Defining a Teradata Library with the TERADATA Engine in SAS® Management Console

Table of Contents

Overview

.As a SAS software user, you might need to query a Teradata table from SAS® business-intelligence (BI) clients. There are several ways that you can access a Teradata database from a BI client application. However, the preferred method, which is described in this document, is to define a Teradata library in SAS® 9.2 Management Console or later. SAS Management Console provides a point of control for security management of this library.

To define a Teradata library in SAS Management Console with the TERADATA engine, you need to have SAS/ACCESS® 9.2 (or later) Interface to Teradata installed on your SAS server. You also must have the Teradata client libraries installed, as explained in the next section, “Step 1: Install the Teradata Client Software.” You should install the Teradata client software and connect to the Teradata database (steps 1 and 2) before you begin defining the library (steps 3-5).

Step1: Install the Teradata Client Software

Before you can connect to the Teradata database from SAS, you must have Teradata client software (specifically the CLIv2 libraries) installed on the SAS server from which you are running SAS/ACCESS Interface to Teradata. The Teradata client libraries are not part of the SAS installation; they are installed from the Teradata Tools and Utilities (TTU) application. Your database administrator will have to help you install these libraries. After your administrator installs and configures the Teradata client libraries, test the connection outside of SAS by using the Basic Teradata Query (BTEQ) utility. A successful logon to the Teradata database server from the BTEQ utility indicates that the Teradata client is configured and that you can proceed with the SAS configuration. However, if the connection from the BTEQ utility fails, you should contact your database administrator.

Windows Operating Environments

If your SAS server runs under Windows, ensure that the path to the Teradata client libraries is appended to the Windows PATH system variable. Generally, the path to the client libraries is **`C:\Program Files\NCR\Teradata Client\bin`**. You also might need to add an entry in the host file on your PC to provide the network address of the Teradata server. Typically, this means adding a **`dbccop1`** entry to your host file. For more details about the entry, contact your Teradata database administrator.

UNIX Operating Environments

If your SAS server runs under UNIX, set the appropriate environment variables in the sasenv_local file that is located in the **`!SASROOT/bin`** directory. The executable file for SAS/ACCESS Interface to Teradata uses shared libraries, which are referred to as *shared objects* under UNIX. These shared objects typically reside in the **`/usr/lib`** directory. You must add the location of the shared libraries to one of the system environment variables. The following examples show where to set these locations in AIX, Linux, Solaris, and HP-UX operating environments.

AIX Environment

```
$ LIBPATH=path-to-Teradata-client:$LIBPATH
$ export LIBPATH
```

Linux and Solaris Environments

```
$ LD_LIBRARY_PATH=path-to-Teradata-client:$LD_LIBRARY_PATH
$ export LD_LIBRARY_PATH
```

HP-UX Environment

```
$ SHLIB_PATH=path-to-Teradata-client:$SHLIB_PATH
$ export SHLIB_PATH
$ LD_PRELOAD=/usr/lib/hpux64/libpthread.so.1
$ export LD_PRELOAD
```

For more information, see the *Configuration Guide for SAS® 9.2 Foundation for UNIX® Environments*.

Step 2: Connect to the Teradata Database from the Foundation SAS® Server

After you verify the system requirements, test the Teradata connection from Foundation SAS using a LIBNAME statement, as shown below:

```
libname teralib teradata server=server-name user=user-ID pwd=password;
```

In this statement, use your own Teradata connection information for the *server-name*, *user-ID,* and *password* values.

After you assign the LIBNAME statement, you can view your Teradata tables in the TERALIB library either by viewing it from the SAS Explorer window or by submitting a DATASETS procedure that is similar to the following:

```
proc datasets lib=teralib;
quit;
```

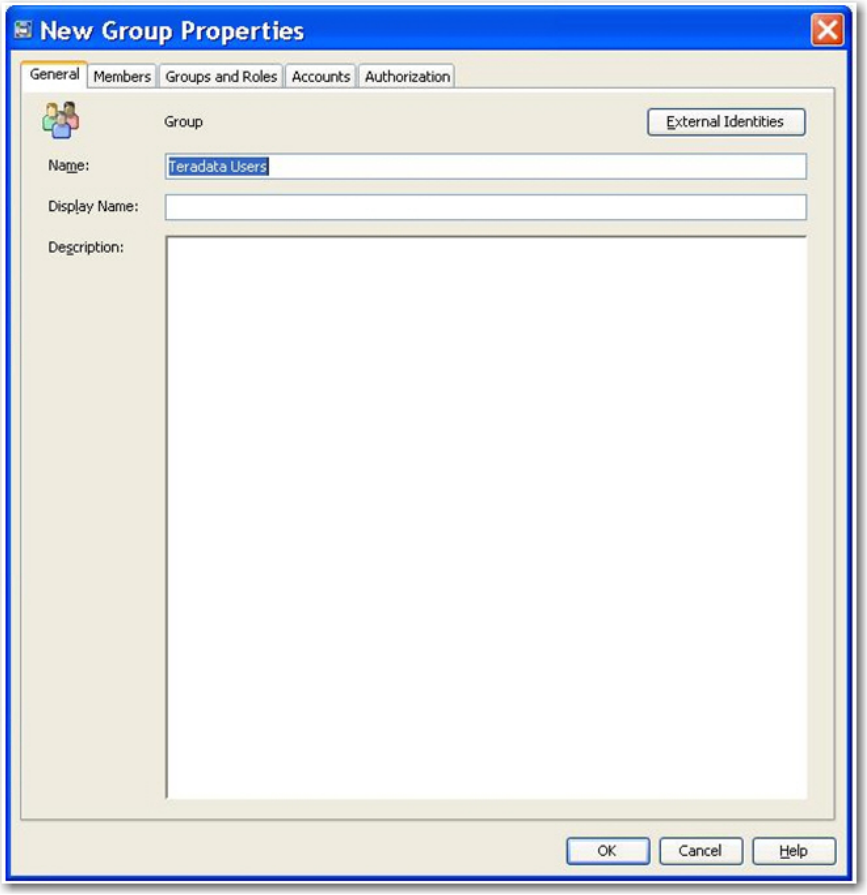
If you use PROC DATASETS, the tables are displayed in the SAS log.

Step 3: Add Teradata Users in SAS® Management Console

This section provides steps for adding logon credentials for a user group and for individual users.

Adding Logon Credentials for a User Group (Shared Teradata Credentials)

1. To add a new user group for the Teradata logon credentials, select **Actions ►New ►Group** from the User Manager plug-in to open the New Group Properties dialog box.
2. Name your new user group (for example, **`Teradata Users`**), as shown in the following display:

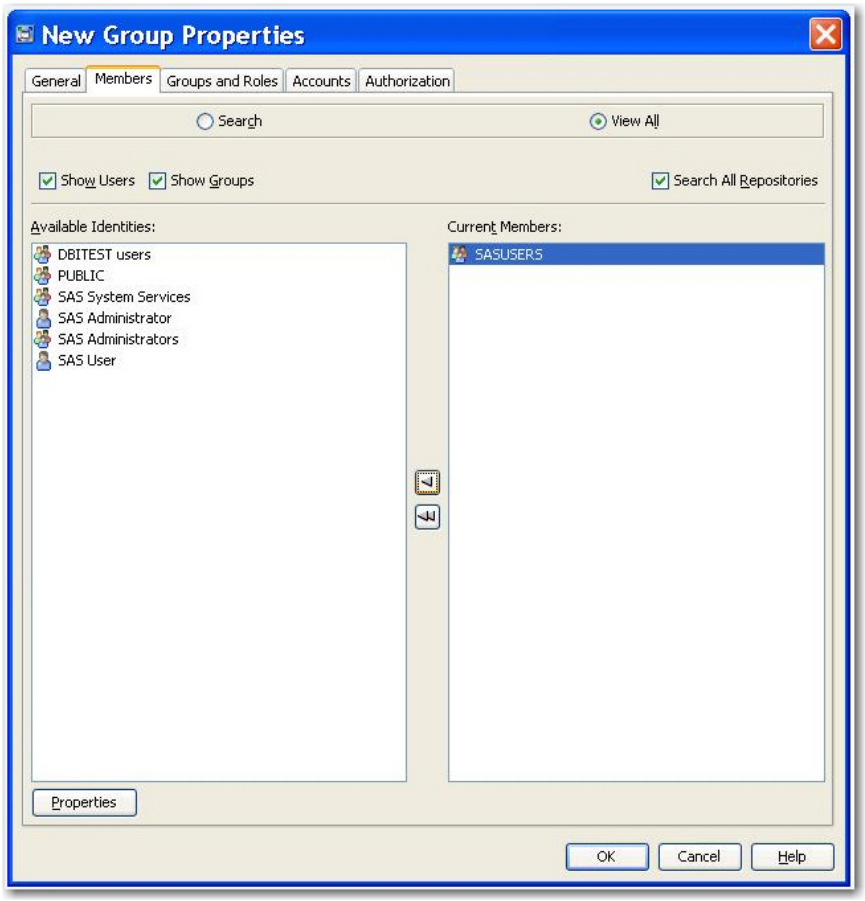


Note: You do not have to click OK to save your information. You can do that after you complete the information on each of the necessary tabs.

3. On the **`Accounts`** tab, click **New** to open the New Login Properties dialog box. You add your Teradata logon credentials in this dialog box.

For the **Authentication Domain** value, select your authentication domain (in this example, **Teraauth**) from the drop-down list. If the appropriate authentication domain is not listed, click **New** to add it. Then click **OK**.

4. Click the **Members** tab and add all of the users that need to access to this library. To provide access for all registered users, give logon access to the SASUSERS group by moving it from the **Available Identities** box to the **Current Members** box. Then click **OK**.

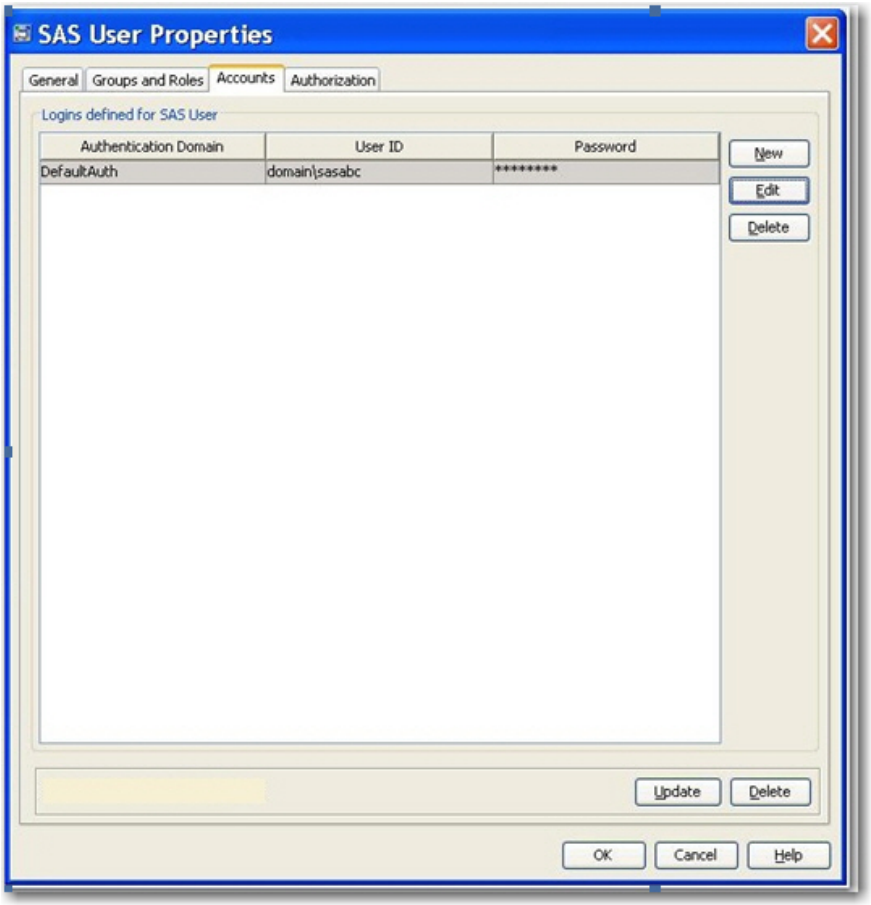


Adding Logon Credentials for Individual Users

Adding credentials for individual users provides the greatest accountability, but it can necessitate storing many sets of Teradata credentials.

1. In User Manager, open the properties of an existing user name for which you want to allow access to the Teradata server.

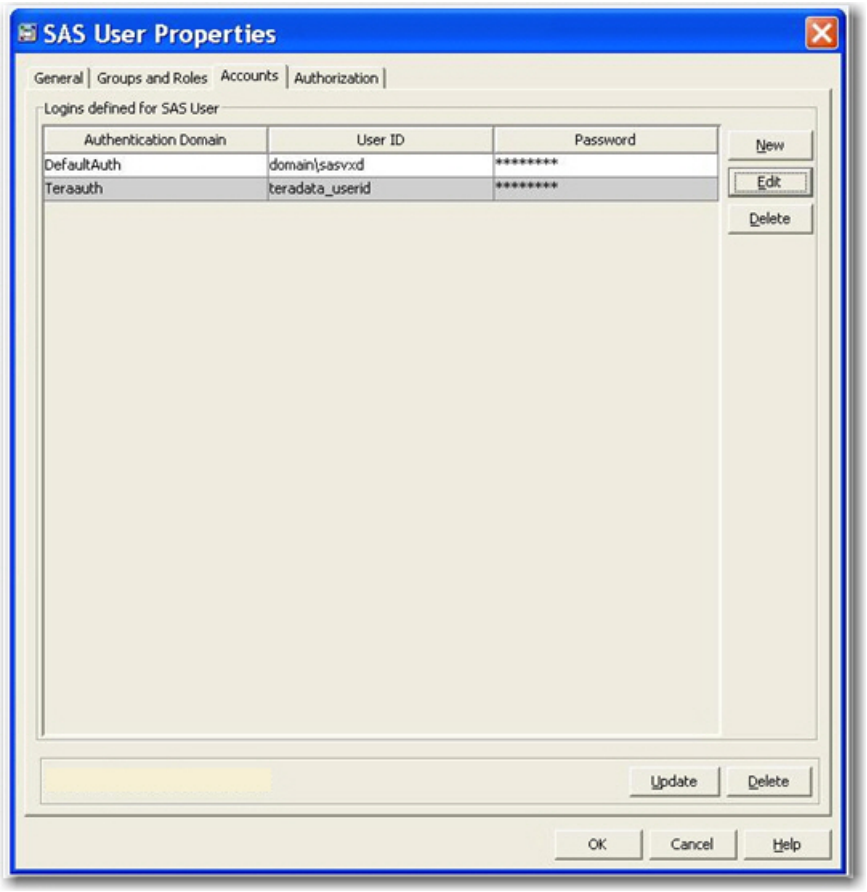
2. Click the **Accounts** tab.



3. On the **Accounts** tab, click **New** to open the New Login Properties dialog box.

Enter the Teradata user ID and password. For the **Authentication Domain** value, select your Teradata authentication domain (for example, **Teraauth**) from the drop-down list.

If the appropriate authentication domain is not listed, click **New** and add the domain.



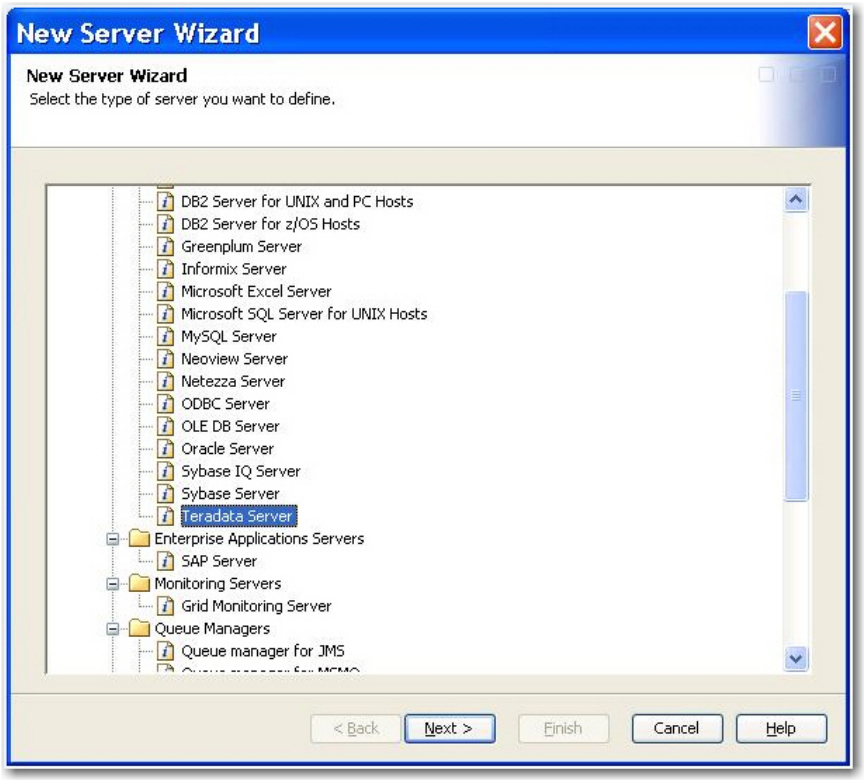
Each user's Teradata logon can use the same authentication domain. As a result, you build only one Teradata server. The user ID and password for the library is determined by the credentials that you use to log on. Users can see their individual logon credential as well as logon credentials of groups for which they are members. For example, when you log on to the SAS® Metadata Server, the server first finds your metadata identity. Then, if you access a particular Teradata server, the corresponding authentication domain that is defined on the server (**`Teraauth`**) maps to a user ID and password with the same authentication domain.

Step 4: Define the Teradata Server

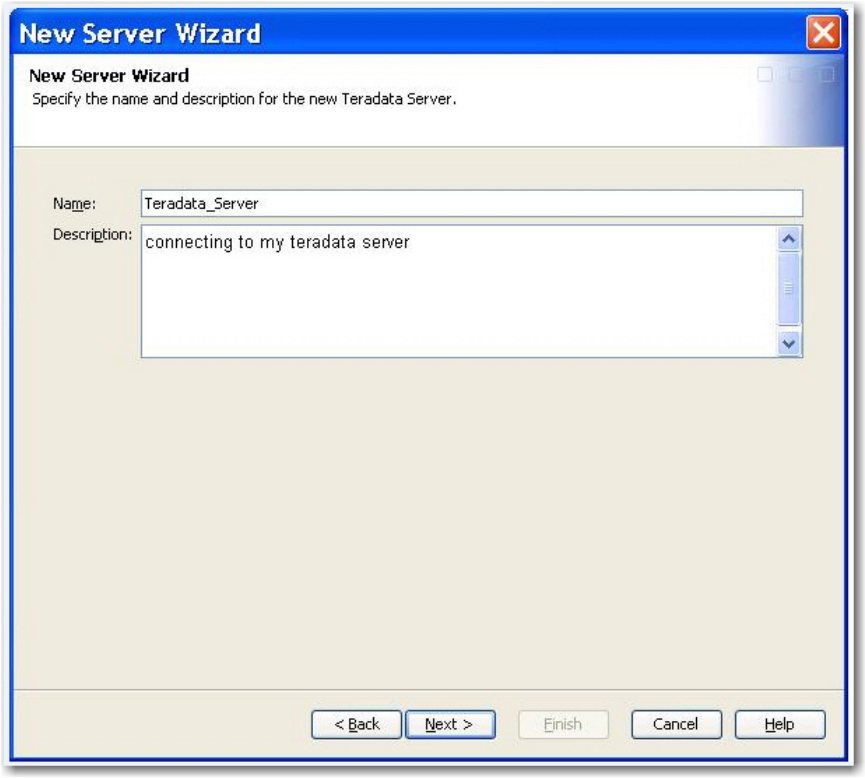
To define the Teradata Server, follow these steps In SAS Management Console:

1. Right-click **`Server Manager`** and select **`New Server`** from the menu that appears. This selection invokes the New Server Wizard.

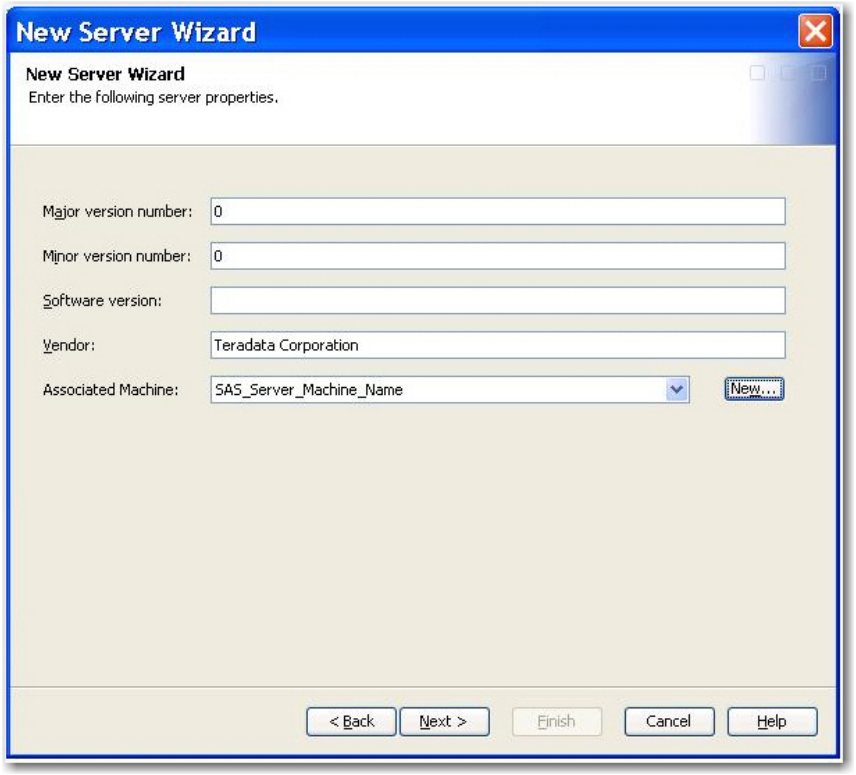
2. Select **Teradata Server** (under **Database Servers**) in the New Server Wizard. Then click **Next**.



3. Specify a name and a description for the new Teradata server, as shown below.



4. Click **Next** to continue to the next page. This page list default server properties. **Do not change** these default values:



5. Click **Next** to continue to the next page. On this page, enter the Teradata connection properties for your Teradata database.

In the previous display:

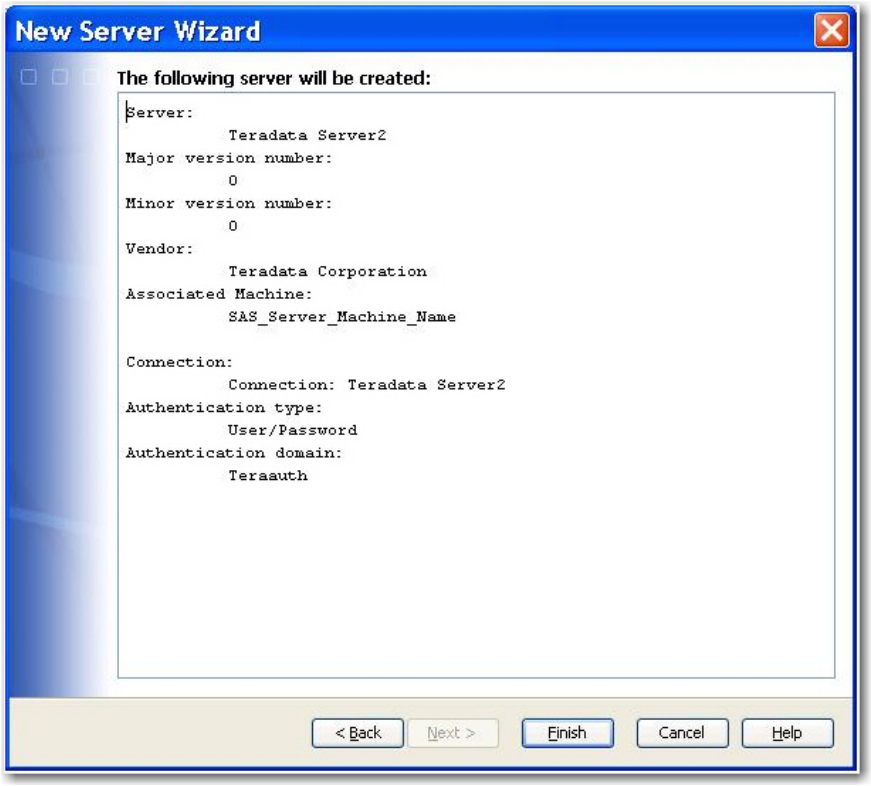
- **Server** specifies an entry in your (client) HOSTS file that provides an IP address for a database server connection. The value for this property corresponds to the value for the SERVER= option in the SAS/ACCESS software LIBNAME statement.

- **Account number** specifies the account number that you want to charge for the Teradata session. **Note:** This property is optional.

- **Authentication type** specifies the credentials (in this example, **User/Password**) that are retrieved from metadata and that correspond to the selected authentication domain, **TeraAuth**.

- **Authentication domain** specifies the domain (in this example, **Teraauth**) that is used to authenticate logon attempts to the Teradata server.

6. Click **Next**, and the wizard displays a summary page with information about the server that is going to be created.



7. Click **Finish** to return to the main SAS Management Console window.

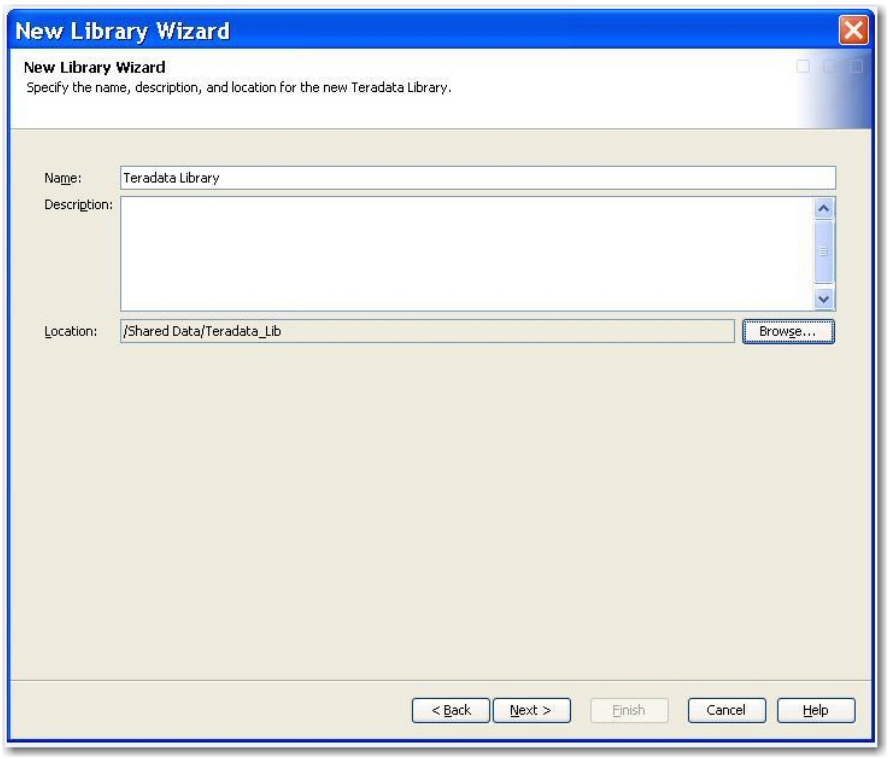
Step 5: Add the Teradata Library with the SAS® Management Console Data Library Manager Plug-in

1. In SAS Management Console, expand the **Data Library Manager** node.

2. Right-click **Libraries** and select **New Library** from the menu to open the New Library Wizard.

3. Under **Database,** select **Teradata Library**. Then click **Next** to continue to the next page of the wizard.

4. Specify a name for your library and select the folder (known as the metadata folder) in which to save this library.

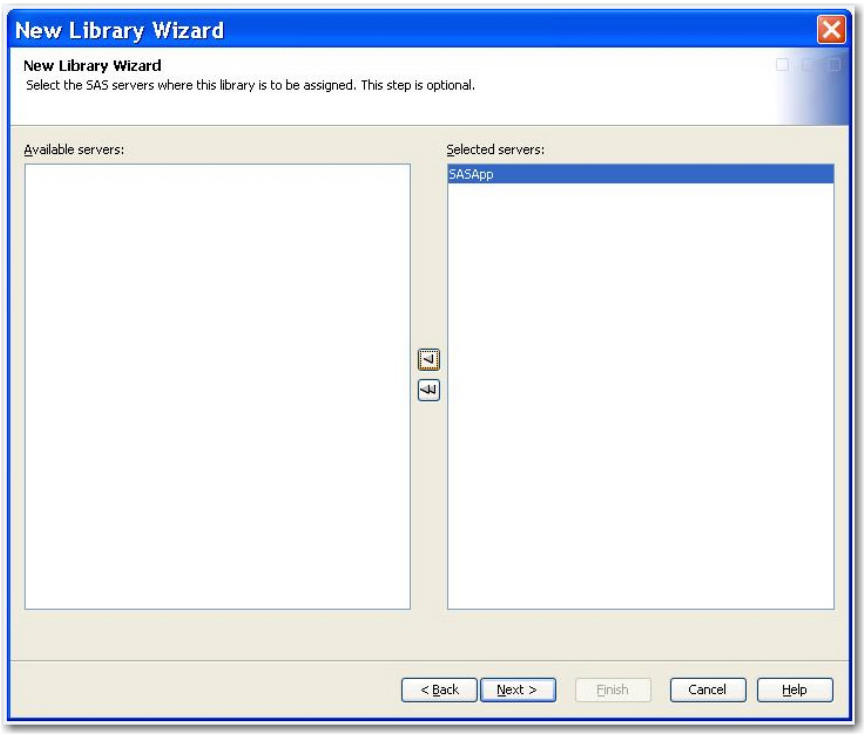


   Using a metadata folder to organize library and table objects in metadata (or any other metadata content that is supported by folders) is an efficient way to control access to those objects. Library and table objects inherit permissions from their respective metadata folders. Controlling access to these objects through permission inheritance is much easier when the objects are stored in the same metadata folder. You control access to the libraries and tables by denying ReadMetadata permission to the folder in which the library and tables are stored.

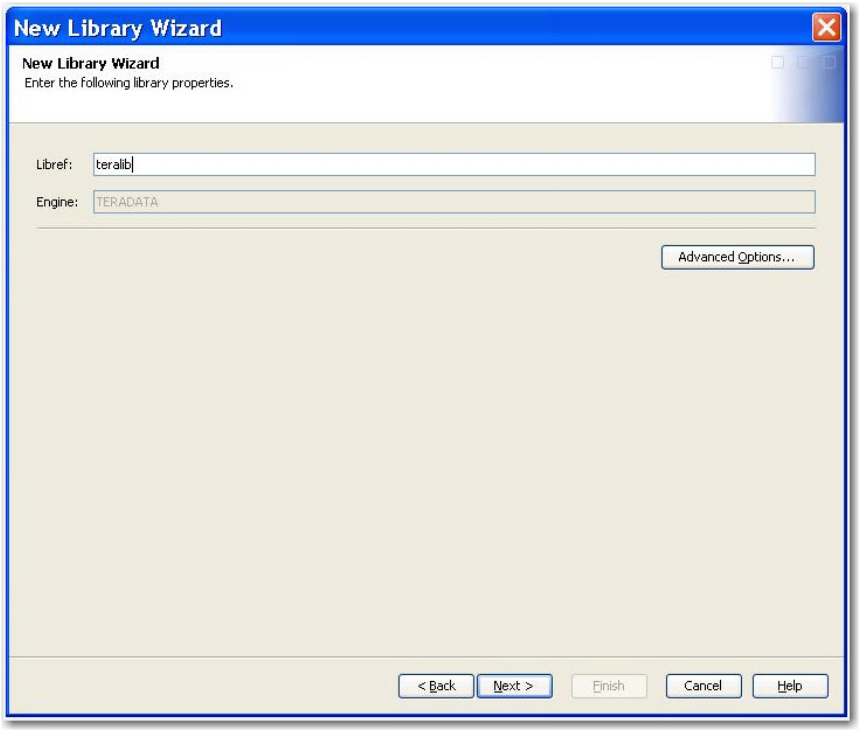
   For more details about metadata folders, see "Working with SAS Folders" in the *SAS® 9.2 Intelligence Platform: System Administration Guide, Second Edition*.

5. Click **Next** to continue to the next page of the wizard.

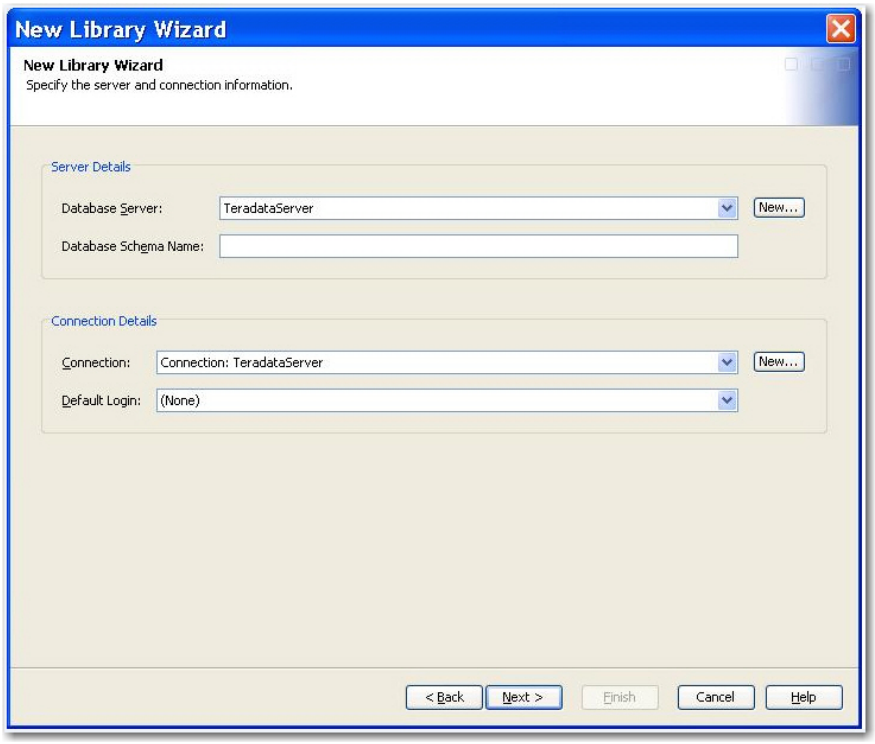
6. Select **SASApp** from the **`Available servers`** list and move it to the **`Selected servers`** list on the right.
   Note: Even though the wizard states that this step is optional, you **must** select **SASApp** as the server.



7. Click **`Next`** to continue to the next page of the wizard. Then enter a value of your choice for **`Libref`**.

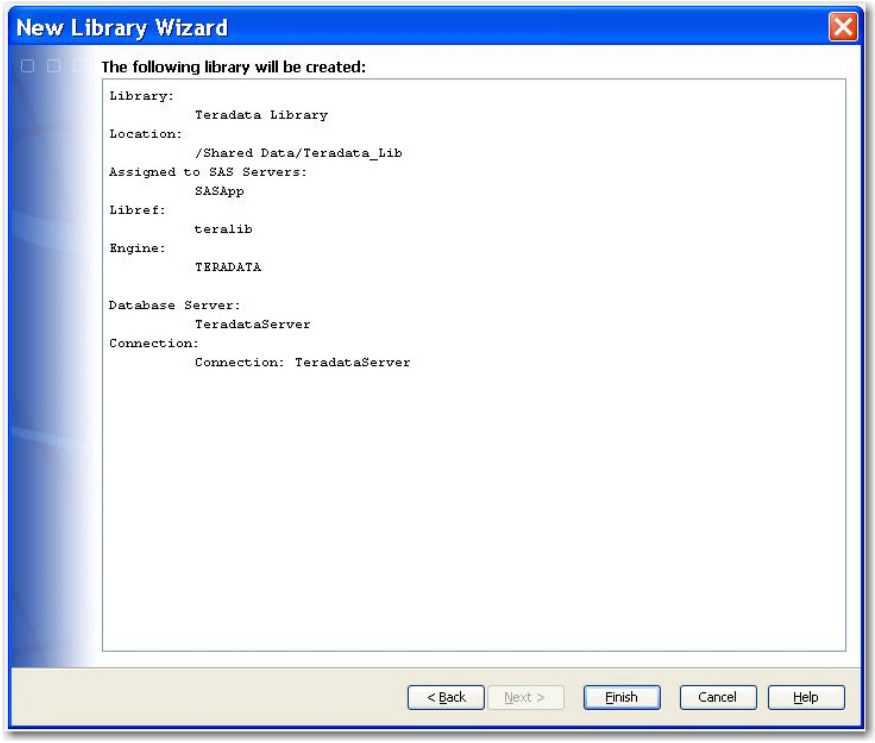


8. Click **Next** to move to the server and connection information page. Select the Teradata server that you created in the section “Step 4: Define the Teradata Server.”



Note: Adding a value for **Database Schema Name** is optional.

9. Click **Next** to advance to the next page, and verify your information. Then click **Finish** to exit the wizard.



At this point, your library is defined. However, to access data from your BI clients, you either need to register the tables or pre-assign the library.

For details about registering tables, see "Connecting to Common Data Sources: Registering and Verifying Tables" in the *SAS® 9.2 Intelligence Platform: Data Administration Guide*.

Resources

SAS Institute Inc. 2009*. Configuration Guide for SAS® 9.2 Foundation for UNIX® Environments.* Cary, NC: SAS Institute Inc. Available at **support.sas.com/documentation/installcenter/en/ikfdtnunxcg/61994/PDF/default/config.pdf**.

SAS Institute Inc. 2009. "Connecting to Common Data Sources: Registering and Verifying Tables," in *SAS® 9.2 Intelligence Platform: Data Administration Guide*. Cary, NC: SAS Institute Inc. Available at **support.sas.com/documentation/cdl/en/bidsag/61236/PDF/default/bidsag.pdf**.

SAS Institute Inc. 2011. "Working with SAS Folders," in *SAS® 9.2 Intelligence Platform: System Administration Guide*, 2d ed. Cary, NC: SAS Institute Inc. Available at **support.sas.com/documentation/cdl/en/bisag/64088/PDF/default/bisag.pdf**.