITY.DQSTANDARDIZE (
    state,
    'State/Province (Full Name)',
    'ENUSA' ) AS STANDARD_STATE
FROM employee
```

Table 14.1 Results for Standard State

State	STANDARD_STATE
NC	North Carolina

```
SELECT SYSPROC.DQ.DQUALITY.DQSTANDARDIZE (
    "HR"."PAYROLL"."employee"."postalCode",
    'Postal Code',
    'ENUSA' ) AS STANDARD_POSTAL_CODE
FROM employee
```

Table 14.2 Results for Standard Postal Code

Postal Code	STANDARD_POSTAL_CODE
275130250	27513-0250

Matching

DQMATCH

Matching analyzes the input data and generates a matchcode for the data. The matchcode represents a condensed version of the character value. Similar strings receive identical matchcodes. You can specify a sensitivity value, ranging from 0–100, indicating the degree of similarity that should be applied to consider something a match. A sensitivity value of 100 yields more information, and 0 yields less. The default recommended sensitivity value is 85.

The value parameter for the **DQMATCH** method accepts the following data types:

`nvarchar(256) | date | timestamp`

Here are sample SELECT statements for matching:

```
SELECT SYSPROC.DQ.DQUALITY.DQMATCH (
    postalCode,
    'Postal Code', 85,
    'ENUSA' ) AS MATCH_POSTAL_CODE
FROM employee
```

Table 14.3 Results for Match Postal Code

Postal Code	MATCH_POSTAL_CODE
275130250	ABC~\$\$\$\$
27513	ABC~\$\$\$\$
27540	DEF~&&&&

```
SELECT SYSPROC.DQ.DQUALITY.DQMATCH (
    "HR"."PAYROLL"."employee"."phone",
    'Phone', 50,
    'ENUSA' ) AS MATCH_PHONE
FROM employee
```

Table 14.4 Results for Match Telephone Number

Telephone Number	MATCH_PHONE
9195551212	ABC~\$\$\$\$
5551212	ABC~\$\$\$\$
1-9195551212	ABC~\$\$\$\$
202-555-0143	XYZ~\$\$\$\$
5550143	XYZ~\$\$\$\$

Pattern Analysis

DQPATTERN

Pattern analysis returns a simple representation of a character pattern based on a text string, which can be used for pattern frequency analysis in profiling jobs. Pattern analysis identifies words or characters in the input data column as numeric, alphabetic, non-alphanumeric, or mixed. The choice of pattern analysis definition determines the nature of the analysis:

* non-alphanumeric, such as punctuation marks or symbols

A alphabetic

M mixture of alphabetic, numeric, and non-alphanumeric

N numeric

Here is a sample SELECT statement for pattern analysis:

```
SELECT SYSPROC.DQ.DQUALITY.DQPATTERN (
  State,
  'Word',
  'ENUSA' ) AS PATTERN_WORD
FROM employee
```

Table 14.5 Results for Pattern Analysis

Input	Word_Pattern
North Carolina	AA
Virginia	A
SC	A

Identification

DQIDENTIFY

Identification analysis returns a value that indicates the category of the content in an input character string. The available categories and return values depend on your choice of identification definition and locale.

Here are sample SELECT statements for identification analysis:

```
SELECT SYSPROC.DQ.DQUALITY.DQIDENTIFY (
  Name,
  'Field Name',
  'ENUSA' ) AS IDENTIFY_FIELD_NAME
FROM employee
```

```
SELECT SYSPROC.DQ.DQUALITY.DQIDENTIFY (
  email,
  'E-mail (Country Identification)',
  'ENUSA' ) AS IDENTIFY_EMAIL
```

```
FROM employee
```

Table 14.6 Results for Identification

EMAIL	COUNTRY
P.Adams@mymail.ca	Canada
Joe.King@bleep.au.com	Australia
joe.smith@trip.com.us	United States
xxoo@internet.es	Spain

Gender

DQGENDER

Gender analysis evaluates the name or other information about an individual to determine the gender of that individual. If the evaluation finds substantial clues that indicate gender, the function returns a value that indicates that the gender is female or male. If the evaluation is inconclusive, the stored procedure returns a value that indicates that the gender is unknown. The exact return value is determined by the specified gender analysis definition and locale.

Here is a sample SELECT statement for gender analysis:

```
SELECT SYSPROC.DQ.DQUALITY.DQGENDER (
    NAME,
    'Name' ,
    'ENUSA' ) AS GENDER_NAME
FROM employee
```

Table 14.7 Gender Analysis

Name	Gender
Jane Smith	F
Joe King	M
S. Adams	U

Case

Use case definitions to apply uppercase and lowercase lettering using context-sensitive rules. Case operates on character content, such as names, organizations, and addresses. You can specify one of three casing types: uppercase, lowercase, or propercase. When uppercase or lowercase is specified, the function applies Unicode uppercase or lowercase mappings to the characters in the input string. When propercasing is specified,

the function applies uppercase mappings to the first letter in each word and lowercase mappings to the remaining letters.

DQLOWERCASE

Here is a sample SELECT statement for lower casing:

```
SELECT SYSPROC.DQ.DQUALITY.DQLOWERCASE (
    name,
    'Lower',
    'ENUSA' ) AS LOWERCASE_PHONE
FROM employee
```

DQUPPERCASE

Here is a sample SELECT statement for upper casing:

```
SELECT SYSPROC.DQ.DQUALITY.DQUPPERCASE (
    name,
    'Upper',
    'ENUSA' ) AS UPPERCASE_PHONE
FROM employee
```

DQPROPERCASE

Here is a sample SELECT statement for proper casing:

```
SELECT SYSPROC.DQ.DQUALITY.DQPROPERCASE (
    name,
    'Proper (Name)',
    'ENUSA' ) AS PROPERCASE_NAME
FROM employee
```

Table 14.8 Results for Casing

Name	DQ Method	Results
Jane Smith	DQLOWERCASE	jane smith
	DQUPPERCASE	JANE SMITH
	DQPROPERCASE	Jane Smith

Parse

DQPARSE

Parsing segments a string into semantically atomic tokens.

The value parameter for the **DQPARSE** method accepts the following data types:

```
nvarchar(256) | date | timestamp
```

Here is a sample SELECT statement using DQPARSE:

```
SELECT SYSPROC.DQ.DQUALITY.DQPARSE (
    address,
    'Address', 'Street Name',
    'ENUSA' ) AS PARSE_ADDRESS_STREET_NAME
```

```

FROM employee

SELECT SYSPROC.DQ.DQUALITY.DQPARSE (
    name,
    'Name (Global)', 'Prefix',
    'ENUSA' ) AS PARSE_NAME_PREFIX
FROM employee

```

Table 14.9 Results for Parsing

Name	Prefix
Ms. Jane Smith	Ms.
Mr. Joe King	Mr.
Mrs. Mary Moffet	Mrs.

Extraction

DQEXTRACT

Extraction definitions are used to extract specific entities or attributes from a text string. Extraction returns one or more extracted text values, or tokens, as output. For example, to extract a name prefix:

```

SELECT SYSPROC.DQ.DQUALITY.DQEXTRACT (
    EXTRACT_COLUMN, 'NAME', 'NAME PREFIX',
    'ENUSA')
FROM employee

```

To extract a name prefix with a given name, enter two function calls in your select statement:

```

SELECT SYSPROC.DQ.DQUALITY.DQEXTRACT (
    EXTRACT_COLUMN, 'NAME', 'NAME PREFIX',
    'ENUSA')

SYSPROC.DQ.DQUALITY.DQEXTRACT (
    EXTRACT_COLUMN, 'NAME', 'GIVEN NAME',
    'ENUSA')
FROM employee

```

Table 14.10 Results for Name Extraction

Name	Output	
Mr. James W. Church	PREFIX	Mr.
	GIVEN	James

Data Quality Functions in SAS Federation Server Manager

Navigation

The data quality methods are installed with your SAS Federation Server. The data quality package and methods are displayed in the tree under a SYSPROC data service that is associated with your federation server object. Here is the hierarchy:

- SAS Federation Server**
- SYSPROC data service**
- SYSPROC catalog**
- DQ Schema**
- DQUALITY package**
- Data quality methods**

Working with Data Quality Methods

Using the Console

Use the following procedure to launch the console and work with the data quality methods:


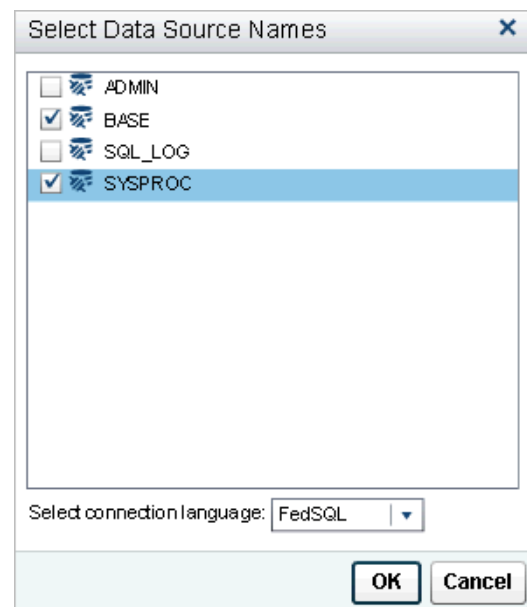
1. Select **Open Console**  from the toolbar on the **Home** tab.
2. Using the drop-down menus on the toolbar, select a **federation server**, and a connection for your data and include the **SYSPROC** data service.

Figure 14.1 Select Data Source Names for Data Quality



- At the bottom of the Select Data Source Names dialog box, use the drop-down menu to select a language (for example, **FedSQL**), and click **OK**.
- Using the **Data** riser in the tree, select **SYSPROC**.
- Expand the following objects to expose the data quality methods: **SYSPROC** ⇒ **SYSPROC** ⇒ **DQ** ⇒ **DQUALITY**.
- Enter your SELECT statement and select a data quality method (for example, **DQSTANDARDIZE**) to move it into the editor :

```
SELECT SYSPROC.DQ.DQUALITY.DQSTANDARDIZE
```

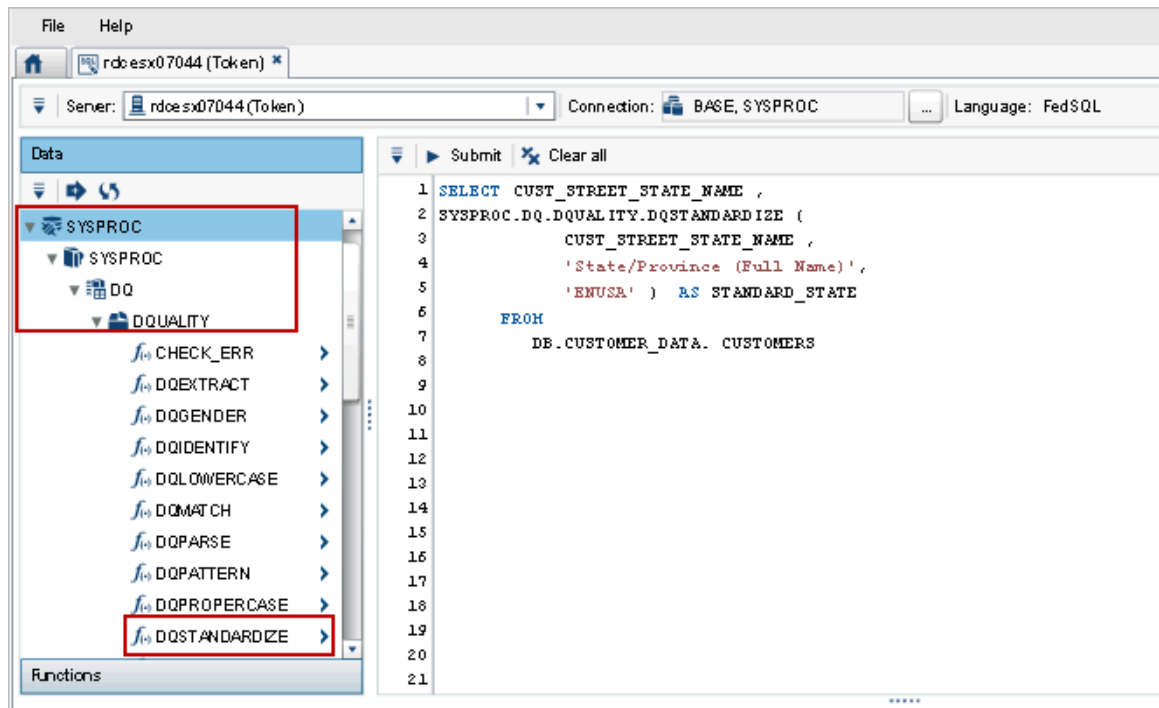
Note: The SET and GET pairs displayed in the tree are generated from attributes used with a DS2 package.

- Append the statement with custom SQL. The following example standardizes state names:

```
SELECT CUST_STREET_STATE_NAME ,
       SYSPROC.DQ.DQUALITY.DQSTANDARDIZE (
         CUST_STREET_STATE_NAME ,
         'State/Province (Full Name)',
         'ENUSA' ) AS STANDARD_STATE
FROM
  DB.CUSTOMER_DATA.CUSTOMERS
```

Note: Use UPPER case when specifying a locale.

Figure 14.2 Using Data Quality Methods – DQSTANDARDIZE



- Click **Submit** to send your code to the server. The results should look similar to the following:

Table 14.11 Results for Standard State

CUST_STREET_STATE_NAME	STANDARD_STATE
NC	North Carolina
VA	Virginia
FL	Florida

Updating the QKB

SAS provides regular updates to the QKB. It is recommended that you update your QKB each time a new one is released. For a listing of the latest enhancements to the QKB, refer to “What’s New in SAS Quality Knowledge Base.” The What’s New document is available on the SAS Quality Knowledge Base (QKB) product documentation page at support.sas.com. Either search on the product name or locate it in the product index. The updated software is available through the [Downloads](#) site.

Chapter 15

Data Masking

Overview	123
Displaying Data Masking Functions in the Console	125
Data Masking in a FedSQL View	125
Data Masking Examples	126
About the Examples	126
Encrypt	127
HASH	128
Transliterated Value (TRANC)	128

Overview

Data Masking in SAS Federation Server is a series of FedSQL functions that are accessed through the Console using the SYSCAT.DM.MASK function in a SELECT statement. Here is a brief description of each of the data masking functions.

ENCRYPT

Encrypt masks the values in a column by encrypting a single value using symmetric key encryption. Encrypted values cannot be decrypted if a KEY argument is not specified and the ENCRYPT_KEY package configuration option is not set.

```
SYSCAT.DM.mask('ENCRYPT', "value"
/*[,
'ALG', 'AES/FIPS|AES|SAS002|BASE64|SAS004|SAS003|SAS001',
'KEY', 'encrypt_key',
'DETERMINISTIC', YES|TRUE|ON|1|NO|FALSE|OFF|0,
'EXPAND_PREC', YES|TRUE|ON|1|NO|FALSE|OFF|0,
'CASE', 'U|L',
'STRIP', 'BLANK|UNICODESP|UNICODESPACE|ANY|ALL|WS'
]*/ )
```

Note: The encrypt function preserves the data type of the original column if the data type is that of character (for example, char, nchar, varchar). If the column is not a character data type, the output produces a binary data result.

DECRYPT

Decrypt unmask the values in a column by decrypting a previously encrypted value using symmetric key encryption. The DECRYPT rule returns NULL if a KEY

argument is not specified and the ENCRYPT_KEY package configuration option is not set.

```
SYSCAT.DM.mask('DECRYPT', "value"
/*[, 'ALG', 'AES/FIPS|AES|SAS002|BASE64|SAS004|SAS003|SAS001',]*/ )
```

HASH

HASH masks the values in a column by hashing a single value into a fixed-length hash digest or HMAC string. HASH is not reversible.

```
SYSCAT.DM.mask('HASH', "value"
/*[, 'ALG', 'MD5|SHA256',
'CASE', 'U|L',
'KEY', 'encrypt_key']*/ )
```

TRANC

TRANC masks the values in a column by transliterating characters from an input string to characters in an output string. Ensure that the mapped result is “lossy” (many instances of mapping multiple input character values to a single output character value) to prevent inference of the original value.

```
SYSCAT.DM.mask('TRANC',
/*Expression*/
)
```

RANDOM

RANDOM masks the values in a numeric column by replacing them uniformly distributed pseudo-random numbers. RANDOM is not reversible.

```
SYSCAT.DM.mask('RANDOM',
/*Expression*/
)
```

RANDATE

RANDATE masks the values in a date column by replacing them with pseudo-random date values.

```
SYSCAT.DM.mask('RANDATE',
/*Expression*/
)
```

RANSTR

RANSTR masks the values in a column by replacing the values with random strings. Strings are generated by an algorithm that uses characters from the source string in the generation process, adding padding characters if necessary. Padding is placed to the left of the string unless RIGHT is specified.

```
SYSCAT.DM.mask('RANSTR',
/*Expression*/
)
```

RANDIG

RANDIG masks the numeric values in a column by replacing digits with strings of random digits. Strings are generated by an algorithm that uses digits derived from the base number system of the source value, adding padding digits if necessary. Padding is always to the left of digits.

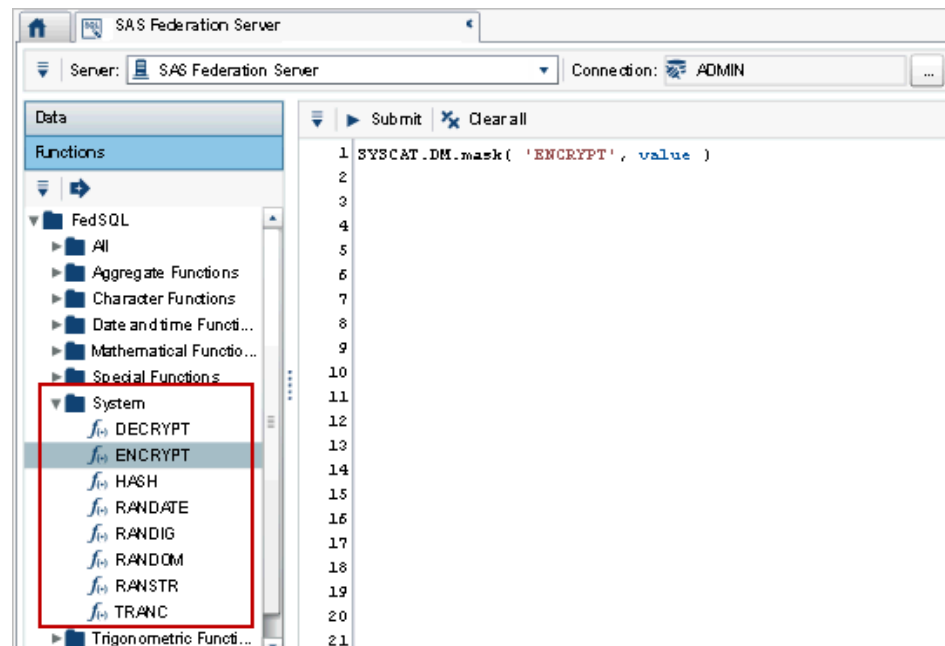
```
SYSCAT.DM.mask('RANDIG',
/*Expression*/
)
```

Displaying Data Masking Functions in the Console

To access the data masking functions in SAS Federation Server Manager:

1. Select a federation server in the tree and log on if necessary.
2. Select **Open Console** from the toolbar on the **Home** tab.
3. In the Console, select a server from the drop-down menu and select the ADMIN DSN (default).
4. In the left pane, select **Functions** and expand the **FedSQL** directory.
5. Select the **System** folder and expand it to display the data masking functions.

Figure 15.1 Data Masking Functions



See “Data Masking” in the *SAS Federation Server: Administrator’s Guide* for a list of arguments for each data masking function.

Data Masking in a FedSQL View

You can add data masking functions when you create a FedSQL view. To create a FedSQL view with masked data:

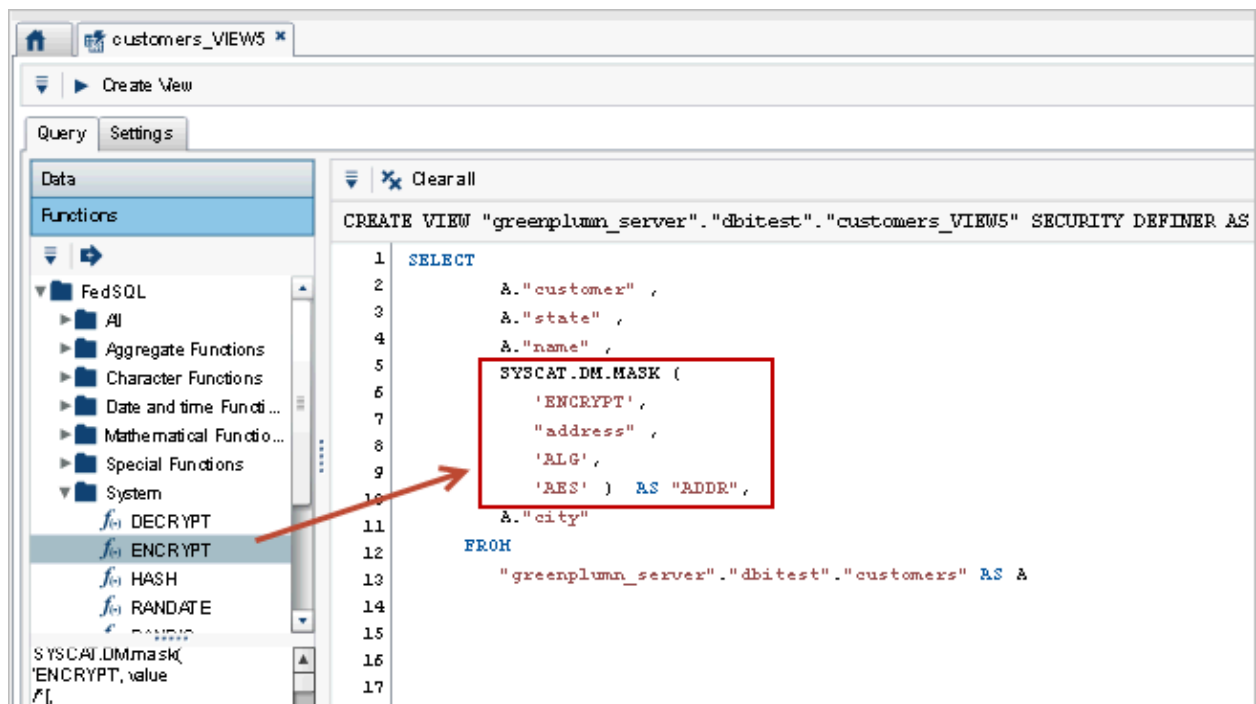
1. Log on to a federation server and navigate to a table: **Federation Server** ⇒ **Data Service** ⇒ **Table**.
2. Select **New FedSQL View from Table**.
3. Using the left panel, select **Functions** ⇒ **FedSQL** ⇒ **System** to expose data masking functions.

4. Select a data masking function from the list and click the arrow ➡ to move the function to the Create View statement on the right.
5. Configure the FedSQL view to include your data masking function. You can place the masking function anywhere in your SELECT statement. Here is an example:

```
SELECT A."customer" , A."state" , A."name" ,
       SYSCAT.DM.MASK (
         'ENCRYPT' ,
         "address" ,
         'ALG' , 'AES' ) AS "ADDR" ,
       A."city"
FROM
  "catalog"."schema"."customers_table" AS A
```

Note: Note that you can create multiple data masking scenarios in a view.

Figure 15.2 Creating a FedSQL View with Masked Data



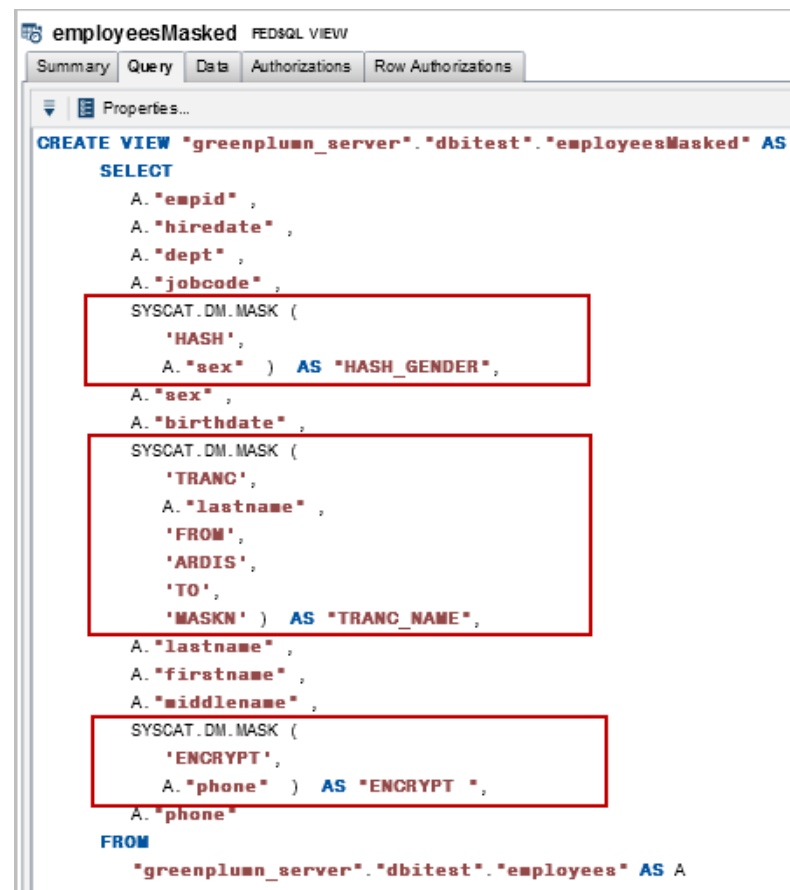
6. Select Create View.

Data Masking Examples

About the Examples

In the examples that follow, an ‘Employees’ table is used to demonstrate the encrypt, hash, and transliterated value (TRANC) data masking functions. Here is the select statement that contains the data masking syntax.

Figure 15.3 FedSQL View Query with Data Masking



Encrypt

The Employees table contains personal phone number data that needs to be encrypted. The SELECT statement contains the ENCRYPT function configured to encrypt the phone column and output to a new column titled ENCRYPT.

```
SYSCAT.DM.MASK ('ENCRYPT', A."phone") AS "ENCRYPT "
```

Here is the result:

Figure 15.4 Data Masking Encrypted Output

#	empid	hiredate	lastname	firstname	middlename	ENCRYPT
1	457232.0	07/15/85	LOVELL	WILLIAM	SINCLAIR	B1EC8159057...
2	459287.0	11/02/84	RODRIGUES	JUAN	M.	2FC72E71FB4...
3	458921.0	08/19/87	KRAUSE	KARL-HEINZ	G.	2FC72E71FB4...
4	123456.0	04/04/89	VARGAS	PAUL	JESUS	(null)
5	237642.0	11/01/76	BATTERSBY	R.	STEPHEN	2FC72E71FB4...
6	423286.0	12/19/88	MIFUNE	YUKIO	TOSHIRO	AEA1DB08F12...
7	456910.0	06/14/78	ARDIS	RICHARD	BINGHAM	AEA1DB08F12...
8	216382.0	06/15/85	PURINTON	PRUDENCE	VALENTINE	AEA1DB08F12...
9	321783.0	09/10/67	GONZALES	GUILLERMO	RICARDO	2C7C8820E70...
10	119012.0	07/01/68	WOLF-PROVE...	G.	ANDREA	2C7C8820E70...

HASH

The HASH rule hashes a single value into a fixed-length hash digest or HMAC string. In this example, gender data is hashed and output to a new column named 'HASH_GENDER'.

```
SYSCAT.DM.MASK ('HASH', A."sex" ) AS "HASH_GENDER"
```

Figure 15.5 Data Masking HASH Output

#	empid	hiredate	HASH_GENDER	sex	birthdate	TRANC_NAME
1	457232.0	07/15/85	CAE8013D2C4...	M	10/15/63	LOVELL
2	459287.0	11/02/84	CAE8013D2C4...	M	01/05/34	AOSAKGUEN
3	458921.0	08/19/87	CAE8013D2C4...	M	05/12/62	KAMUNE
4	123456.0	04/04/89	(null)	(null)	(null)	VMAGMN
5	237642.0	11/01/76	CAE8013D2C4...	M	03/13/54	BMTTEANBY
6	423286.0	12/19/88	CAE8013D2C4...	M	10/31/64	MKFUNE
7	456910.0	06/14/78	CAE8013D2C4...	M	09/24/53	MASKN
8	216382.0	06/15/85	2F5886124B90...	F	07/24/63	PUAKNTON
9	321783.0	09/10/67	CAE8013D2C4...	M	06/03/35	GONZMLEN
10	119012.0	07/01/68	2F5886124B90...	F	01/05/46	WOLF-PAOV

Transliterated Value (TRANC)

TRANC masks the values in a column by transliterating characters from the input string to characters in the output string. In this example, employee's last names are masked using transliterated values.

```
SYSCAT.DM.MASK ('TRANC', A."lastname" ,
                'FROM', 'ARDIS',
                'TO', 'MASKN' ) AS "TRANC_NAME",
```

Figure 15.6 Data Masking Transliterated Value Output

employeesMasked FEDSQL VIEW

Summary Query Data Authorizations Row Authorizations

View Data Go to: 1

#	empid	hire date	HASH_GENDER	birthdate	TRANC_NAME	lastname
1	457232.0	07/15/85	CAE8013D2C4...	10/15/63	LOVELL	LOVELL
2	459287.0	11/02/64	CAE8013D2C4...	01/05/34	AOSAKGUEN	RODRIGUES
3	458921.0	08/19/67	CAE8013D2C4...	05/12/62	KAMUNE	KRAUSE
4	123456.0	04/04/89	(null)	(null)	VMAGMN	VARGAS
5	237642.0	11/01/76	CAE8013D2C4...	03/13/54	BMTTEANBY	BATTERSBY
6	423288.0	12/19/68	CAE8013D2C4...	10/31/64	MKFUNE	MIFUNE
7	456910.0	06/14/78	CAE8013D2C4...	09/24/53	MASKN	ARDIS
8	216382.0	06/15/85	2F5886124B90...	07/24/63	PUAKNTON	PURINTON
9	321783.0	09/10/67	CAE8013D2C4...	06/03/35	GONZMLEN	GONZALES
10	119012.0	07/01/68	2F5886124B90...	01/05/46	WOLF-PAOV	WOLF-PROVE...

Chapter 16

SQL Logging

About SQL Logging	131
Enable SQL Logging	131
Viewing the SQL Log	132
Overview	132
Viewing the Reports	132
Individual Requests Report	133
Summarized Requests Report	133
User Report	134
Customizing the Reports	135
Exporting Report Data	136
SQL Logging Transactions	137

About SQL Logging

SQL Logging is the ability to view SQL statements and DS2 functions submitted to SAS Federation Server. With SQL logging enabled, you can view information such as the user who submitted the request accompanied by a breakdown of the request into prepare, execute and cursor phases. By using SQL logging, an administrator can easily determine what users are accessing the system, when they were connected, and what work was performed. For additional details about SQL logging, see the *SAS Federation Server: Administrator's Guide*.

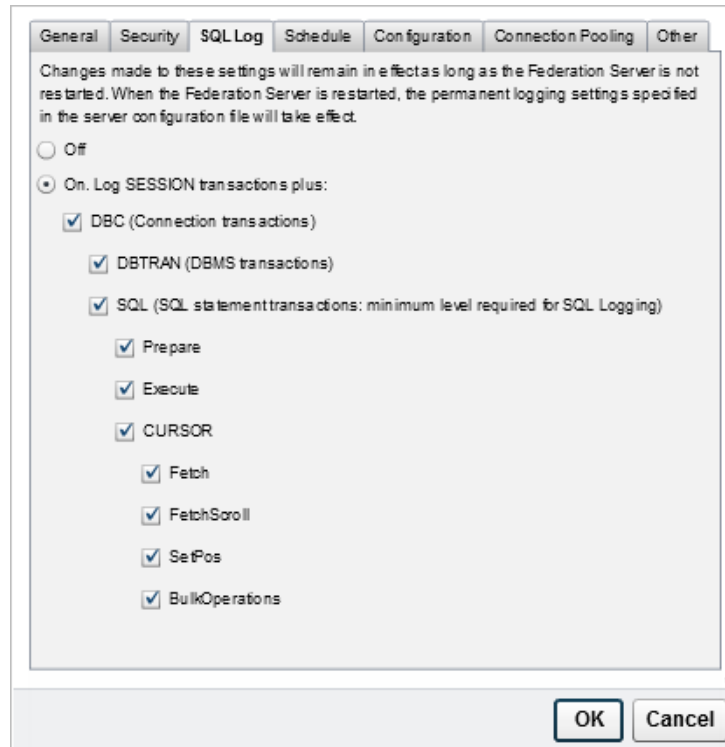
Enable SQL Logging

Use the following task to enable logging for a server session. Any parameters set for SQL logging during the session revert to the default configuration settings upon restart of the federation server. The information captured here does not affect regular server logging which is set within the `dfs_log.xml` configuration file on SAS Federation Server.

1. Select a federation server object in the tree.
2. Open the **Action** menu in the upper left corner and select **Properties**.
3. At the federation server properties dialog box, click the **SQL Log** tab.

- Click to select **On, Log SESSION** and select the events to record. See “[SQL Logging Transactions](#)” for a brief description of the behavior for each of these transactions.
- Click **OK** to accept the changes and close federation server properties.

Figure 16.1 Federation Server Properties: SQL Log




Viewing the SQL Log

Overview

SQL Logging status is displayed under the **Summary** tab of selected federation server object. SQL Logging must be active for the selected server and you must be logged in to a federation server to view logging results.

Viewing the Reports

SQL Log reports are accessed through the **Summary** tab of SAS Federation Server properties. To access the reports:

- Select a federation server object in the tree and log on.
- On the **Summary** tab, open the **Action** menu.
- Select **Open SQL Log** from the drop-down menu. You can also click the **Open SQL Log** icon  on the toolbar.
- Using the drop-down menu, select one of the reports and click **Open**.

- Individual Requests
- Summarized Requests
- User Report

Individual Requests Report

This detailed report shows complete SQL statements submitted by users within the previous hour. The data is sorted by start time in descending order. Total elapsed time is recorded in milliseconds. The following list shows some of the columns reflected in the default report.

- User ID
- Login ID
- Requests
- Statement ID
- IP Address
- Host Name and Port

To extend the time interval for this report or set additional parameters, use the Filter options located in the upper right corner of the window. You can also customize the report to add or remove columns.

Additional information for each report entry is contained in the following tabs listed beneath the item selected in the report.

Details

Lists details of the selected request or transaction.

Request

Displays the statement that was used for the selected request.

Plan

Specifies the execution plan that the underlying database uses to execute the SQL statement. It appears in XML format on the screen. If there is a problem with the XML, such as truncation, the message Invalid Plan is displayed until the problem is fixed.

Cache Access

Displays data if a cache view was accessed. The **Cache Access** column reflects a status of `true` if cache data was used and `false` if the request did not access any cached views.

Summarized Requests Report

The rows in the table are grouped based on the hashkey returned from the server. The hashkey in the server hashes to the same value when the submitted SQL is identical. This table is sorted with the most recent submission at the top. The default report includes the following columns:

- Requests
- Number of Requests
- Last Submitted

- Mean Request Lifetime (s)
- Mean Cursor lifetime (s)
- Cache Access
- Mean Work Time (s)

To extend the time interval for this report or set additional parameters, use the Filter options located in the upper right corner of the window. You can also customize the report to add or remove columns.

Additional information for each report entry is contained in the following tabs listed beneath the item selected in the report.

Details

Lists details of the selected statement or transaction.

Request

Displays the SQL used for the selected statement or transaction.

Plan

Specifies the SQL execution plan that the underlying database uses to execute the SQL statement. It appears in XML format on the screen. If there is a problem with the XML, such as truncation, the message Invalid Plan is displayed until the problem is fixed.

Cache Access

Displays data in this view if any of the SQL Statements accessed a cache view. Cache is reflected as 'true' in the Cache Access column of the table.

User Report

The SQL user report shows the same information that is displayed in the SQL Statements Table, but it is organized in a hierarchical manner. The hierarchy is: **user** ⇒ **session** ⇒ **connection** ⇒ **SQL statement**. The default report includes the following columns:

- Transaction
- Details
- Request Lifetime (s)
- Data (kb)
- Cache Access
- Start Time

To extend the time interval for this report or set additional parameters, use the Filter options located in the upper right corner of the window. You can also customize the report to add or remove columns.

Additional information for each report entry is contained in the following tabs listed beneath the item selected in the report.

Details

Lists details of the selected statement or transaction.

Request

Displays the SQL used for the selected statement or transaction.

Plan

Specifies the SQL execution plan that the underlying database uses to execute the SQL statement. It appears in XML format on the screen. If there is a problem with the XML, such as truncation, the message Invalid Plan is displayed until the problem is fixed.

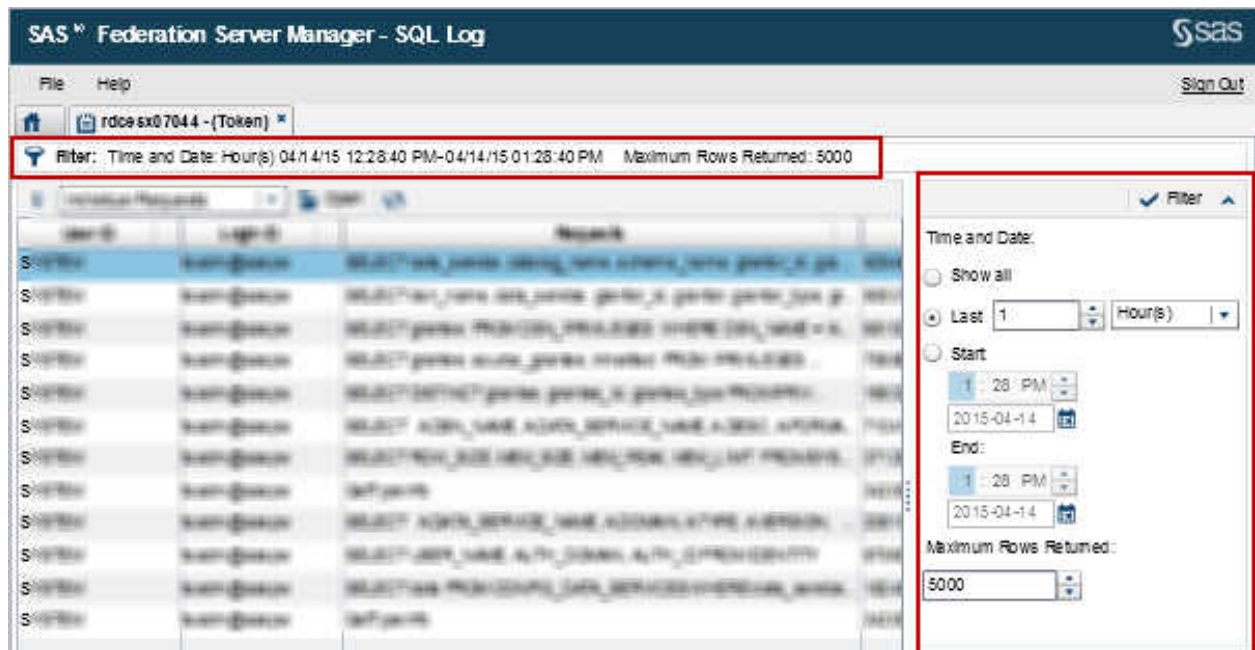
Cache Access

Displays data in this view if any of the SQL Statements accessed a cache view. Cache is reflected as 'true' in the Cache Access column of the table.

Customizing the Reports**Setting Filters for Logging**

When you first open SQL Log, a Filter status message is displayed, showing the current filters in effect for the session. You can set filters for each report using the filter configuration panel to the right of individual reports.

Figure 16.2 SQL Log Filter Options



Using the **Filter** option on the right side of the SQL Log panel, you can set filters to limit how much information is queried from the server.

1. Click the chevron next to **Filter** to display the filter options for SQL logging and configure one of the following options:
 - Select **Show all** to display everything. However, selecting this option can impede performance.
 - Select **Last** and enter a number of Minutes, Hours, Days, or Weeks to filter results for a specific time frame.
 - Select **Start** and enter a start time and date, and an end time and date, to filter results for a specific period of time.
 - Set a number for **Maximum Rows Returned** to prevent potential long-running queries. The default setting for this option is 5000.

2. To save your selections, click the checkmark next to **Filter**.

Figure 16.3 SQL Logging Filters

Time and Date:

☐ Showall

☐ Last 1 Hour(s)

☒ Start:

7 : 15 AM

2015-06-08

End:

6 : 15 PM

2015-06-10

Maximum Rows Returned:

5000

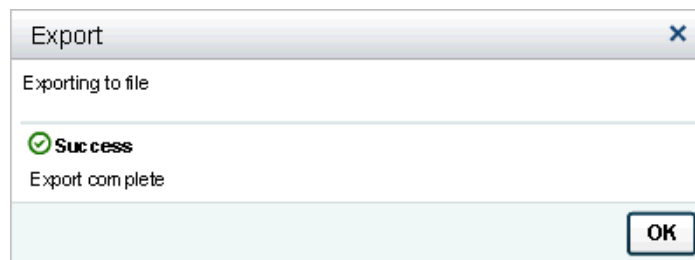
Exporting Report Data

You can export report data for SQL Logging to a CSV file, but you must have permissions to write to the folder that you are exporting to. To export report data to a CSV file:

1. Open the **Action** menu in the upper left corner of the report and select **Export to CSV**.
2. Select a location for the exported file and click **Save**.
3. You should see a message indicating that the export is complete. Click **OK** to return to SQL Logging.

Note: If a message is not displayed, verify that you have permissions to write to the folder that you are exporting to.

Figure 16.4 SQL Log Report Export Complete



SQL Logging Transactions

The following table shows the ARM transactions that are captured in SAS Federation Server Manager. When SQL Logging is enabled, information in each of the transactions is captured. Following the table is a brief explanation of each of the transactions.

Name	Type	Associated Namespace
SESSION	Session transaction	Perf.ARM.FederationServer.Session.Transaction.SESSION
DBC	Database connection	Perf.ARM.SQLServices.Connection.Transaction.DBC
DBTRAN	Database transaction	Perf.ARM.SQLServices.Connection.Transaction.DBTRAN
SQL	SQL statement	Perf.ARM.SQLServices.Statement.Transaction.SQL
Prepare	SQL Statement	Perf.ARM.SQLServices.Statement.Prepare
Execute	SQL Statement	Perf.ARM.SQLServices.Statement.Execute
CURSOR	SQL Statement	Perf.ARM.SQLServices.Statement.Transaction.CURSOR
Fetch Scroll	SQL Statement	Perf.ARM.SQLServices.Statement.FetchScroll
SetPos	SQL Statement	Perf.ARM.SQLServices.Statement.SetPos
BulkOps	SQL Statement	Perf.ARM.SQLServices.Statement.BulkOperations
Fetch	SQL Statement	Perf.ARM.SQLServices.Statement.Fetch

SESSION

(Session Transaction) A session transaction starts when a user initiates a server session.

DBC

(Database Connection) A database connection transaction is a child object of the SESSION transaction. A database connection begins when a user connects to a data source and ends when the user disconnects from the data source.

DBTRAN

(RDBMS Transaction) DBTRAN is the actual database transaction. It is a child object of the DBC transaction. A DBTRAN transaction begins with an established driver connection, or when a previous transaction is committed or rolled back, and a new one begins. DBTRAN records are written to the log only if AUTOCOMMIT is set to OFF. The DBTRAN transaction stops when AUTOCOMMIT is set to ON or when a COMMIT or ROLLBACK command is issued. SQL statements can span DBTRAN transaction boundaries

SQL

(SQL Statement) SQL is a logical transaction. It encapsulates a series of activities related to one SQL statement. It is a child object of a DBC transaction. An SQL transaction starts when a user issues an SQL statement. Regardless the statement type (DQL, DML, or DDL) the SQL transaction stops when the statement is either closed or unprepared. Subsequent executions of the same statement are recorded under the same SQL transaction, even if the statement is a DQL and the result set associated with it is closed.

Prepare

The Prepare transaction measures the Prepare phase of an SQL statement. It is a child object of an SQL transaction. The Prepare transaction starts when a user Prepares an SQL statement and stops when the call to prepare returns.

Execute

The execute transaction measures the Execute phase of an SQL statement. It is a child object of the SQL transaction. The Execute transaction starts when a user executes an SQL statement and stops when the call to execute returns.

CURSOR

CURSOR is a logical transaction. CURSOR is a child object of an SQL transaction and it encapsulates all operations executed in a cursor, including reading, positioning and updates. The CURSOR transaction starts when the Execute transaction finishes. It stops when the cursor is closed. All operations on the same result set belong to the same CURSOR transaction.

Fetch/Fetch Scroll

The FETCH transaction is a child object of the CURSOR transaction. The FETCH transaction has an Execute transaction as its predecessor. It is started when a user issues the first fetch on a result set using Fetch or Fetch Scroll. It stops when the call to Fetch or Fetch Scroll returns

SetPos

The SetPos transaction is a child object to a CURSOR transaction. The SetPos transaction has an execute transaction as its predecessor. It is started when a user issues a SetPos call and stops when the call returns.

BulkOperations

The BulkOperations transaction is a child object of a CURSOR transaction. The BulkOperations transaction has an Execute transaction as its predecessor. It is started when a user issues a call to BulkOperations and stops when the call returns.