THE POWER TO KNOW.

SAS® Customer Link Analytics 6.5: Administrator's Guide

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2016. *SAS® Customer Link Analytics 6.5: Administrator's Guide*. Cary, NC: SAS Institute Inc.

SAS® Customer Link Analytics 6.5: Administrator's Guide

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

June 2016

Contents

Using This Book

Audience

This guide is written for administrators who want to install and configure SAS Customer Link Analytics. The administrator must be able to install, configure, administer, and use SAS Intelligence Platform, which SAS Customer Link Analytics uses. For details, see http://support.sas.com/documentation/onlinedoc/intellplatform.

The system administrator should have the skills to perform the following types of installation, configuration, and administration tasks:

- use SAS Download Manager to download SAS Software Depot to each machine where the installation will be performed.
- install and configure SAS Intelligence Platform and the solution. The system administrator should install and configure the required SAS Enterprise Intelligence Platform software on the required operating system.
- administer solution metadata. The system administrator must use SAS Management Console to maintain the metadata for servers, users, and other global resources that are required by the solution.

A thorough understanding of the target network configuration is critical, especially when setting up the grid software.

Document Conventions

The following table lists the conventions that are used in this document:

Document Conventions

Convention	Description
<SAS Home>	Represents the path to the folder where SAS is installed. For example, on a Windows computer, this path can be **`C:/Program Files/SASHome`**.
<SAS configuration directory>	Represents the path to the folder where SAS configuration data is stored. For example, on a Windows computer, this path can be **`C:/SAS/Config`**.

Convention	Description
<Project path>	Represents the path to the folder where the project's data is stored. This path is configured as a software component property. For more information, see "SAS Customer Link Analytics Server Component Properties" on page 52. For example, this path can be: **`/Shared Data/SAS Customer Link Analytics/ Cust Link Analytics 6.5/Projects`**.

Part 1

Installation and Configuration

2

1

Introduction to SAS Customer Link Analytics

Overview of SAS Customer Link Analytics

In recent years, customers have become more sophisticated and well-informed in their buying decisions than ever before. They rely on and seek advice from their network of friends, family, and acquaintances. As a result, there is explosive growth in the number of customer acquisitions. Also, with the increasing market penetration, the traditional methods of campaigning, such as telemarketing and advertising, are no longer necessarily applicable. Therefore, it is imperative for marketers to develop marketing strategies based on meaningful insights that are gained from network or transactional data analysis. This data captures interactions in the customer base, such as how much they interact, with whom they interact, and so on. The strength of relationships within their network and outside their network can reveal more information than static attributes such as their demographic information. Marketers can then use these insights to target their customers more accurately and effectively.

SAS Customer Link Analytics is a comprehensive solution for analyzing and controlling network data processing. It enables marketers to analyze network data, identify communities within the network, and quantify the relative importance of nodes within a community or network from various aspects. In addition, it enables marketing analysts to identify the role that each node plays within its community.

How SAS Customer Link Analytics Works

Overview

SAS Customer Link Analytics is a comprehensive solution that interacts with an external source system to extract and process subscription network data. It operates in distributed mode and non-distributed mode. For more information, see “Modes of Processing” on page 35.

Figure 1.1 *Solution Flow Diagram: Non-Distributed Mode*

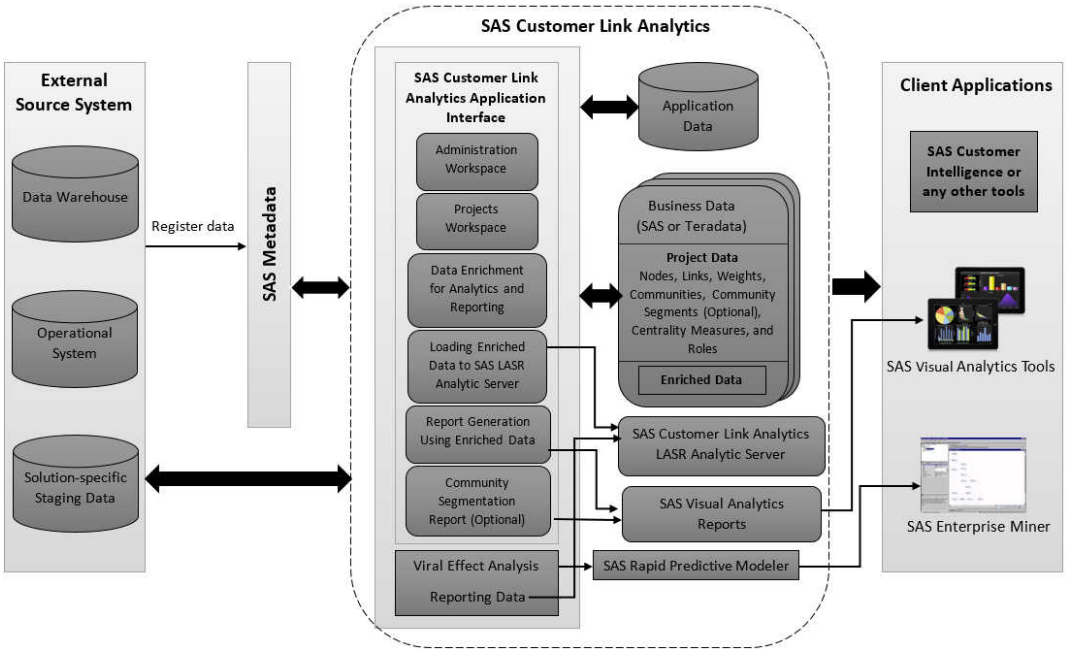
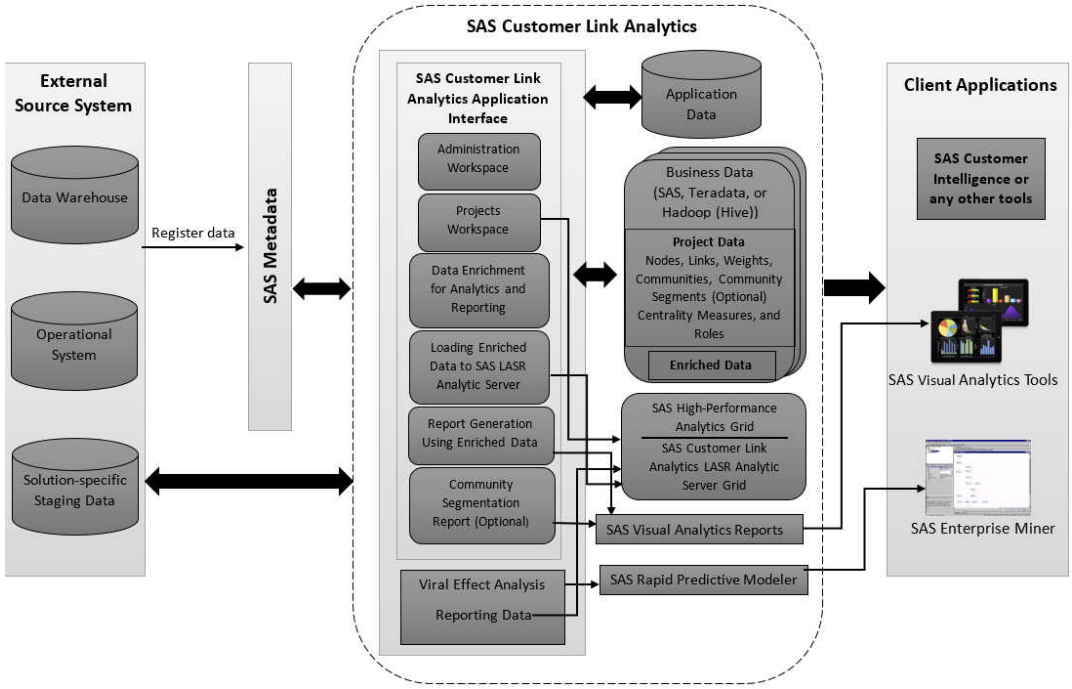


Figure 1.2 *Solution Flow Diagram: Distributed Mode*



The solution comprises the following components:

External source system
: the system with which SAS Customer Link Analytics interfaces to extract transactional data and other information such as node and link attributes. SAS Customer Link Analytics uses this data to build communities. The external source system can be a data warehouse, an operational system, or staging data that is specific to SAS Customer Link Analytics.

SAS metadata
: the data from the external source system has to be registered in SAS metadata. This registered data can then be imported into SAS Customer Link Analytics and used by the application. In addition, SAS Customer Link Analytics registers the project's output data, enriched data, viral effect analysis results, and community report in the metadata.

SAS Customer Link Analytics interface
: a workflow-based application that enables you to perform the important tasks listed here:

  Administration workspace
  : enables you to configure the metadata that is required for the SAS Customer Link Analytics workflow.

  Projects workspace
  : enables you to define projects, configure and run workflow steps, enrich the project's output data, load the enriched data to the SAS Customer Link Analytics LASR Analytic Server, and view the community report.

  Data enrichment for analytics and reporting
  : enables you to enrich the project's output data based on predefined categories. SAS Customer Link Analytics uses this data for reporting. Moreover, you can use the enriched data in tools such as SAS Customer Intelligence for further analysis.

Loading enriched data to the SAS Customer Link Analytics LASR Analytic Server
: SAS Customer Link Analytics enables you to load the enriched data to the SAS Customer Link Analytics LASR Analytic Server. SAS Customer Link Analytics uses this data to produce the community report.

Report generation using enriched data
: SAS Customer Link Analytics produces the community report based on the enriched data that it loads into the SAS Customer Link Analytics LASR Analytic Server. It also enables you to view this report in the SAS Visual Analytics Viewer seamlessly by using the Flex application switcher.

Community Segmentation report (Optional)
: SAS Customer Link Analytics produces the community segmentation report if you choose to create community segments when you run the Centrality Measures Computation workflow step. It also enables you to view this report in the SAS Visual Analytics Viewer in a seamless manner by using the Flex application switcher.

Viral effect analysis
: enables analysis of SAS Customer Link Analytics output, construction of an analytical model using SAS Rapid Predictive Modeler, and generation of analytical scores and reporting data for viral effect analysis. The analytical model can be further explored in SAS Enterprise Miner and the scores can be used by marketing automation tools.

Application data
: stores project-specific data and configuration details of the source data. Also, stores the summary of results that the SAS Customer Link Analytics solution produces when each workflow step is run. These results include information about communities, roles, and centrality measures.

  Application data also contains the configuration details that are required for running the data enrichment process.

Business data
: stores the final output that SAS Customer Link Analytics produces when all the workflow steps of a project are run. This output contains node-level information such as the role ID, community ID, and centrality values. In addition, business data contains data that SAS Customer Link Analytics produces when you run the data enrichment process.

  Business data also contains the intermediate tables that SAS Customer Link Analytics creates when the workflow steps are run.

SAS High-Performance Analytics Server Grid
: in distributed mode, provides tools for performing analytic tasks of community building and centrality measure computation in a high-performance environment. This environment is characterized by massively parallel processing (MPP) on a distributed system. For more information, see *SAS Customer Link Analytics: Administrator's Guide*.

SAS Customer Link Analytics LASR Analytic Server Grid
: an analytic platform that provides a secure, multi-user grid environment for concurrent access to enriched data that SAS Customer Link Analytics loads into memory to produce the community report. In the distributed mode, the server distributes data and the workload among multiple machines and performs massively parallel processing. However, in the non-distributed mode, server is deployed on a single machine as the workload and data volumes do not require a distributed computing environment. In addition, if

the business data is stored in Hadoop, SAS Customer Link Analytics runs the analytical processes of Community Building and Centrality Measures Computation workflow steps alongside LASR.

SAS Visual Analytics tools
: reporting tools for business analysts to explore, view, and analyze data and create and view reports that help them make business decisions.

SAS Customer Intelligence or any other similar tools
: a suite of marketing automation tools that enable organizations to manage interactions along the customer journey in a personalized and profitable way. SAS Customer Intelligence provides analytically driven capabilities in the four areas that the modern marketing organization needs in today's digital world: strategy and operations, marketing analytics, multichannel engagement, and digital experience.

  You can use the project's output data and the enriched data that SAS Customer Link Analytics produces for defining target lists or campaigns in SAS Customer Intelligence and taking marketing actions. You can thereby enhance your campaigning strategies.

Solution Flow

The SAS Customer Link Analytics solution flow includes the following steps:

1. Register data from the external source system in SAS metadata. This step is not within the scope of this document.
2. Log on to SAS Customer Link Analytics as an administrator and perform the following tasks. For more information about each of these steps, see *SAS Customer Link Analytics: User's Guide*.
   a. Import tables that are registered in SAS metadata, configure them, and then refresh the transactional tables.
   b. Define source data profiles.
3. Log on to SAS Customer Link Analytics as a network analyst and define a project.
4. Complete the workflow steps that are listed here:
   a. Select the nodes and links whose data you want to analyze and then extract summarized transaction data.
   b. Filter the links based on specific parameters and assign weights to the links.
   c. Build communities by selecting the appropriate analytical approach.
   d. Select centrality measures that you want to compute and provide input parameters to compute these measures. You can also generate community segments and view the community segmentation report.
   e. Assign a role to each node of the communities.
5. Promote a project to batch mode.
6. Log on to SAS Customer Link Analytics as a network analyst, and complete the following tasks:

a Enrich the output data of a project.

b Load the node-level enriched data into the SAS Customer Link Analytics LASR Analytic Server.

Note: You can also perform this step if you log on as a business user.

c Create and view the community report.

Note: You can also perform this step if you log on as a business user.

7 (Optional) Perform viral effect analysis.

Distributed versus Non-Distributed Deployments

SAS Customer Link Analytics can run on a computer grid or on a single computer system with multiple CPUs. Running on a computer grid is referred to as a *distributed mode* of processing. Running on a single computer system with multiple CPUs is referred to as a *non-distributed mode* of processing. For details, see "Modes of Processing" on page 35.

Overview of SAS Customer Link Analytics Architecture

The SAS Customer Link Analytics architecture is designed to efficiently process large volumes of network and link data and produce results such as communities and roles. The architecture enables the solution to use this data to support user-driven workflows through the application user interface (UI). SAS Customer Link Analytics has a multi-tier architecture that separates the workflow-related activities from data-intensive process routines and distributes functionality across computer resources that are most suitable for these tasks. SAS Customer Link Analytics uses the capability of SAS High-Performance Analytics to maximize performance. You can scale the architecture to meet the demands of your workload. For a large organization, the tiers can be installed across many machines with different operating systems. For tasks such as developing prototypes and presenting demonstrations, all the tiers can be installed on a single machine. Similarly, if you are implementing SAS Customer Link Analytics for small enterprises, then you can install all the tiers on a single machine.

The SAS Customer Link Analytics architecture consists of the following four tiers:

Data Tier
: The data tier stores application data (also called *configuration data*) and business data (also called *transactional data*). The application data is stored in a PostgreSQL database. However, business data can reside in SAS, Teradata, or Hadoop based on the deployment setup at your implementation site. Access to the business data that is used for processing is managed using the appropriate SAS/ACCESS engine.

Server Tier

The SAS Customer Link Analytics middle tier invokes the SAS stored procedures that are a part of SAS Customer Link Analytics Server. These stored procedures perform analytical and data processing depending on certain user-specified parameters. The configuration and execution parameters are stored in the application data in the PostgreSQL Server. These processes access the underlying business data through the appropriate SAS/ACCESS engine. If SAS Customer Link Analytics operates with high-performance capabilities, then SAS High-Performance Analytics procedures are used to process the data. As a result, there is greater improvement in performance. Customers who have high volumes of data and a tight service-level agreement (SLA) should consider this version of SAS Customer Link Analytics.

SAS Customer Link Analytics uses the SAS Customer Link Analytics LASR Analytic Server for executing analytical procedures alongside LASR in the high-performance offering in which the business data is stored in Hadoop. SAS Customer Link Analytics renders SAS Visual Analytics based reports and loads the final business data output to the SAS Customer Link Analytics LASR Analytic Server. After processing is complete, the resulting business data is saved in the data tier, whereas the parameters and status flags are saved in the application data. The SAS Metadata Server is used to access certain configuration properties such as library definitions and log paths. Also, the business data output of certain processes is registered in SAS metadata. SAS/CONNECT is used to spawn multiple SAS sessions when certain data processing must run in parallel.

Middle Tier

The middle tier of SAS Customer Link Analytics provides an environment in which the SAS Customer Link Analytics client, along with other business intelligence web applications, can execute in an integrated environment. These applications run in a web application server and communicate with the user by sending and receiving data from the user's web browser. The middle tier of SAS Customer Link Analytics uses the SAS web infrastructure platform. Most of the platform services, such as services for executing the stored procedures and for interfacing with SAS Management Console, are deployed on this platform. SAS Customer Link Analytics also indirectly communicates with the SAS Visual Analytics middle tier. This communication is triggered when SAS Customer Link Analytics makes a request to SAS Visual Analytics Viewer on the client side to fetch a report. In addition, the middle-tier applications depend on the servers that are deployed on the server tier to process, query, and analyze data.

Client Tier

The SAS Customer Link Analytics web interface is a Flex—based UI that provides capabilities for various user roles. This interface accepts processing parameters from the user and invokes the underlying APIs from the middle tier. The predefined 2G reports that SAS Customer Link Analytics produces can be viewed in the SAS Visual Analytics Viewer. Switching between SAS Customer Link Analytics and the SAS Visual Analytics Viewer is enabled using the application switcher.

Figure 1.3 SAS Customer Link Analytics Architecture: Non-Distributed Mode

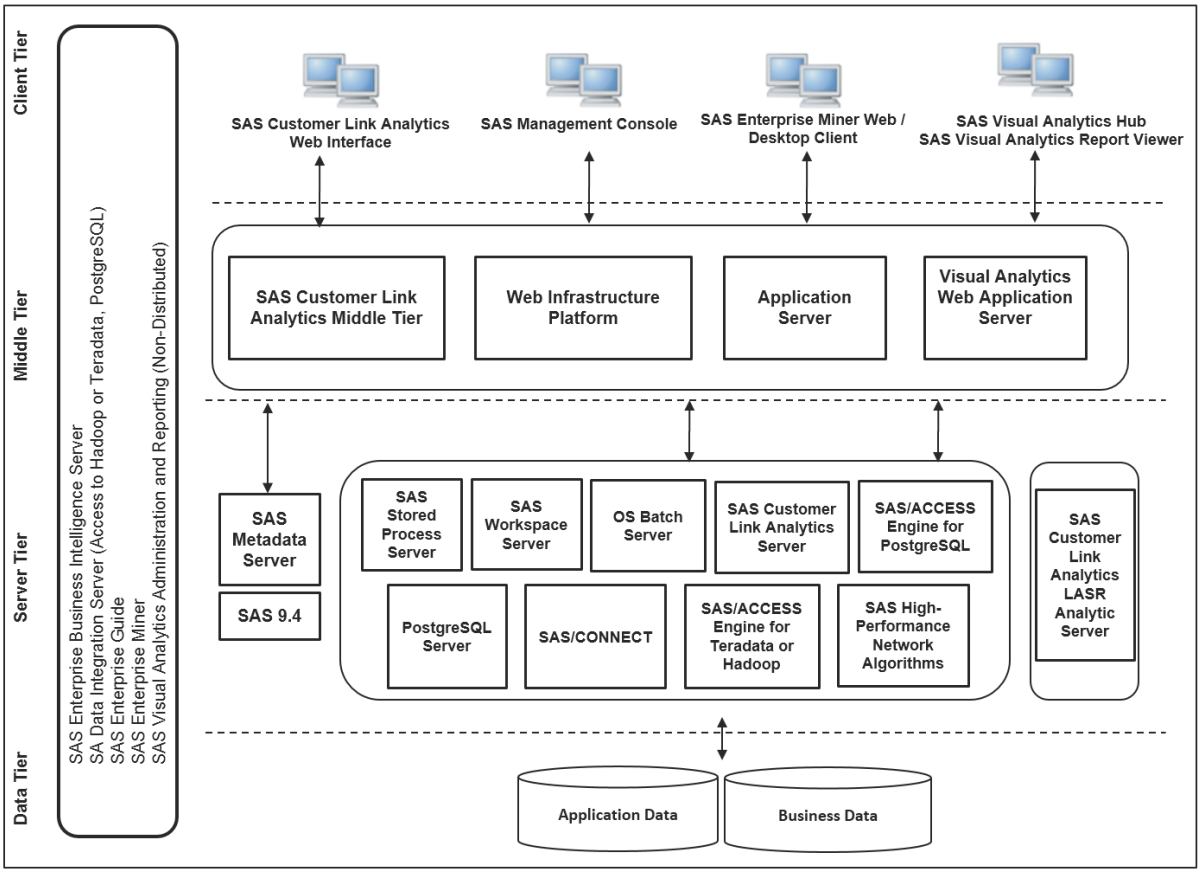
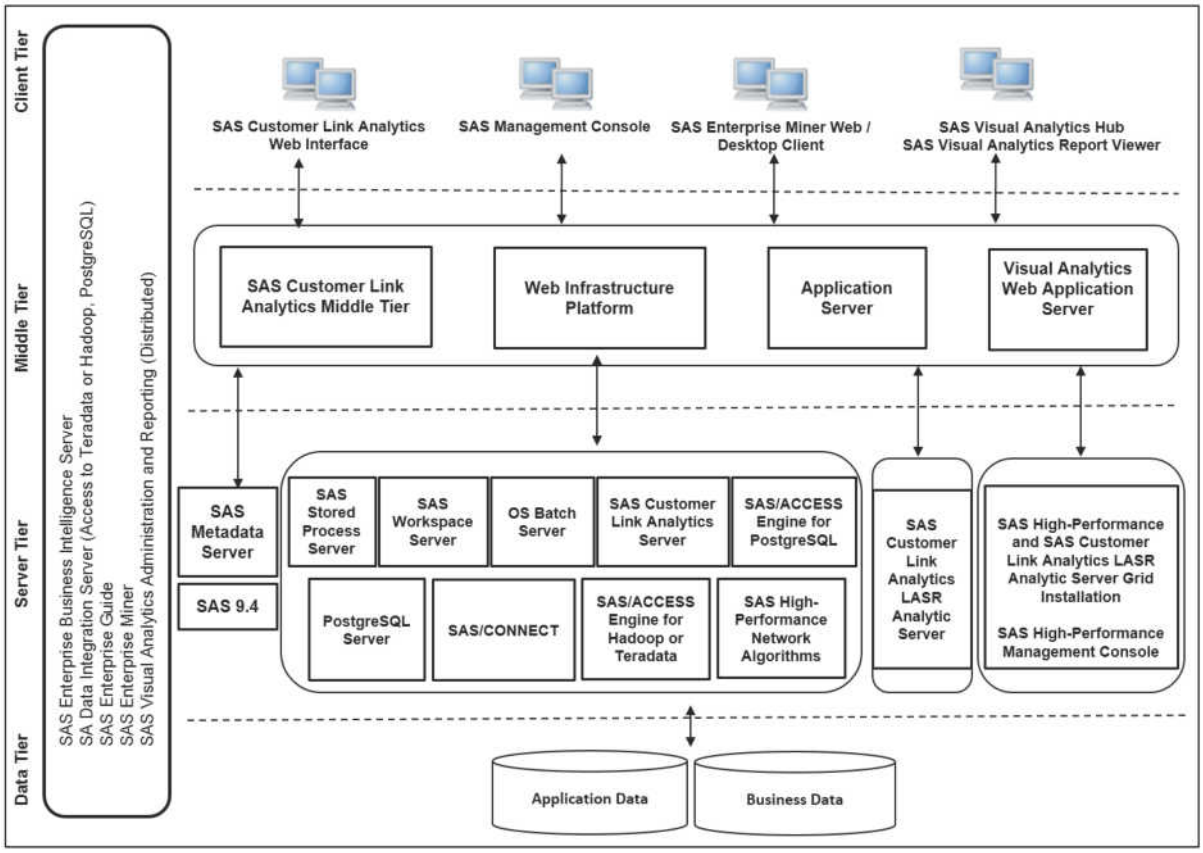


Figure 1.4 *SAS Customer Link Analytics Architecture: Distributed Mode*



2

Pre-Installation Instructions

Verify System Requirements

Review the system requirements documentation to ensure that your system meets the appropriate requirements. For more information, see *System Requirements for SAS Customer Link Analytics*. You can access the documentation from the locations listed here:

- http://support.sas.com/documentation/installcenter/en/iksocnfluatofrsr/69502/HTML/default/index.html
- http://support.sas.com/documentation/installcenter/en/ikhpclaofrsr/69490/HTML/default/index.html

Obtain a Deployment Plan

Before you can install SAS Customer Link Analytics, you must obtain a deployment plan. The *deployment plan* is a summary of the software that is installed and configured during your installation. A deployment plan file, named plan.xml, contains information about what software should be installed and configured on each machine in your environment. This plan serves as input to the SAS installation and configuration tools. SAS includes a standard deployment plan. You can use this standard plan or create your own plan. For more information, see “About Deployment Plans” in *SAS Intelligence Platform: Installation and Configuration Guide*, which is located at http://support.sas.com/documentation/onlinedoc/intellplatform.

Create a SAS Software Depot

Download the software that is listed in your SAS software order with SAS Download Manager. A SAS Software Depot is created, which includes the SAS installation data (SID) file. The SID file is used by SAS to install and license SAS software. It is a control file that contains license information that is required to install SAS. After you have downloaded SAS Software Depot, you can then use SAS Deployment Wizard to install your software. Verify that Base SAS is listed as a selected product. Then, select additional products specific to your environment. For more information, see “Creating a SAS Software Depot” in *SAS Intelligence Platform: Installation and Configuration Guide* at http://support.sas.com/documentation/onlinedoc/intellplatform.

Install and Configure the SAS High-Performance Analytics Environment

In distributed mode, you need to use the SAS High-Performance Analytics environment component of the SAS High-Performance Analytics infrastructure to install and configure components on machines in the grid network. For deployment instructions, see *SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide*, which is available at the following location: http://support.sas.com/documentation/solutions/hpainfrastructure/.

Set UNIX Directory Permissions

Note: This is a new permissions requirement that is introduced in SAS 9.4.

To deploy SAS Customer Link Analytics in UNIX environments, you must create and grant WRITE permissions on the `/etc/opt/vmware/vfabric` directory. Refer to the SAS Pre-Installation Checklist that is included with your deployment plan for instructions about how to set up this directory.

Creating and Verifying SSH Keys

You must create Secure Shell (SSH) keys if you are working with SAS Customer Link Analytics in a distributed computing environment and have not opted for Kerberos authentication. You do not need SSH keys if you are working with SAS Customer Link Analytics in a non-distributed computing environment.

SAS Customer Link Analytics uses passwordless SSH for access to the machines in the grid network and to the server tier.

Each SAS Customer Link Analytics user requires an SSH key pair for authentication with the grid network. SSH keys must be established for any user who will be running SAS jobs on the grid. For example, users who will be logging on to the server and running SAS code to create and run their projects require SSH keys.

You can use one of two methods to set up SSH keys for these users:

- Create a separate account and SSH key pair for each SAS Customer Link Analytics user. Each user can create SAS Customer Link Analytics projects on the grid. Users can also create and execute projects using the UI.

  This method requires that you set up SSH keys for all SAS Customer Link Analytics users on each grid node. Each user's SSH credentials are used for authentication with the grid nodes in order to provide traceability of user sessions to individual user accounts. This method is the safest and is recommended, though it is more difficult to implement.

- Create SSH keys and one generic account for the user associated with the account. This dedicated or generic user must be registered with the SAS Customer Link Analytics workspace server. All jobs on the grid use this account after the user has successfully been authenticated to the UI by using the credentials in the metadata server. This method is less secure than creating a separate account for each user, but it is simpler to implement.

  Note: You can use the SAS High-Performance Management Console that is available as a component of the SAS High-Performance Analytics infrastructure to create new users and set up SSH key authentication for the users.

Prerequisite Setup for Teradata

Install the Teradata Client

If the business data that you import into SAS Customer Link Analytics is stored in Teradata, make sure that you install and configure the Teradata client on a machine on which the SAS Customer Link Analytics Server Tier will be installed and configured. Contact your database administrator to set up the client software. Also, make sure that the required databases are created on the Teradata server.

Create a Super User

Create a user on the Teradata server. This user is a super user who will perform all the operations that are relevant for SAS Customer Link Analytics. In addition, this user will own all the databases that you will create.

Create Databases for Business Data

Create appropriate databases on the Teradata server. For example, you can create the following databases:

Table 2.1 *Teradata Databases for Business Data*

Libref	Metadata Library Name	Schema Name
Sia_bdop	Sia_bdm_output	Sia_bdop
Sia_bdim	Sia_bdm_intmdt	Sia_bdim
Sia_anop	Sia_analytics_output	Sia_anop
Sia_ani	Sia_analytics_inmdt	Sia_ani

During SAS Deployment Wizard installation, you are prompted to specify a schema name for each of these libraries. Default values are provided for these prompts as mentioned in this table. However, you can change these values according to your planned database setup.

Grant Privileges to the Super User

By using the Teradata client, you grant the following permissions to the super user. Contact your database administrator for assistance.

```
GRANT ALL ON <Database name> to <Super user name>
```

In this command, replace <*Database name*> with the schema name that is mentioned in Table 2.1 on page 16. Also, replace <*Super user name*> with the appropriate user name that you created earlier. For details, see “Create a Super User” on page 16.

For example, for the Sia_bdop schema and the clauser super user, enter the following command:

```
GRANT ALL ON Sia_bdop to clauser;
```

Note: Make sure that you grant all permissions to the super user for each database that is listed in Table 2.1 on page 16.

Export Environment Variables

Export environment variables for the Teradata client according to your platform.

Table 2.2 *Environment Variables*

Platform	Environment Variables
Linux for Intel Architecture, Linux for x64, and Solaris for x64	LD_LIBRARY_PATH=TPT-API-LIBRARY-LOCATION NLSPATH=TPT-API-MESSAGE-CATALOG-LOCATION

Specifying Database Names

Various Teradata libraries are created during SAS Deployment Wizard installation. You are prompted to specify a database name for each of these libraries. Default values are provided for these prompts. However, you can change these values according to your planned database setup.

Note: It is not mandatory that these databases be available during SAS Deployment Wizard installation.

Prerequisite Setup for Hadoop (Hive) and Kerberos Authentication

Complete all the tasks that are explained in Chapter 5, Administrator's Guide for Hadoop, of *SAS 9.4 In-Database Products: Administrator's Guide*. This guide is available at the following location: http://support.sas.com/documentation/onlinedoc/indbtech/index.html.

Prerequisite Setup for SAS Embedded Process

Complete all the tasks that are required for installing and configuring the SAS Embedded Process. To do so, refer to *SAS 9.4 In-Database Products: Administrator's Guide*. This guide is available at the following location: http://support.sas.com/documentation/onlinedoc/indbtech/index.html. Refer to Chapter 5, Administrator's Guide for Hadoop, if your business data is stored in Hadoop (Hive). Refer to Chapter 10, Administrator's Guide for Teradata, if your business data is stored in Teradata.

Default File Locations

SAS Deployment Wizard installs and configures your SAS software. The application installation files are installed in a default location referred to as `<SAS Home>`. For example, on a Windows machine, `<SAS Home>` is `C:/Program Files/SASHome`.

The following table lists the default locations of the installation and configuration files for SAS Customer Link Analytics.

Table 2.3 *Default File Locations*

Location Name	Windows Path	UNIX Path
`<SAS Home>`	`C:/Program Files/SASHome`	`/usr/local/SASHome`
`<SAS configuration directory>`	`C:/SAS/Config`	`/usr/local/config`

3

Installation Instructions

Overview of Installing SAS Customer Link Analytics

There are several concepts to understand and components to manage when you install SAS Customer Link Analytics, including the following:

- understanding distributed deployments versus non-distributed deployments
- the SAS High-Performance Analytics environment

General information about using SAS Deployment Wizard to install SAS software components that are specified in your deployment plan is documented in *SAS Intelligence Platform: Installation and Configuration Guide*, which is available at the following location: http://support.sas.com/documentation/onlinedoc/intellplatform/index.html. Review this information before you install SAS Customer Link Analytics.

Installing SAS Customer Link Analytics

Follow the instructions in the SAS Intelligence Platform documentation to install SAS Customer Link Analytics. Many prompts of SAS Deployment Wizard are specific to SAS Intelligence Platform and other SAS solutions, and information about these prompts is beyond the scope of this guide. For instructions about installing SAS Intelligence Platform, see “Installing and Configuring Your SAS Software” in *SAS Intelligence Platform: Installation and Configuration Guide*.

You will encounter all the prompts that are specific to SAS Customer Link Analytics during the deployment if you select the **Typical** or **Custom** mode of installation. However, if you choose the **Express** mode, you will encounter only those prompts that do not have a default value. For the rest of the prompts, the installation proceeds with the default value that is set for these prompts. In this case, you cannot configure the prompt values according to your requirement. For example, the default value for the **Database Type** prompt is **SAS**. Unless you choose the **Typical** or **Custom** mode, you will not be prompted to choose the other database options such as **Hadoop** or **Teradata**.

TIP On the wizard pages of the SAS Deployment Wizard, click **Help** to view the details of any prompt. This information helps you enter the correct value for the prompt.

4

Post-Installation Instructions

Overview of Post-Installation Tasks

At the end of the installation process, SAS Deployment Wizard produces an HTML file, named Instructions.html. To complete your installation, you need the information that is provided in the Instructions.html file. In addition, you need the information that is specific to SAS Customer Link Analytics, which is documented in this chapter.

This chapter provides details about how to complete the SAS Customer Link Analytics post-installation tasks.

Update SAS Scripts to Grant Permission to User Groups in UNIX Environments

Using the umask option, you can grant permission to users on a conditional basis if the user is part of the SAS Customer Link Analytics user group.

Note: This example might require changes to fit your server configuration. In particular, this example might result in changed permissions on other SAS files.

To set these permissions:

1. On each SAS Workspace Server, open **/sasconfigdir/Lev1/SASApp/appservercontext_env.sh**.
2. Enter the configuration information for your operating environment. Here is the general format of this code:

   Note: The following code uses grave accents, not quotation marks.

   ```
   CMD=<your-operating-system-path>
   CURR_GID=`eval $CMD -g`
   GID=<solution-group-id>
   if [$CURR_GID -eq $GID]; then umask 002 fi
   ```

   a. In the CMD=<*your-operating-system-path*>, specify the full path on your server where the ID command is stored. You can get this information by entering a `which id` or `whence id` command on your console.

   b. In the GID=<*solution-group-id*>, specify the group ID. Type `id` on your console to get the GID and UID information.

   c. A value of 002 is recommended for the umask option.

   Here is the code example for the LNX (Linux) environment:

   ```
   #!/bin/bash
   CMD=/usr/bin/id
   CURR_GID=‘eval $CMD -g‘
   GID=500
   if [ "$CURR_GID" -eq "$GID" ] ; then umask 002 fi
   ```

Creating User Groups and the Metadata User in SAS Management Console

Create Users and Assign Groups and Roles

You need to create a single user who can access the business data and the application data and log on to SAS Customer Link Analytics. To do so, you have

to create a user who is a member of the default group and the default roles that SAS Deployment Wizard creates.

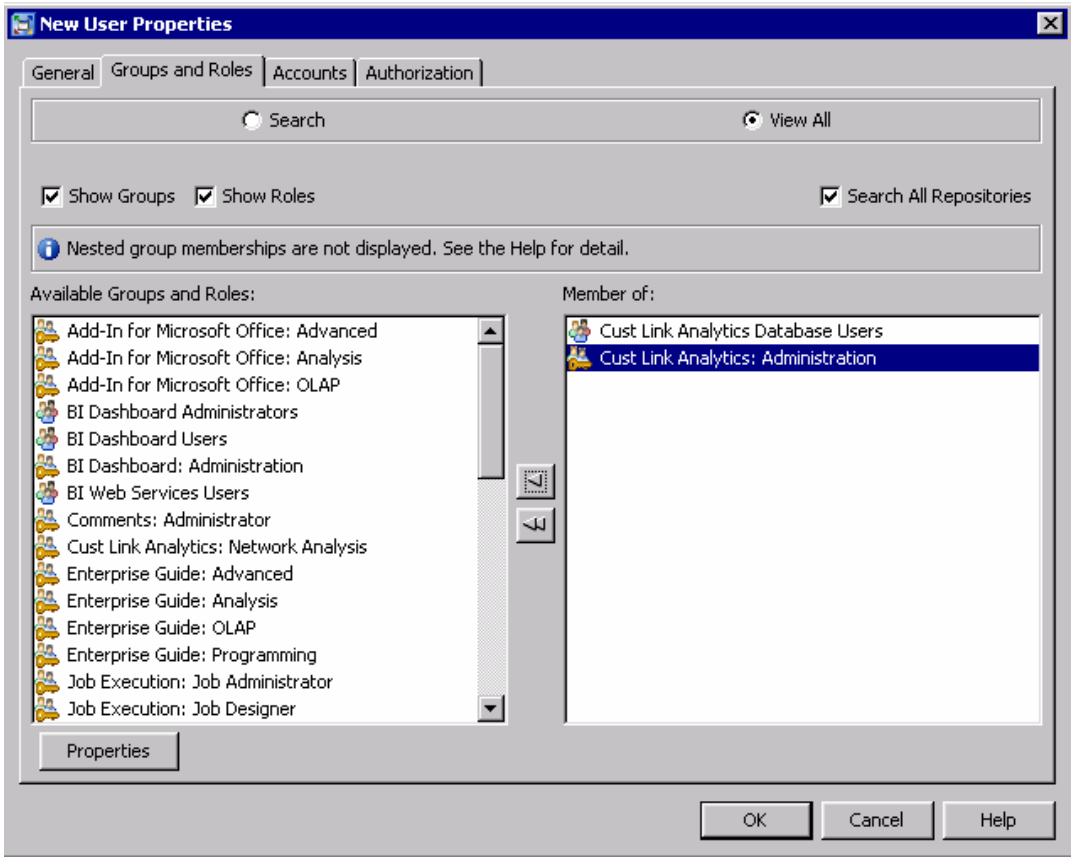
To configure a SAS Customer Link Analytics metadata user account:

1. Start SAS Management Console and connect as a SAS administrator (for example, sasadm@saspw).
2. Select the **Plug-ins** tab.
3. Right-click the **User Manager** plug-in, and then select **New** ▸ **User** from the pop-up menu. The New User Properties dialog box appears.
4. On the **General** tab, enter the user name. For example, you can enter the name as **CLAUSER**.
5. On the **Group and Roles** tab, add the group and roles depending on the type of role that you want to define:

Table 4.1 *Roles and Capabilities*

Type of Role	Capabilities	Groups and Roles
Administrator	Provides capabilities to view, create, and delete source data profiles and tables.	■ Cust Link Analytics Database Users ■ Cust Link Analytics: Administration
Network analyst	Provides capabilities to view and create projects, enrich project's data, load data to the SAS Customer Link Analytics LASR Analytic Server, and create and view community report.	■ Cust Link Analytics Database Users ■ Cust Link Analytics: Network Analysis
Business user	Provides capabilities to view all projects, load data to the SAS Customer Link Analytics LASR Analytic Server, and create and view community report.	■ Cust Link Analytics Database Users ■ Cust Link Analytics: Business User

Figure 4.1 New User Properties



6 Click **OK**.

7 Make sure that the **Visual Analytics: Analysis** role is assigned to the **Cust Link Analytics Database Users** group. If is not assigned, you have to add this role to the user group.

8 Close SAS Management Console.

Create Login Accounts for IWA Setup

Perform this task if SAS Customer Link Analytics is deployed on an Integrated Windows Authentication (IWA) setup. For each metadata user that you create, you have to define two login accounts.

To create login accounts for a metadata user:

1 In SAS Management Console, create a new metadata user or open the properties of an existing one. For example, you can view the properties of the CLAUSER that you created earlier. For more information, see step 4 of "Create Users and Assign Groups and Roles" on page 22.

2 On the **Accounts** tab, click **New**. The New Login Properties window appears.

3 Create two login accounts for the user ID that you defined in step 1.

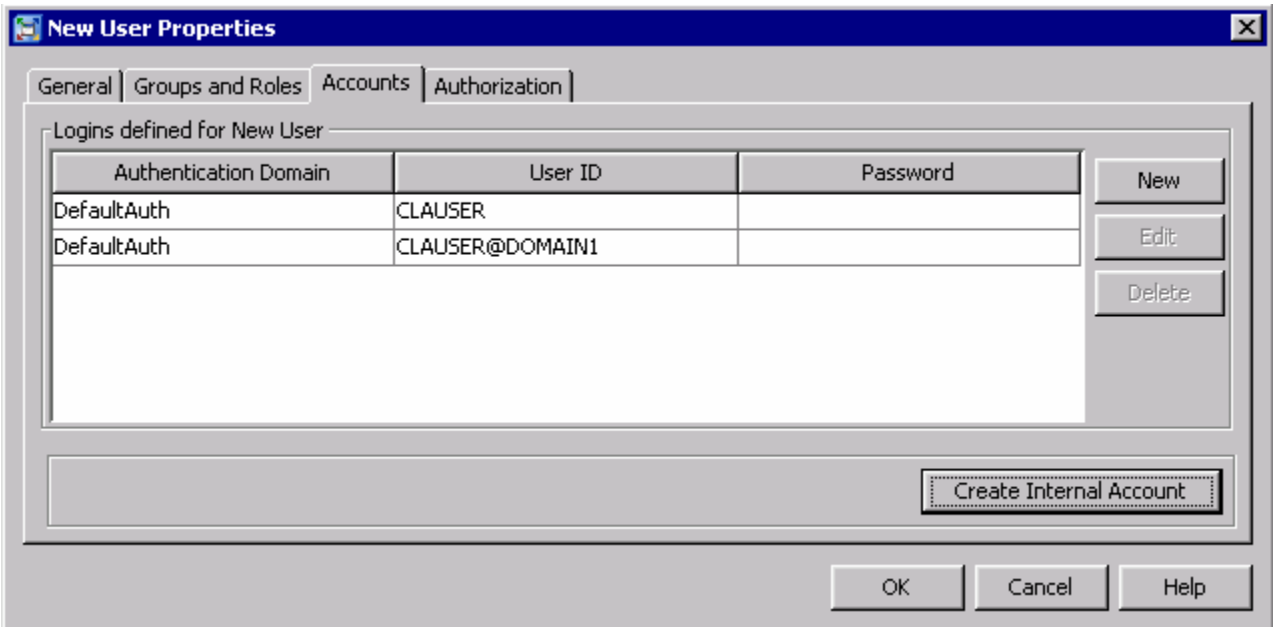
Create the login IDs in the following formats:

- *<User ID>*
- *<User ID>@<domain>*

For example, for the CLAUSER user ID, you can define the following login accounts:

- CLAUSER
- CLAUSER@CLADOMAIN1

Figure 4.2 *New User Properties Window*



Note: A warning message is displayed if you create both accounts with the same authentication domain. Click **Yes** to confirm your action.

4 Click **OK**.

5 Close SAS Management Console.

6 From the **Start** menu, select **Administrative Tools** ▸ **Local Security Policy**.

7 In the left pane, expand **Local Policies** ▸ **User Rights Assignment**. In the right pane, right-click **Log on as a batch job**, and select **Properties**.

8 Click **Add User or Group** and add the metadata user for which you created the two login accounts.

9 Click **OK**.

Starting the SAS Customer Link Analytics LASR Analytic Server

A metadata user who has administrative rights must start the SAS Customer Link Analytics LASR Analytic Server, the server that is created for SAS

Customer Link Analytics. This user must be assigned to the following roles or groups:

- Visual Analytics Data Administrator
- Visual Analytics: Administration
- Visual Data Builder Administrators

You can use the SAS Visual Analytics Administrator to start the server instance. To start the SAS Customer Link Analytics LASR Analytic Server, you need to know the server's host name. The host name is the value that you enter for the **SAS Customer Link Analytics LASR Analytic Server host name** prompt when you install SAS Customer Link Analytics.

Make sure that you start the SAS Customer Link Analytics LASR Analytic Server before you start creating projects and run workflow steps. Otherwise, you will not be able to perform any reporting tasks in a non-Hadoop environment and both reporting and analytical tasks in the Hadoop (Hive) environment.

Deploy the Loop Job for Data Enrichment

To control the execution of the data enrichment process, you have to deploy the loop job, sialoopjob.

To deploy the loop job, complete the following steps:

1. Connect to SAS Data Integration Studio with administrative privileges.
2. On the **Folders** tab, expand **Products** ▸ **SAS Customer Link Analytics** ▸ **Cust Link Analytics 6.5** ▸ **Jobs**.
3. Right-click **sialoopjob**, and then select **Deploy** from the **Scheduling** menu. The Deploy a Job for Scheduling window appears.
4. Specify the **Deployment Directory** and **Deployed Job Name** according to the default value that is set for the corresponding software component properties, clasvrc.loop.job.location and clasvrc.loop.job.name. For more information, see Table 6.5 on page 52.

   Note: It is recommended that you specify the values according to the default value that is set up for the software component properties. However, if you specify other values, then make sure that you also change the value of the software component properties. For more information, see "View or Modify Software Component Properties" on page 51.
5. Click **OK**.

Deploy the Loop Job for Viral Effect Analysis

To control the execution of generating ABT variables for viral effect analysis, you have to deploy the loop job, siascenloopjob.

To deploy the loop job, complete the following steps:

1 Connect to SAS Data Integration Studio with administrative privileges.

2 On the **Folders** tab, expand **Products** ▸ **SAS Customer Link Analytics** ▸ **Cust Link Analytics 6.5** ▸ **Jobs**.

3 Right-click **siascenloopjob**, and then select **Deploy** from the **Scheduling** menu. The Deploy a Job for Scheduling window appears.

4 Specify the **Deployment Directory** and **Deployed Job Name** according to the default value that is set for the corresponding software component properties, clasvrc.abt.loop.job.location and clasvrc.abt.loop.job.name. For more information, see Table 6.5 on page 52.

Note: It is recommended that you specify the values according to the default value that is set up for the software component properties. However, if you specify other values, then make sure that you also change the value of the software component properties. For more information, see "View or Modify Software Component Properties" on page 51.

5 Click **OK**. By default, the siascenloopjob job runs all the ABT-building processes concurrently. To run these processes sequentially, see "Run ABT-Building Processes Sequentially" on page 64.

Configure the SAS Server for SAS Enterprise Miner Project Location

If the SAS Customer Link Analytics Server is not deployed on the default application server, SASApp - Logical Workspace Server, then you must set up the appropriate SAS Server for the SAS Enterprise Miner project location parameter.

To set up the SAS Server for the SAS Enterprise Miner project location parameter:

1 Connect to SAS Management Console with administrative privileges.

2 On the **Folders** tab, expand **Products** ▸ **SAS Customer Link Analytics** ▸ **Cust Link Analytics 6.5** ▸ **Viral Effect Analysis**.

3 In the right pane, right-click **sia_stp_update_scenario_param** and select **Properties**.

4 On the **Execution** tab, note the value of **Application server**.

5 On the **Parameters** tab, select **SAS Enterprise Miner project location**, and click **Edit**.

6 On the **Prompt Type and Values** tab, from the **SAS Server** list, select the same value that you noted for **Application server** in step 4.

7 Click **OK** to save the changes.

Sample Reporting Templates and the LASR Table

Metadata Locations

To create the community report and the community segmentation report, SAS Customer Link Analytics provides you ready-to-use reporting templates for the English (en) locale. When you install SAS Customer Link Analytics, these templates are registered in the following SAS metadata location: `/Products/SAS Customer Link Analytics/Cust Link Analytics 6.5/Sample Reports`.

In addition, these templates use the CLA_NODE_SAMPLE_LASR_DATA LASR table. This table is registered in the following metadata location: `/Products/SAS Customer Link Analytics/Cust Link Analytics 6.5/Data Sources/LASR`.

Table 4.2 *Reporting Templates*

Template Filename	Purpose
CLA_sample_default_node_level_rpt	Generates a community report for the following node-level enrichment categories: ■ **Roles and communities**
CLA_sample_derived_indicator_node_level_rpt	Generates a community report for the following node-level enrichment categories: ■ **Roles and communities** ■ **Churn and acquisition indicators**
CLA_sample_acquisition_churn_node_level_rpt	Generates a community report for the following node-level enrichment categories: ■ **Roles and communities** ■ **Relation with churned and acquired nodes**
CLA_sample_acquisition_churn_and_derived_node_level_rpt	Generates a community report for the following node-level enrichment categories: ■ **Roles and communities** ■ **Churn and acquisition indicators** ■ **Relation with churned and acquired nodes**
CLA_sample_community_segmentation_rpt	Generates a community segmentation report.

For more information about the community report and the community segmentation report, see the *SAS Customer Link Analytics: User's Guide*.

Make sure that you do not modify the reporting templates for any purpose other than applying localizations. Also, make sure that you do not modify the LASR table. Otherwise, you will not be able to generate the community report.

Localizing the Sample Reporting Templates

You might want to localize your reports according to other browser locales such as French (fr) or German (de). In this case, you need to apply localizations to the reporting templates using SAS Visual Analytics Designer. For more information about how to localize a report, see the *One Report, Many Languages: Using SAS Visual Analytics 7.1 to Localize Your Reports* technical paper that is available at the following location: http://support.sas.com/documentation/onlinedoc/va/7.1/LocalizeReports.pdf.

Set Up the Secure Attribute for Session Cookies

Perform this task if SAS Web Server is configured by SAS Deployment Wizard to support the HTTPS protocol.

The secure attribute for cookies directs a web browser to only send cookies through an encrypted HTTPS connection.

To configure the SAS Web Application Server to return the session ID with the secure attribute, complete the following steps:

1. Open the server.xml file. This file is available in the following location: **<*SAS configuration directory*>/Lev1/Web/WebAppServer/SASServer<*n*>_<*m*>/conf**.
2. Add `secure="true"` to the existing `Connector` element.
3. Save the file.
4. Restart SAS Web Application Server.

Verifying Values of WORK, MEMSIZE, and SORTSIZE Options

If you want to work with SAS Customer Link Analytics in non-distributed mode, you must verify the value of certain SAS system options. To do so, open the sasv9.cfg file, which is available in the following location: **<*SAS Home*>/SASFoundation/9.4/nls/en**. Make sure that you specify an appropriate value for the following options:

WORK
: specify an appropriate value to ensure that enough space is available for the current SAS session.

MEMSIZE
specify an appropriate value for this option depending on the size of the data that a SAS Customer Link Analytics project will be processing.

SORTSIZE
specify an appropriate value for this option depending on the size of the data that a SAS Customer Link Analytics project will be processing.

If you do not specify appropriate values for these options, the Community Detection workflow step or the Centrality Measures Computation workflow step might fail to execute because of insufficient memory.

Unconfiguring SAS Customer Link Analytics

Prerequisite Tasks

Before you unconfigure SAS Customer Link Analytics, complete the following tasks:

1. Create a backup of the following folders:
   - `<SAS configuration directory>/Lev1/AppData/SASCustomerLinkAnalytics`
   - `<SAS configuration directory>/Lev1/SASCustomerLinkAnalyticsDataServer`
2. Create a backup of the data that is stored in the application data tables (sia_apdm).
3. Create a backup of the application metadata if you have made any customizations that you want to save for later use. To do so, complete the following steps:
   a. Start SAS Management Console, and then open the appropriate connection profile to connect to the desired metadata server.
   b. On the **Folders** tab, select **SAS Folders** ▸ **Products**.
   c. Create a backup of the following folder: **SAS Customer Link Analytics**.
   d. Similarly, create a backup of the **SAS Customer Link Analytics** folder that is available in the following location: **SAS Folders** ▸ **Shared Data**.
   e. Close SAS Management Console.

Remove SAS Customer Analytics for Communications

Use SAS Deployment Manager to remove the following software components of SAS Customer Link Analytics:

- **Cust Link Analytics 6.5**
- **Cust Link Analytics Svr Cfg 6.5**

- **SAS Customer Link Analytics Data Server 6.5**
- **Cust Link Anlytcs LASR 6.5**
- **Cust Link Anlytcs LASR Cfg 6.5**

Post-Unconfiguration Tasks

After you remove the software components of SAS Customer Link Analytics, delete the application metadata.

To delete the application metadata, complete the following tasks:

1. Start SAS Management Console, and then open the appropriate connection profile to connect to the desired metadata server.
2. On the **Folders** tab, select **SAS Folders** ▸ **Products**.
3. Delete the **SAS Customer Link Analytics** folder.
4. Similarly, delete the **SAS Customer Link Analytics** folder that is available in the following location: **SAS Folders** ▸ **Shared Data**.
5. Close SAS Management Console.
6. Delete the following folders:
   - `<SAS configuration directory>/Lev1/AppData/SASCustomerLinkAnalytics`
   - `<SAS configuration directory>/Lev1/SASCustomerLinkAnalyticsDataServer`
7. If you want to reconfigure SAS Customer Link Analytics, perform the following tasks:
   - Drop the Customer Link Analytics Data Server database (claapdm). You might also want to back up your data and restore it after the configuration is complete.
   - Drop the Customer Link Analytics Data Server login role (claadmin user).
   - Drop the Customer Link Analytics Data Server group role (claapdm_admin).

Part 2

Application Management

5

Modes of Execution

Modes of Processing

Overview

SAS Customer Link Analytics operates in distributed and non-distributed modes. In distributed mode, SAS Customer Link Analytics runs on a computer grid. However, in non-distributed mode, SAS Customer Link Analytics runs on a single computer system with multiple CPUs.

Non-Distributed Mode

In non-distributed mode, multiple processors share hardware resources such as disks and memory. They are controlled by a single operating system. The workload for a parallel job is distributed across the processors in the system.

In non-distributed mode, SAS Customer Link Analytics runs multiple concurrent threads on a multicore machine in order to take advantage of parallel execution on multiple processing units.

Distributed Mode

In distributed mode, many computers are physically housed in the same chassis. In a distributed environment, performance is improved because no resources must be shared among physical computers. However, a file system is commonly shared across the network. This configuration allows program files to be shared instead of installed on individual nodes in the system.

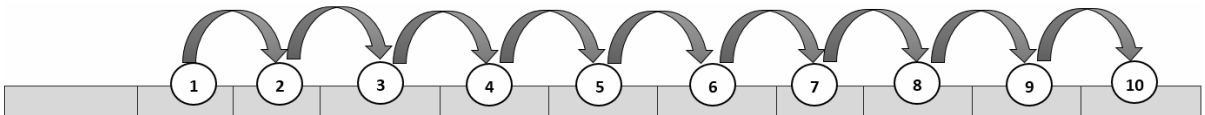
The analytical processes on the appliance are separate from the database processes. Therefore, the technique is referred to as *alongside-the-database* execution, in contrast to *in-database* execution, where the analytic code executes within the database process.

Data Flow for Mandatory Workflow Steps

Data Flow Diagram

The following diagram indicates how data flows in distributed and non-distributed modes for mandatory workflow steps. If you choose community segmentation, then the flow of data for its tasks differs.

Figure 5.1 *Data Flow for Mandatory Workflow Steps*



Process / Workflow Step	1 Project Creation and Data Extraction	2 Link and Node Filtering	3 Link Weight Computation	4 Copy Data	5 Community Building	6 Centrality Measures Computation	7 Copy Data	8 Role Assignment	9 Data Enrichment	10 Data Loading to LASR for Reporting Using HPDS2
Operating System	UNIX or Windows									
Business Data: In Hadoop (Hive)										
Distributed Mode (Symmetric / Asymmetric)	Hadoop (Hive)	Hadoop (Hive)	Hadoop (Hive)	Copy to SAS LASR Analytic Server	Alongside LASR	Alongside LASR	Copy to Hadoop (Hive)	Hadoop (Hive)	Hadoop (Hive)	Hadoop (Hive)
Business Data: In Teradata										
Distributed Mode (Symmetric / Asymmetric)	Teradata	Teradata	Teradata	-	Alongside-the-Database	Alongside-the-Database	-	Teradata	Teradata	Teradata
Non-distributed Mode	Teradata	Teradata	Teradata	Copy to SAS	SAS	SAS	Copy to Teradata	Teradata	Teradata	Teradata
Business Data: In SAS										
	SAS	SAS	SAS	-	SAS	SAS	-	SAS	SAS	SAS

Distributed Processing with Hadoop (Hive) as the Data Store

In distributed mode, SAS Customer Link Analytics can run with Hadoop (Hive) as a data store. In this mode, SAS Customer Link Analytics optimally leverages the features of the SAS High-Performance Analytics architecture and distributed processing.

If the source data is in Hadoop (Hive) and the library definition of the source library is in accordance with SAS In-Database processing rules, then the Data Extraction workflow step runs as an in-database process. As a result, data is not extracted from Hadoop (Hive) to SAS.

The Link and Node Processing workflow step also runs as an in-database process. However, the Community Detection and Centrality Measure Computation workflow steps use the SAS High-Performance Analytics architecture and run alongside LASR mode. In this case, data is moved from Hadoop (Hive) to the SAS Customer Link Analytics LASR Analytic Server using asymmetric mode. The analytic processes are then run alongside LASR mode. After the analytical processing is complete, data is moved back to Hadoop (Hive) using asymmetric mode.

The Role Assignment workflow step processes the data using SAS In-Database techniques. Therefore, there is no data movement between SAS and Hadoop (Hive).

After all the workflow steps have run successfully, the processes of data enrichment and data loading into the SAS Customer Link Analytics LASR Analytic Server are run to generate the community report. These processes are

run using SAS In-Database techniques. Therefore, for these processes, there also is no data movement between SAS and Hadoop (Hive).

Distributed Processing with Teradata as the Data Store

In distributed mode, SAS Customer Link Analytics can run also with Teradata as a data store. In this mode, SAS Customer Link Analytics optimally leverages the features of the SAS High-Performance Analytics architecture and distributed processing.

If the source data is in Teradata and the library definition of the source library is in accordance with the SAS In-Database processing rules, then the Data Extraction workflow step runs as an in-database process. As a result, data is not extracted from Teradata to SAS.

The Link and Node Processing workflow step also runs as an in-database process. However, the Community Detection and Centrality Measure Computation workflow steps use the SAS High-Performance Analytics architecture and run in alongside-the-database mode based on whether the SAS High-Performance Analytics grid configuration is symmetric or asymmetric. In this case, data movement between the SAS grid and Teradata is minimal and analytical procedures use the full potential of distributed computing.

The Role Assignment workflow step processes the data using the SAS In-Database techniques. Therefore, there is no data movement between SAS and Teradata.

After all the workflow steps are run successfully, the processes of data enrichment and data loading to the SAS Customer Link Analytics LASR Analytic Server are run to generate the community report. These processes are run using the SAS In-Database techniques. Therefore, for these processes also there is no data movement between SAS and Teradata.

Non-Distributed Processing with Teradata as Data Store

SAS Customer Link Analytics is configured to run with Teradata as a data store without using SAS High-Performance Analytics. In this case, SAS Customer Link Analytics processes data using in-database processing wherever possible. However, the analytical procedures are executed in symmetric multiprocessing (SMP) mode.

If the source data is in Teradata and the library definition of the source library is in accordance with the SAS In-Database processing rules, then the Data Extraction workflow step runs as an in-database process. Data is not extracted from Teradata to SAS.

The Link and Node Processing workflow step always runs as an in-database process in this configuration.

The Community Detection and Centrality Measure Computation workflow steps execute in the SAS server using SMP mode. In this case, data is extracted from Teradata only once at the beginning of the Community Detection workflow step. After all the analytical processes are complete, data is loaded back into Teradata.

The Role Assignment workflow step again processes the data using the SAS In-Database techniques and there is no data movement between SAS and Teradata.

After all the workflow steps are run successfully, the processes of data enrichment and data loading to the SAS Customer Link Analytics LASR Analytic Server are run to generate the community report. These processes are run using the SAS In-Database techniques. Therefore, for these processes also there is no data movement between SAS and Teradata.

Non-Distributed Processing with SAS as the Data Store

SAS Customer Link Analytics can be configured to run with SAS as a data store. In this case, SAS Customer Link Analytics executes both data processes and analytical procedures in SMP mode. Also, data is loaded to the SAS Customer Link Analytics LASR Analytic Server in SMP mode.

Graph Size Limitations for Non-Distributed Mode

In non-distributed mode, the maximum data (nodes or links) that SAS Customer Link Analytics can process is 2,147,483,647. If the graph contains entities beyond this value, then the processing fails. Therefore, it is recommended that you choose the distributed mode if you need to process high volumes of data.

Data Flow for Community Segmentation

Data Flow Diagram

The following diagram indicates how data flows in distributed and non-distributed modes for community segmentation.

Figure 5.2 Data Flow: Community Segmentation

Process	Design Mode 1 ABT Creation	2 Variable Reduction	3 Model Building	4 Scores Writeback	5 Report Data Creation	6 Loading Reporting Data to LASR	Batch Mode 1 Scoring ABT Creation	2 Copy Data	3 Score Code Application	4 Score Writeback	5 Report Data Creation	6 Loading Reporting Data to LASR
Operating System	UNIX or Windows											
Business Data: In Hadoop (Hive)												
Distributed Mode (Symmetric / Asymmetric)	Hadoop (Hive)	Alongside-the-Database	Alongside-the-Database	Hadoop (Hive)	Hadoop (Hive)	Hadoop (Hive)	Hadoop (Hive)	Copy to SAS	SAS	Hadoop (Hive)	Hadoop (Hive)	Hadoop (Hive)
Business Data: In Teradata												
Distributed Mode (Symmetric / Asymmetric)	Teradata	Alongside-the-Database	Alongside-the-Database	Teradata	Teradata	Teradata	Teradata	Copy to SAS	SAS	Teradata	Teradata	Teradata
Non-distributed Mode	Teradata	SAS	SAS	Teradata	Teradata	Teradata	Teradata	Copy to SAS	SAS	Teradata	Teradata	Teradata
Business Data: In SAS												
Non-distributed Mode	SAS	SAS	SAS	SAS	SAS	SAS	SAS	-	SAS	SAS	SAS	SAS

Distributed Processing with Hadoop (Hive) as the Data Store

In design mode, ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run as in-database processes. As a result, data is not extracted from Hadoop (Hive) to SAS, and these processes run using SAS In-Database techniques. However, the variable reduction and model building processes use the SAS High-Performance Analytics architecture and run in alongside-the-database mode. In this case, data movement between the SAS grid and Hadoop (Hive) is minimal and analytical procedures use the full potential of distributed computing.

In batch mode, scoring ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run as in-database processes. Therefore, for these processes, there also is no data movement between SAS and Hadoop (Hive). However, score code application executes in the SAS server. Therefore, data is extracted from Hadoop (Hive). After the analytical processing is complete, data is loaded back into Hadoop (Hive).

Distributed Processing with Teradata as the Data Store

In design mode, ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run as in-database processes. As a result, data is not extracted from Teradata to

SAS, and these processes are run using SAS In-Database techniques. However, the variable reduction and model building processes use the SAS High-Performance Analytics architecture and run in alongside-the-database mode. In this case, data movement between the SAS grid and Teradata is minimal and analytical procedures use the full potential of distributed computing.

In batch mode, scoring ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run using SAS In-Database techniques. Therefore, for these processes, there also is no data movement between SAS and Teradata. However, score code application executes in the SAS server. Therefore, data is extracted from Teradata and loaded into SAS. After the analytical processing is complete, data is loaded back into Teradata.

Non-Distributed Processing with Teradata as the Data Store

In design mode, ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run as in-database processes. As a result, data is not extracted from Teradata to SAS, and these processes are run using SAS In-Database techniques. However, the variable reduction and model building processes execute in the SAS server. Therefore, these processes extract the data from Teradata and load it into SAS. After the analytical processing is complete, the processes load the data back into Teradata.

In batch mode, scoring ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run using SAS In-Database techniques. Therefore, for these processes, there also is no data movement between SAS and Teradata. However, score code application executes in the SAS server. Therefore, data is extracted from Teradata and loaded into SAS. After the analytical processing is complete, data is loaded back into Teradata.

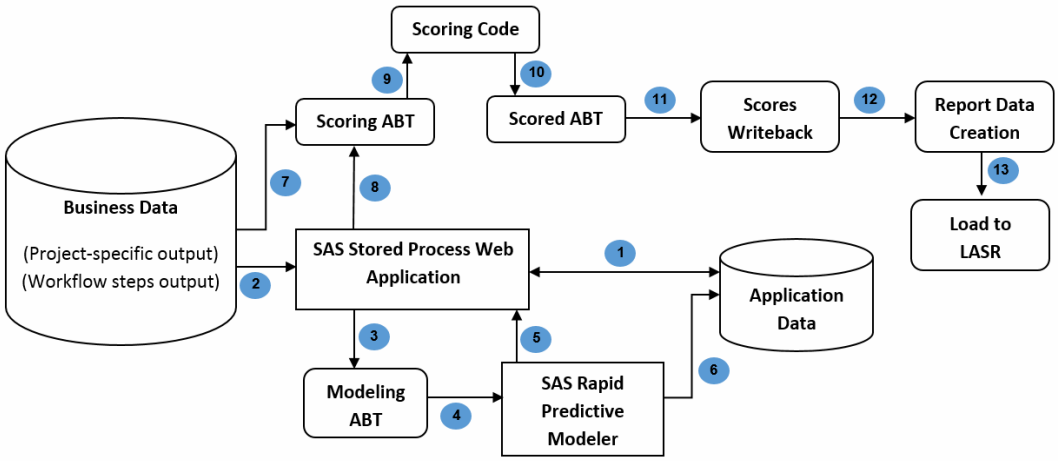
Non-Distributed Processing with SAS as the Data Store

For both design and batch modes, SAS Customer Link Analytics executes all processes in SAS and in SMP mode.

Data Flow for Viral Effect Analysis

In SAS Customer Link Analytics, the viral effect analysis functionality is provided through a set of stored processes. The data flow of viral effect analysis includes the following steps:

Figure 5.3 Data Flow for Viral Effect Analysis



1 When you create a scenario for a project by using a stored process, configuration information of that scenario is stored in the application data tables.

2 SAS Stored Process Web Application uses the output data of projects, configuration data from the application data tables, and source data.

3 Using this information, SAS Stored Process Web Application builds the modeling analytical base table (ABT).

4 The modeling ABT is provided as an input data set for SAS Rapid Predictive Modeler.

5 SAS Rapid Predictive Modeler builds a predictive model and creates a workspace for the SAS Enterprise Miner project.

6 The model information is captured and stored in the application data tables.

7 SAS Stored Process Web Application uses the output data of projects, configuration data from the application data tables, and source data.

8 Using this information, SAS Stored Process Web Application builds the scoring ABT.

9 The scoring ABT is provided as an input to the scoring process.

10 The scored ABT is generated as a result of the scoring process.

11 The scores are written back to the analytics output library.

12 The reporting data is generated.

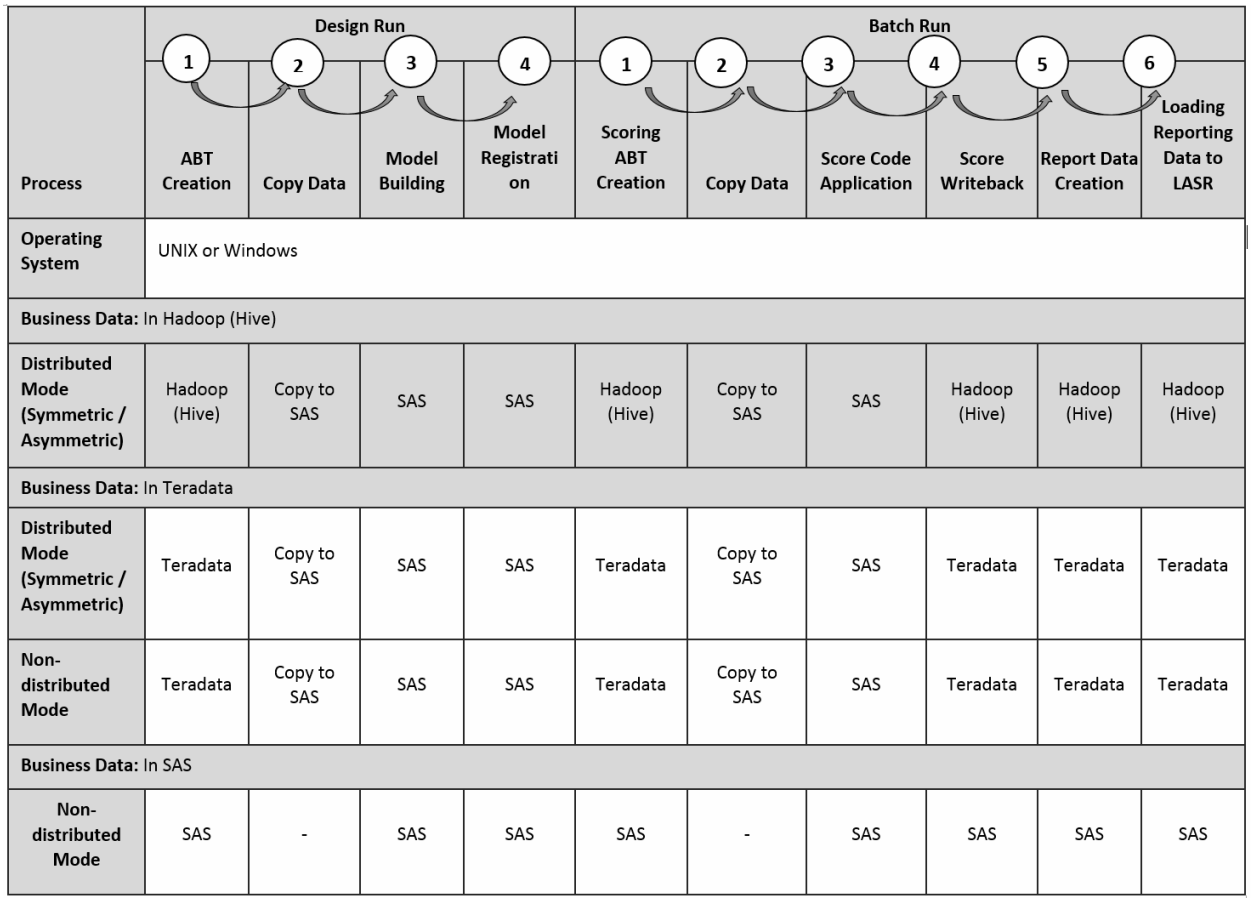
13 The reporting data is uploaded to the SAS Customer Link Analytics LASR Analytic Server.

Data Flow for Viral Effect Analysis

Data Flow Diagram

The following diagram indicates how data flows in distributed and non-distributed modes for viral effect analysis.

Figure 5.4 *Data Flow: Viral Effect Analysis*



Distributed Processing with Hadoop (Hive) as the Data Store

In design run, ABT creation runs as an in-database process. As a result, data is not extracted from Hadoop (Hive) to SAS. However, the model building and model registration processes run in the SAS server. In this case, data is moved from Hadoop (Hive) to SAS.

In batch mode, scoring ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run using SAS In-Database techniques. Therefore, for these processes, there also is no data movement between SAS and Hadoop (Hive). However, score code

application executes in the SAS server. Therefore, data is extracted from Hadoop (Hive). After the analytical processing is complete, data is loaded back into Hadoop (Hive).

Distributed and Non-Distributed Processing with Teradata as the Data Store

In design run, ABT creation runs as an in-database process. As a result, data is not extracted from Teradata to SAS. However, the model building and model registration processes run in the SAS server. In this case, data is moved from Teradata to SAS.

In batch mode, scoring ABT creation, scores writeback, report data creation, and loading reporting data to SAS LASR Analytic Customer Link Analytics Server run using SAS In-Database techniques. Therefore, for these processes, there also is no data movement between SAS and Hadoop (Hive). However, score code application executes in the SAS server. Therefore, data is extracted from Teradata. After the analytical processing is complete, data is loaded back into Teradata.

Non-Distributed Processing with SAS as the Data Store

For both design and batch mode runs, SAS Customer Link Analytics executes all processes in SAS and in SMP mode.

6

Configuring the Application

Log File Locations

Installation Log File

When you complete the installation, the sia_apdm_config_wrapper.log file is created in the following location: `<SAS configuration directory>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/logs`. This file contains the logs of configuration for the application data tables. As a verification task, you can read this log file and make sure that it does not contain any errors.

Log Files for Projects and Scenarios

Log File Details for Administrative Tasks

The log files that are created for the administrative tasks are stored in the following location: **<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/admin/logs**.

Table 6.1 *Log Files for Tasks in Administration Workspace*

Task or Workflow Step	Log File Created in Design Mode	Log File Created in Batch Mode
Refresh statistics	sia_refresh_stat_<*table_pk*>.log	-
Extract scenario ID	sia_stp_get_project_scenario_dtl.log	-

Log File Details for Tasks in Projects Workspace

The log files that are created for the tasks that you perform in the Projects workspace are stored in the following location: **<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/projects/logs**.

Table 6.2 *Log Files for Tasks in Projects Workspace*

Task or Workflow Step	Log File Created in Design Mode	Log File Created in Batch Mode
Create project	sia_stp_project_creation_<*project_pk*>.log	-
Data extraction	sia_stp_exec_src_data_extr_<*project_pk*>.log	sia_stp_exec_src_data_extr_<*project_pk*>.log
Link and node filtering	sia_stp_exec_lnk_nde_process_<*project_pk*>.log	sia_stp_exec_lnk_nde_process_<*project_pk*>.log
	sia_stp_lnk_wt_exp_validate_<*project_pk*>.log	-
Community building	sia_stp_exec_run_comm_<*project_pk*>.log	sia_stp_exec_run_comm_<*project_pk*>.log
	sia_stp_exec_bld_comm_<*project_pk*>.log	-
Centrality measures computation	sia_centrality_msr_comp_wrapper_<*project_pk*>.log	sia_centrality_msr_comp_wrapper_*project_pk*>.log

Task or Workflow Step	Log File Created in Design Mode	Log File Created in Batch Mode
Role assignment	sia_stp_exec_role_assignmnt_<*project_pk*>.log	sia_stp_exec_role_assignmnt_<*project_pk*>.log
	sia_stp_role_exp_validate_<*project_pk*>.log	-
Push project to batch mode	sia_batch_push_<*project_pk*>.log	-
Pull project to design mode	sia_batch_to_design_<*project_pk*>.log	-
Execute project in batch mode	-	sia_batch_<*project_pk*>.log
Reset a workflow step	sia_stp_exec_reset_<*project_pk*>.log	-
Enrich project's output data **Note:** In addition to this log file, certain other log files are created depending on the enrichment category that you choose and the mode of execution of the data enrichment process that you have set up. For more information, see "Log Files for Data Enrichment Categories and Mode of Execution" on page 49.	sia_stp_prepare_data_<*project_pk*>.log	sia_stp_prepare_data_<*project_pk*>.log
Copy data to SAS Customer Link Analytics LASR Analytic Server	sia_stp_copy_data_to_lasr_<*project_pk*>.log	sia_stp_copy_data_to_lasr_<*project_pk*>.log

Log File Details for Scenario-Related Tasks

The Log files that are created for scenario-related tasks are stored in the following location: `<SAS configuration directory>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/scenario/logs`

Table 6.3 *Log Files for Scenario-Related Tasks*

Task	Log File Created in Design Mode	Log File Created in Batch Mode
Create a scenario	sia_stp_scenario_creation_<*scenario_nm*>.log	-
Update scenario parameters	sia_update_scenario_param_<*scenario_pk*>.log	-

Task	Log File Created in Design Mode	Log File Created in Batch Mode
Build ABT	sia_stp_build_abt_<*scenario_pk*>.log	-
	sia_abt_loopjob_<*scenario_pk*>.log	-
	sia_scen_bld_abt_roles_commns_<*scenario_pk*>.log	-
	sia_scen_bld_abt_role_lvl_var_<*scenario_pk*>.log	-
	sia_scen_bld_abt_node_attribs_<*scenario_pk*>.log	-
	sia_scen_bld_abt_event_measures_<*scenario_pk*>.log	-
	sia_scen_bld_abt_event_indicator_<*scenario_pk*>.log	-
	ia_scen_bld_abt_event_count_<*scenario_pk*>.log	-
	sia_scen_bld_abt_comm_lvl_stat_<*scenario_pk*>.log	-
	sia_scen_bld_abt_aggr_trans_<*scenario_pk*>.log	-
Build viral model	sia_stp_exec_build_viral_model_<*scenario_pk*>.log	-
Register model	sia_register_viral_model_<*scenario_pk*>.log	-
Capture model information	sia_capture_model_info_<*scenario_pk*>.log	-
Publish scenario for scoring	sia_stp_pblsh_scen_for_scoring_<*scenario_pk*>.log	-

Task	Log File Created in Design Mode	Log File Created in Batch Mode
Execute a scenario in batch mode	-	sia_abt_loopjob_<*scenario_pk*>.log
	-	sia_stp_build_abt_<*scenario_pk*>.log
	-	sia_scen_bld_abt_roles_commns_<*scenario_pk*>.log
	-	sia_scen_bld_abt_role_lvl_var_<*scenario_pk*>.log
	-	sia_scen_bld_abt_node_attribs_<*scenario_pk*>.log
	-	sia_scen_bld_abt_event_measures_<*scenario_pk*>.log
	-	sia_scen_bld_abt_event_indicator_<*scenario_pk*>.log
	-	sia_scen_bld_abt_event_count_<*scenario_pk*>.log
	-	sia_scen_bld_abt_comm_lvl_stat<*scenario_pk*>.log
	-	sia_scen_bld_abt_aggr_trans_<*scenario_pk*>.log
	-	sia_scenario_scoring_job_<*scenario_pk*>.log
	-	sia_scenario_scoring_wrapper_<*scenario_pk*>.log
Pull scenario to design mode	sia_stp_push_scen_to_design_<*scenario_pk*>.log	-

The sia_appl_debug_flg parameter determines the level of details that are logged in the log files. This parameter is defined in the PARAM_VALUE application data table. If the value of this parameter is set to Y, then a detailed log is generated. However, if it is set to N, then minimal information is logged. In addition, the type of details that are logged in the file are determined by the sia_appl_debug_options parameter. Also, if you set the value of the sia_sql_ip_trace_flg parameter to Y, then the log file contains additional information about the SQL trace messages. .For more information, see on page 79

Log Files for Data Enrichment Categories and Mode of Execution

Log files are created for the data enrichment process depending on the enrichment categories that you choose at node or link level. These files are

stored in the following location: **<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/projects/logs**.

Table 6.4 *Log Files for Data Enrichment Categories*

Category Level	Enrichment Category Name	Log File Created in Design and Batch Mode
Node	Roles and communities	sia_prep_data_deflt_out_<*project_pk*>.log
	Roles and communities over time	sia_prep_data_deflt_node_prev_<*project_pk*>.log
	Aggregated transactional data	sia_prep_data_nodelvl_aggrns_<*project_pk*>.log
	Node attributes	sia_prep_data_node_attribs_<*project_pk*>.log
	Churn and acquisition indicators	sia_prep_data_drvd_ind_<*project_pk*>.log
	Community-level statistics	sia_prep_data_commlvl_aggrns_<*project_pk*>.log
	Associations with neighboring roles	sia_prep_data_role_lvl_aggr_node_<*project_pk*>.log
	Relation with churned and acquired nodes	sia_prep_data_churn_acq_vars_<*project_pk*>.log
Link	Roles and communities	sia_prep_data_deflt_link_data_<*project_pk*>.log
	Node attributes	sia_prep_data_node_attrib_link_<*project_pk*>.log
	Churn and acquisition indicators	sia_prep_data_drvd_ind_link_<*project_pk*>.log

The following additional log files are created depending on whether the mode of execution of the data enrichment process is parallel or sequential. For more information, see the sia_de_job_exec_mode parameter in "Project-Specific Parameters" on page 79. For parallel execution, the sialoopjob_<project_pk>.log file is created. However, for sequential execution, the sia_call_seq_de_jobs__<*project_pk*>.log is created.

These files are stored in the following location: **<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/projects/logs**.

Middle-Tier Log File

The logs of the middle-tier component are maintained in the SASCustLinkAnlytics6.5.log file. This file is available in the following default

location: **<SAS configuration directory>/Lev1/Web/Logs/<SASServer11_1>**. For example, on a Windows machine, by default, the log file is available in the following location: **<SAS configuration directory>/Lev1/Web/Logs/SASServer11_1**. However, if you perform a custom installation, the folder name that indicates the SAS server might change depending on the SAS server that you configure.

Working with Software Component Properties

Overview

SAS Customer Link Analytics has software component properties that are defined for the following components:

- SAS Customer Link Analytics Server
- SAS Customer Link Analytics Middle Tier

It is recommended that you do not modify the default values of these component properties. However, if you modify the values, you must re-deploy SAS Customer Link Analytics on your web application server in order to reflect the changes that you made.

View or Modify Software Component Properties

To view or modify software component properties, complete the following steps:

1. Open SAS Management Console, and connect to the appropriate metadata server.
2. On the **Plug-ins** tab, select **Application Management** ▸ **Configuration Manager**.
   - To access the SAS Customer Link Analytics Server component properties:
     1. Right-click **Cust Link Analytics Svr Cfg 6.5** and select **Properties**. The Cust Link Analytics Svr Cfg 6.5 Properties window appears.
     2. Select the **Advanced** tab and view the properties. For details, see Table 6.5 on page 52.
   - To access the SAS Customer Link Analytics Middle-Tier component properties:
     1. Expand **SAS Application Infrastructure**.
     2. Right-click **Cust Link Analytics 6.5** and select **Properties**. The Cust Link Analytics 6.5 Properties window appears.
     3. Select the **Advanced** tab and view the properties. For details, see Table 6.6 on page 58.
3. (Optional) Change the default value of the properties, if required, and save the changes.

4 Close SAS Management Console.

SAS Customer Link Analytics Server Component Properties

The value of some of the software component properties is populated depending on the value that you specify for the corresponding SAS Deployment Wizard prompt.

Table 6.5 Server Properties

Property Name	Sample Value	Description
clasvrc.abt.loop.job.location	`C:\SAS\Config\Lev1\AppData\SASCustomerLinkAnalytics\6.5\scenario\batchcode`	The physical location in which the siascenloopjob.sas job is stored. **Note:** Make sure that the value of this property and the value that you specify for **Deployment Directory** when you deploy the loop job is the same. For more information, see "Deploy the Loop Job for Viral Effect Analysis" on page 26.
clasvrc.abt.loop.job.name	siascenloopjob.sas	Stores the name of the loop job that controls the execution of the process that generates the ABT variables for viral effect analysis. **Note:** Make sure that the value of this property and the value that you specify for **Deployed Job Name** when you deploy the loop job is the same. For more information, see "Deploy the Loop Job for Viral Effect Analysis" on page 26.
clasvrc.analytics.inter.libref	sia_ani or sia_hive	Displays the library reference that you specify for the **Analytics Data Intermediate Schema Name** prompt when you install SAS Customer Link Analytics. SAS Customer Link Analytics uses this reference to access the analytics intermediate library. For SAS and Teradata, the default value is sia_ani. However, for Hadoop (Hive) the default value is sia_hive. If you change this value, make sure that you specify the reference of a pre-assigned library. Also, the reference that you specify must be the same as it is declared in the metadata.

Property Name	Sample Value	Description
clasvrc.analytics.output.libref	sia_anop or sia_hive	Displays the library reference that you specify for the **Analytics Data Output Schema Name** prompt when you install SAS Customer Link Analytics. SAS Customer Link Analytics uses this reference to access the analytics output library. For SAS and Teradata, the default value is sia_anop. However, for Hadoop (Hive), the default value is sia_hive. If you change this value, make sure that you specify the reference of a pre-assigned library. Also, the reference that you specify must be the same as it is declared in the metadata.
clasvrc.apdm.libref	sia_apdm	Displays the library reference that you specify for the application data when you install SAS Customer Link Analytics. SAS Customer Link Analytics uses this reference to access the library that stores the application data tables. If you change this value, you must specify the reference of a pre-assigned PostgreSQL library. Also, the reference that you specify must be the same as it is declared in the metadata.
clasvrc.appdata.location	<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5	Indicates the parent location in which folders that store application logs, SAS data, batch code, and score code are stored.
clasvrc.applications.location	<*SAS configuration directory*>/Lev1/Applications/SASCustomerLinkAnalytics6.5	The physical folder path for SAS Customer Link Analytics in which stored processes and user-created SAS macros required for viral effect analysis are stored.

Property Name	Sample Value	Description
clasvrc.bdm.intermediate.libref	sia_bdim or sia_hive	Displays the library reference that you specify for the **Business Data Intermediate Schema Name** prompt when you install SAS Customer Link Analytics. SAS Customer Link Analytics uses this reference to access the intermediate library that stores the business data tables. For SAS and Teradata, the default value is sia_bdim. However, for Hadoop (Hive), the default value is sia_hive. If you change this value, make sure that you specify the reference of a pre-assigned library. Also, the reference that you specify must be the same as it is declared in the metadata.
clasvrc.bdm.output.libref	sia_bdop or sia_hive	Displays the library reference that you specify for the **Business Data Output Schema Name** prompt when you install SAS Customer Link Analytics. SAS Customer Link Analytics uses this reference to access the output library that stores the business data tables. For SAS and Teradata, the default value is sia_bdop. However, for Hadoop (Hive), the default value is sia_hive. If you change this value, make sure that you specify the reference of a pre-assigned library. Also, the reference that you specify must be the same as it is declared in the metadata.
clasvrc.bdm.smp.output.libref	sia_sasd	Displays the library reference for the SAS library that stores the intermediate SAS data. This data is created when a project is run.
clasvrc.cla.bdm.dbms.type	SAS, Hadoop, or Teradata	Indicates whether SAS Customer Link Analytics uses a SAS, Hadoop (Hive), or Teradata database to store business data. By default, the value that you specify during installation is displayed. If you change the value of this property, make sure that you maintain consistency between the relevant library references in the software component properties and the actual configuration.

Property Name	Sample Value	Description
clasvrc.cla.execution.mode	NONDISTRIBUTED or DISTRIBUTED	Displays the mode of execution that you select when you install SAS Customer Link Analytics. If you configure SAS Customer Link Analytics for Teradata, then you can select the execution mode as distributed or non-distributed. However, for the Hadoop (Hive) database, the execution mode can be distributed and for the SAS database, the execution mode can be non-distributed.
clasvrc.cla.grid.installloc	`/opt/v940m2/INSTALL/TKGRID_REP`	Applicable for distributed mode. The value of this property indicates the location in which the grid is installed and configured. By default, the value that you specify for the **Installation Location** prompt for the High-Performance Analytics Grid Server when you install SAS Customer Link Analytics is displayed. SAS Customer Link Analytics uses this location to set up certain execution parameters. SAS Deployment Wizard deploys the relevant code in this location.
clasvrc.cla.grid.server	Teradbm	Applicable for distributed mode. The host name of the grid server that you specify for the **Host Name** prompt for the High-Performance Analytics Grid Server when you install SAS Customer Link Analytics is displayed as the value of this property. SAS Customer Link Analytics uses this value to set up certain parameters that are required to execute processes in distributed mode.
clasvrc.cla.lasr.grid.port	10071	Displays the port of the SAS Customer Link Analytics LASR Analytic Server. By default, the value that you specify for the **Port for SAS Customer Link Analytics LASR Analytic Server** prompt when you install SAS Customer Link Analytics is displayed.

Property Name	Sample Value	Description
clasvrc.cla.lasr.installloc	/opt/v940m2/INSTALL/TKGrid	Indicates the location in which the grid that is used for the SAS Customer Link Analytics LASR Analytic Server is installed and configured. By default, the value that you specify for the **Root location on the SAS Customer Link Analytics LASR Environment to be used for signature files** prompt when you install SAS Customer Link Analytics is displayed.
clasvrc.cla.lasr.server	BIGMATH	Indicates the machine name of the SAS Customer Link Analytics LASR Analytic Server. By default, the value that you specify when you install SAS Customer Link Analytics is displayed.
clasvrc.cla.lasr.type	DISTRIBUTED or NONDISTRIBUTED	Indicates whether the type of the SAS Customer Link Analytics LASR Analytic Server is **Distributed** or **Non-Distributed**.
clasvrc.cla.servercontext	SASApp	Displays the server context.
clasvrc.config.dir	`C:\SAS\Config\Lev1`	Displays the physical path of the SAS configuration directory. SAS Customer Link Analytics uses this path as a relative path to define other paths such as log locations.
clasvrc.hadoop.auth.type	User and Password, User, and Kerberos	Indicates the authentication type that is applicable for Hadoop (Hive).
clasvrc.hpa.grid.mode	sym or asym	Applicable for distributed mode. The value of this property is sym (Symmetric) or asym (Asymmetric) depending on the value that you specify for the type of grid server configuration.
clasvrc.lasr.libref	sia_lasr	Displays the library reference that is assigned by default for the SAS Customer Link Analytics LASR Analytic Server when you install SAS Customer Link Analytics.

Property Name	Sample Value	Description
clasvrc.loop.job.location	<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/projects/batchcode	Displays the location in which the batch code of the loop job that is deployed for the data enrichment process is stored. **Note:** Make sure that the value of this property and the value that you specify for **Deployment Directory** when you deploy the loop job is the same. For more information, see "Deploy the Loop Job for Data Enrichment" on page 26.
clasvrc.loop.job.name	sialoopjob.sas	Displays the name of the loop job that is deployed for controlling the execution of the data enrichment process. **Note:** Make sure that the value of this property and the value that you specify for **Deployed Job Name** when you deploy the loop job is the same. For more information, see "Deploy the Loop Job for Data Enrichment" on page 26.
clasvrc.metadata.server.name	<*Metadata server name*>.com	Displays the name of the metadata server.
clasvrc.metadata.server.port	8561	Displays the port of the metadata server.
clasvrc.projectfolder.location	`/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Projects`	Indicates the metadata location in which the project data is stored. This location is called the *project path*.
clasvrc.repository.name	Foundation	Displays the repository name of the metadata server.
clasvrc.scenariofldr.location	`/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Viral Effect Analysis`	Indicates the metadata location in which the scenario data is stored.
clasvrc.version	6.5	Displays the version of SAS Customer Link Analytics.

SAS Customer Link Analytics Middle Tier Component Properties

Table 6.6 *Middle-Tier Properties*

Property Name	Sample Value	Description
clamid.EXECUTION_MODE	NONDISTRIBUTED or DISTRIBUTED	Displays the mode of execution that you select when you install SAS Customer Link Analytics.
clamid.STP_FOLDER_PATH	`/System/Applications/SAS Customer Link Analytics/ Cust Link Analytics 6.5/ Application Stored Process/`	Displays the location in which the stored processes are stored.
clamid.samplereports.location	/Products/SAS Customer Link Analytics/Cust Link Analytics 6.5/ Sample Reports	Displays the metadata location in which the default community reports are stored.

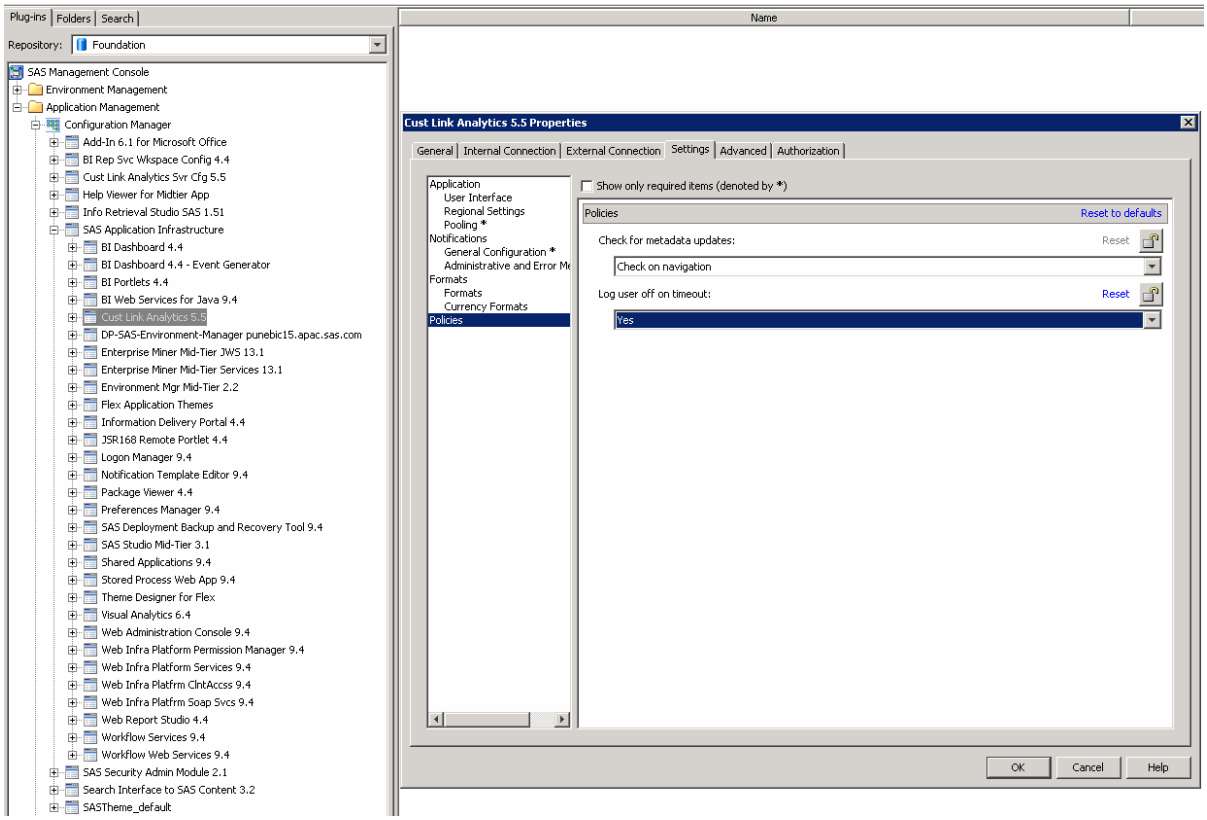
Change the Policy Settings for Session Timeout

You might have logged on to SAS Customer Link Analytics. However, if your session is idle due to inactivity, then the session timeout page appears. On this page, you can either select **LogOff** or **Return to Application**. By default, when you click **Return to Application**, your session is reloaded and the main page appears. In this case, you are not prompted for your password. However, if you want to require users to enter their password before returning to the application, you can change the appropriate policy setting in SAS Management Console.

To change the policy setting:

1. Open SAS Management Console and connect to an appropriate profile.
2. On the **Plug-ins** tab, select **Application Management** ▸ **Configuration Manager** ▸ **SAS Application Infrastructure**.
3. Right-click **Cust Link Analytics 6.5** and select **Properties**. The Cust Link Analytics 6.5 Properties window appears.
4. On the **Settings** tab, select the **Policies** page.
5. From the **Log user off on timeout** list, select **Yes**.

Figure 6.1 Policy Setting



6 Click **OK**.

7 Close SAS Management Console.

Using the Lockdown Path List

Overview of the LOCKDOWN Statement

The LOCKDOWN statement secures a SAS Foundation server by restricting access from within a server process to the host operating environment. It enables you to limit access to the back-end file system and to specific SAS features for a SAS session executing in a server or batch processing mode. This restriction prevents back-end SAS servers such as the SAS Workspace Server or the SAS Stored Process Server from accessing file system paths that are not defined in the lockdown path list (also called a *whitelist*). The lockdown path list specifies the files and directories that a SAS session can access when the SAS servers are locked down. All the subdirectories of the directory that is specified in the path list can be accessed. However, attempts made to access paths outside of the path list are denied.

A SAS server that is locked down is constrained as mentioned here:

- The server can access only the host directories and files that are specified in the lockdown path list.
- The server cannot run the DATA step javaobj methods.

- The server cannot run the GROOVY procedure.
- The server cannot run the JAVAINFO procedure.
- The server cannot invoke the following functions: MODULE, ADDR, ADDRLONG, PEEK, PEEKLONG, PEEKC, PEEKCLONG, POKE, and POKELONG.

Adding the Lockdown Path

To ensure that the lockdown feature functions appropriately for SAS Customer Link Analytics, you have to update the lockdown path list in certain files.

To update the lockdown path in the appropriate files:

1. Search for the autoexec_usermods.sas file. Make sure that the search retrieves the files in the following folders:
   - **<*SAS configuration directory*>/Lev1/SASApp/ConnectServer**
   - **<*SAS configuration directory*>/Lev1/SASApp/WorkspaceServer**
   - **<*SAS configuration directory*>/Lev1/SASApp/StoredProcessServer**
   - **<*SAS configuration directory*>/Lev1/SASApp/PooledWorkspaceServer**
   - **<*SAS configuration directory*>/Lev1/SASApp/BatchServer**
   - **<*SAS configuration directory*>/Lev1/SASApp**
   - **<*SAS configuration directory*>/Lev1/SASMeta**
2. In each of the file that is retrieved in the locations listed here, add the following lines:

```
LOCKDOWN PATH ='<SAS configuration directory>/Lev1/AppData/SASCustomerLinkAnalytics';

LOCKDOWN PATH ='<SAS configuration directory>/Lev1/Applications

/SASCustomerLinkAnalytics6.5';

LOCKDOWN PATH ='<SAS configuration directory>/Lev1/AppData

/SASCustLinkAnlytcsLASRCfg/sigfiles';

LOCKDOWN PATH ='<SAS Home>/ReportFontsforClients/9.4';
```

   Note:
   - In the lockdown path statement, replace **<*SAS configuration directory*>** with the actual path, such as **C:/SAS/Config**.
   - If the SAS Enterprise Miner project location that you specify while executing a scenario is different from the paths that are listed here, then you have to also add this path in each of the files. For example, assume that the project location on a Windows machine is **C:/modeling/em_projects**. In this case, add the following lockdown path: `LOCKDOWN PATH ='C:/modeling/em_projects';`.
3. Save the file.
4. In SAS Data Integration Studio, run the following command: `%siainit;`. Make sure that it does not result into any errors.

5 Log on to SAS Customer Link Analytics. Make sure that you can run a project in batch and design mode without any errors. Also, make sure that you can successfully perform scenario analysis.

Confirming the Structure of a Table

Importing Tables in SAS Customer Link Analytics

SAS Customer Link Analytics enables you to import tables from which you extract source data that is required for configuring and running the workflow steps of a project. For more information about how to import a table, see *SAS Customer Link Analytics: User's Guide*.

To enable users to import tables, you must register them in the metadata. Before you register a table, you must confirm the structure of the table. The structure of a table differs depending on the type of table that you are registering. SAS Customer Link Analytics enables you to register the following types of tables:

- Transactional tables
- Attribute tables
- Inclusion lists

For more information about these tables, see *SAS Customer Link Analytics: User's Guide*.

Structure of a Transactional Table

Make sure that the transactional table contains the following columns:

- From node ID
- To node ID
- Transactional date
- Transactional measure

A transactional table can have more than one transactional measure. In addition, it can contain dimension or date type columns.

Note: Make sure that a timestamp is attached to the transactional date column. It cannot be only a date type column.

Structure of an Attribute Table

An attribute table is further classified as a node attribute table or a link attribute table.

Node attribute table
: Make sure that the node attribute table contains the from node ID column. The data type and length of this column must be the same as the from node ID of the transactional table. In addition, a node attribute can contain one or more dimension and date type columns.

Link attribute table
: Make sure that the link attribute table contains the from node ID column and the to node ID column. The data type and length of these columns must be the same as the from node ID and the to node ID of the transactional table. In addition, a link attribute can contain one or more dimension and date type columns.

Structure of an Inclusion List

An inclusion list is further classified as a node inclusion list or a link inclusion list.

Node inclusion list
: Make sure that the node inclusion list contains only the from node ID column. The data type and length of this column must be the same as the from node ID of the transactional table.

Link inclusion list
: Make sure that the link inclusion list contains only the from node ID column and the to node ID column. The data type and length of these columns must be the same as the from node ID and the to node ID of the transactional table.

Verifying the Mandatory Columns in the Event Details Table

Overview of Event Details Table

You have to create an event details table for viral effect analysis and register it in SAS metadata. Depending on the scenario type for which you are defining the event details table, you can define one or more event details tables according to your requirements. For example, you can define an event details table for the **Churn** scenario type and another one for the **Product adoption** scenario type. However, you must ensure that each table contains certain mandatory columns.

Mandatory Column Types for Viral Churn

The event details table that you create for viral churn analysis must contain two mandatory columns: the ID that uniquely identifies a node and the date on which a node churns out of the network.

This table can contain information only about nodes that have churned from the network. In addition, it can contain information about nodes that have not churned from the network.

Note: If the table contains information only about nodes that have not churned from the network, the processing of viral effect analysis fails while creating a target variable. For more information about target variables, see *SAS Customer Link Analytics: User's Guide*.

Table 6.7 *Viral Churn: Sample Structure of Event Details Table in SAS*

Column Name	Data Type	Length	Description
node_id	Number	8	Indicates the column that uniquely identifies a node.
churn_dt	Number	8	Indicates the date on which a node has churned out of the network.

Note: The churn date column must have a date in the datetime format.

Mandatory Column Types for Viral Product Adoption

The event details table that you create for viral product adoption must contain three mandatory columns: the ID that uniquely identifies a node, the ID that uniquely identifies a product that a node owns, and the date on which a node starts owning a product.

This table can contain information only about nodes that have adopted a product. In addition, it can contain information about nodes that have not adopted a product.

Note: If the table contains information only about nodes that have not adopted a project, the processing of viral effect analysis fails while creating a target variable. For more information about target variables, see *SAS Customer Link Analytics: User's Guide*.

Table 6.8 *Viral Product Adoption: Sample Structure of Event Details Table in SAS*

Column Name	Data Type	Length	Description
node_id	Number	8	Indicates the column that uniquely identifies a node.
product_adoption_dt	Number	8	Indicates the date on which a node starts owning a product.
product_id	Text	9	Indicates the column that uniquely identifies a product that a node owns.

Note: The product adoption date column must have a date in the datetime format.

Run ABT-Building Processes Sequentially

When you deploy the siascenloopjob job, the ABT-building processes of viral effect analysis are run concurrently. However, if you want to run them sequentially, you have to modify its properties.

To change the properties for sequential processing:

1. Connect to SAS Data Integration Studio with administrative privileges.
2. On the **Folders** tab, expand **Products** ▸ **SAS Customer Link Analytics** ▸ **Cust Link Analytics 6.5** ▸ **Jobs**.
3. Double-click **siascenloopjob**.
4. In the job diagram, right-click **loop_parallel_cla_scenario** and select **Properties**.
5. On the **Loop Options** tab, select the **Use this number** option for **Maximum number of concurrent processes**. Make sure that the value for this option is set to 1.
6. Click **OK**.
7. Redeploy the siascenloopjob job. For more information, see “Deploy the Loop Job for Viral Effect Analysis” on page 26.

Update the SAS Enterprise Miner Project Workspace Path

You might create various SAS Rapid Predictive Modeler models for viral effect analysis. These models are created as SAS Enterprise Miner project workspaces. If you move the SAS Enterprise Miner project workspace to another location, then make sure that theSAS Customer Link Analytics execution user has Read and Write permission for the new folder.

If you change the location of the project workspace, perform the following steps:

1. Connect to the PostgreSQL client.
2. Connect to the PostgreSQL database that is created for SAS Customer Link Analytics.
3. Open the public schema.
4. Open the scenario_param table.
5. Filter the data based on the following conditions:

   Column = *scenario_param_id* and **Value** = *sia_project_location*. This filter condition displays a list of all the SAS Enterprise Miner project workspaces that are created in the source environment.

6 For each value of sia_project_location, edit the value of the scenario_param_userdefined_value column. Specify the new location in which the corresponding SAS Enterprise Miner project workspace is saved.

7 Save the changes.

7

Batch Processing

Overview of Batch Processing

When you have successfully run all the workflow steps of a project that you created in the Projects workspace, the project completes one run in design mode. You can then push this project to batch mode. Similarly, when you publish a scenario that you have defined for a project that is in batch or design mode, you can run that scenario in batch mode. All these tasks involve batch processing. For details about pushing a project to batch mode and publishing a scenario, see *SAS Customer Link Analytics: User's Guide*.

For batch mode, a SAS code file is created for each project or scenario in a predefined location. You have to manually execute this code or schedule it to run at a predefined frequency. If you schedule the batch code, the batch process runs seamless without any manual intervention.

Running a Project in Batch Mode

Overview

When you push a project to batch mode, a batch code file (sia_batch_exe_<*Project name*>_<*Project ID*>.sas) is created for this project in the following folder: `<SAS configuration directory>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/projects/batchcode`. For more information, see *SAS Customer Link Analytics: User's Guide*. You can run this

code in the SAS environment or schedule it to run at regular intervals. To run the batch code, you must have access to the application data tables and business data tables. Also, you can define the frequency at which you want to schedule the batch run according to your business needs.

When you run the batch code, the following tasks are run in this sequence:

1. All the workflow steps of the project are run.
2. The data enrichment process is run.
3. (Optional) The enriched data is loaded into the SAS Customer Link Analytics LASR Analytic Server.

When these steps run successfully, the project completes one run in batch mode. The status of the batch run is displayed in the SAS Customer Link Analytics interface.

Running the Workflow Steps in Batch Mode

When you run the batch code, the workflow steps run according to how you have configured them in design mode. If you select the **Create community segments** check box in the Centrality Measure Computation workflow step, additional steps are run. For more information, see “Batch Processing for Community Segmentation” on page 69.

If an error occurs when the workflow steps run in batch mode, you can control the execution in the next batch run. To do so, you can set the value of the Rerunvl parameter that is defined in the batch code. The default value of this parameter is N. This value indicates that if a workflow step fails to execute in a batch run, then the next batch run starts from the workflow step that failed to execute. However, if you want to run all the workflow steps again, regardless of which workflow step failed to run, set the value of this parameter to Y.

Running the Data Enrichment Process in Batch Mode

The batch run of the data enrichment process begins after the batch run of all the workflow steps has successfully completed. The batch run of data enrichment is processed regardless of whether its design run is complete. The data is enriched depending on the enrichment categories that you have selected for the project.

If an error occurs while the data enrichment process is running in batch mode, then the Rerunv1 parameter determines its execution in the next run. If the value of the parameter is set to N, then the batch run begins from the data enrichment process. However, if the parameter is set to Y, all the workflow steps are rerun first, followed by the data enrichment process.

Running the Data Loading Process in Batch Mode

For each project, you can decide whether you want to load the enriched data in a batch run. To do so, set the sia_load_lasr_ind parameter that is stored in the Project_process_param table. The default value of this parameter is Y. It indicates that data will be loaded into the SAS Customer Link Analytics LASR Analytic Server after the data enrichment process has run successfully in batch mode.

If an error occurs while the data loading process is running in batch mode, then the Rerunv1 parameter determines its execution in the next run. If the value of the parameter is set to N, then only the loading process is rerun in the next batch run. However, if the parameter is set to Y, all the workflow steps are rerun first, followed by the data enrichment process, and then the data loading process.

Batch Processing for Community Segmentation

The batch code processes the community segmentation batch flow if you have chosen to create segments for the communities in the design mode of the Centrality Measures Computation workflow step.

The following steps are processed in the batch flow of community segmentation:

1 Building scoring ABT.

A scoring ABT is built for the project. The scoring ABT contains variables that are the same as the variables of the modeling ABT. These variables provide community-level statistics and are computed based on centrality measures, transactional measures, link weights, and the diameter and density of the communities. For more information about the modeling ABT variables, see *SAS Customer Link Analytics: User's Guide*.

2 Building reduced scoring ABT.

A reduced scoring ABT is built by extracting information only about the reduced variables from the scoring ABT that is built in step 1. The reduced scoring ABT is stored as a SAS table. For more information about the reduced variables, see *SAS Customer Link Analytics: User's Guide*.

3 Applying score code.

The score code that is produced during the design run is applied to the reduced scoring ABT and a scored ABT is built. The scored ABT contains segment information and indicates the distribution of communities across segments. One or more communities can be grouped into a segment. As a result, a segment ID is assigned to each community. For more information about the score code, see *SAS Customer Link Analytics: User's Guide*.

4 Score writeback.

The segment ID that is assigned to a community is also assigned to each node of the community. This information is written back to the business data.

5 Publishing results.

The segment-level statistics are produced and published on the **Community Segmentation Results** tab. For more information, see *SAS Customer Link Analytics: User's Guide*.

Running a Scenario in Batch Mode

Overview

Batch execution is the process of applying an analytical model to new data in order to compute outputs. When you publish a scenario to batch mode, a batch code file (sia_scenario_scoring_job_<*Scenario name*>_<*Scenario ID*>) is generated for this scenario in the following folder: **<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/scenario/batchcode**. For details, see *SAS Customer Link Analytics: User's Guide*. You can run this code in the SAS environment or schedule it to run at regular intervals. You can define the frequency at which you want to schedule the batch run according to your business requirements. However, it is recommended that you schedule the batch run of a scenario with the batch run of the project that is associated with the scenario. This ensures that the scores of the latest network are produced. Also, a batch run of a scenario can be scheduled more frequently compared to the batch run of the project. The frequency with which you want to schedule the batch run of the scenario depends on the frequency with which the data in the event details table is updated. For more information, see "Verifying the Mandatory Columns in the Event Details Table" on page 62.

In the batch run, if the project associated with the scenario has any configuration changes that impact the significant variables, the scenario needs to be pushed to design mode. For more information, see "Verifying the Configuration Changes for a Project" on page 72. For information about pulling a project back into design mode, see *SAS Customer Link Analtyics: User's Guide*.

In other cases, the batch run executes the scoring process only. The scoring process includes scoring ABT creation, score code application, score writeback, and report data creation.

The Scoring Job

When you publish a scenario, a batch code file (also called, a *scoring job*) is created. SAS Customer Link Analytics does not provide the framework to run or schedule this job. As an administrator, you must manually schedule this job (through an external scheduler). Typically, the scoring job is scheduled to run more frequently than the batch execution of a project.

Note: This job uses certain pre-assigned libraries such as the sia_apdm library that are defined in the SAS Metadata Server. Therefore, make sure that these pre-assigned libraries are available to the session in which this job is run.

The scoring job, scoring_run_job_exec_scoring_job_<*scenario_name*>_<*scenario_pk*>.sas, contains code that is similar to the code included here:

```
%macro sia_scenario_scoring;

%siainit;

%let sia_smd_err_key = E_00000;

%if &sia_rc.= 0 %then %do;

%sia_scenario_scoring_job(sia_scenario_pk=);
```

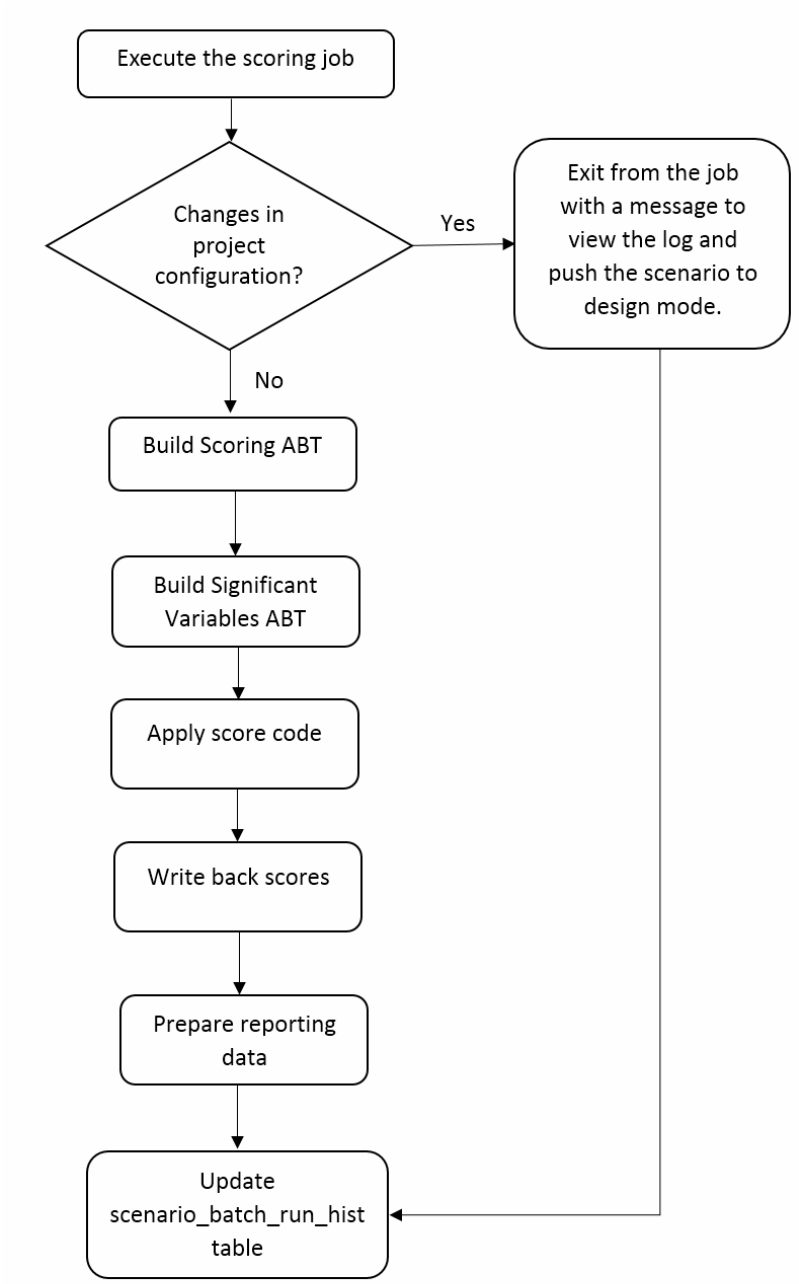
```
%end;

%mend sia_scenario_scoring;

%sia_scenario_scoring;
```

In this code, sia_scenario_pk is the unique sequence number that is assigned to the scenario when it is created.

High-Level Flow of Scenario Batch Execution

Flow Diagram

Figure 7.1 *Flow Diagram for Scenario Batch Execution*



Verifying the Configuration Changes for a Project

The batch run confirms whether there are any configuration changes at the project level that impact the significant variables. These configuration changes could be deletion of a role or deletion of a centrality measure that was

considered in the design run of the scenario. If these changes impact the significant variables, then the batch run stops its execution with a message. This message indicates that you have to push the scenario to design mode and run its analytical flow again. For more information, see *SAS Customer Link Analytics: User's Guide*. However, if these changes do not impact the significant variables, the batch run continues the scoring processes.

For example, assume that in design mode, the analytical model identifies the variable that measures the hub centrality (hub_val) as a significant variable. The batch run verifies that the project configuration is subsequently changed. This change indicates that in the subsequent run of that project, the hub centrality will be deselected in the Centrality Measure Computation workflow step. As a result, the hub centrality is not computed for the project. Because this change impacts the significant variable, the batch run stops its execution and indicates that you have to build the modeling ABT again. Therefore, you have to push the scenario to design mode.

Similarly, assume that in design mode, a role-specific variable is identified as a significant variable. The batch run verifies that a role has been deleted. Because this configuration change impacts the significant variables, the batch run stops its execution, and you have to pull the scenario back into design mode. For more information about pulling a scenario into design mode, see *SAS Customer Link Analytics: User's Guide*.

Scoring Run

In the scoring run, the following steps occur:

1 Create scoring ABT.

  The scoring ABT contains the variables that belong to the significant variables categories.

2 Create significant variables ABT

  From the scoring ABT, the significant variables are extracted to create the significant variables ABT. This ABT is registered in the following metadata location: **/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Viral Effect Analysis/<*Scenario ID*>/Data Sources/Score Data**.

3 Apply score code.

  After the significant variables ABT is created, the next step is applying the score code that is generated as a part of the modeling run of the scenario. The score code of the successfully built model for the given scenario is available in the following location: **<*SAS configuration directory*>/Lev1/AppData/SASCustomerLinkAnalytics/6.5/scenario/modelscorecode.**

  The process reads the score code from this location and applies it on the scoring ABT. As a result, the scores are generated. The scored ABT is stored under the sia_scsd library.

4 Write back scores.

  After the scored ABT is generated, the newly generated scores are written back to the designated area. The scenario_score_writeback table stores the scores for all the runs of the scenario. It also stores information such as the

model for which scoring is performed and the date on which scoring is performed.

Note: For a scenario, the scoring process is run only once on a particular date.

5 Prepare reporting data

In the first scoring run, the reporting data is populated in the scen_rptdata_<*Scenario ID*> table. In the next run, the variables that are populated for the previous run are renamed with the prefix PREV and the same set of variables is added as new columns.

Note: Data of two successive scoring runs is available in the reporting table only if the analytical model of the scenario is the same for both the runs. However, if you rebuilt the model, the data of the previous run will not be available.

The scen_rptdata_<*Scenario ID*> table is registered in the following metadata location: `/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Viral Effect Analysis/<Scenario ID>/Data Sources/Report Data`. This table is then loaded to the SAS Customer Link Analytics LASR Analytic Server as the scen_rpt_LASR_<*Scenario ID*> table. The LASR table is also registered in the same location. For more information about the reporting data and loading it to the SAS Customer Link Analytics LASR Analytic Server, see *SAS Customer Link Analytics: User's Guide*.

6 Update the scenario_batch_run_hist table.

If the reporting data is prepared successfully, a record from the scenario_batch_run_hist application data table is updated with the execution status for the given run number of the scenario. At any stage of the scoring wrapper execution, if the execution fails with errors, this record is updated with the corresponding status.

The scenario_batch_run_hist application data table stores information about all the modeling and scoring runs. The following table shows the sample records in this table.

Table 7.1 *scenario_batch_run_hist Table*

scenario_pk	scenario_run_num	project_run_num	scenario_model_pk	scenario_abt_history_end_date	scenario_execution_start_dttm	scenario_execution_end_dttm	scenario_status_cd
3	2		.		30JAN2014:08:08:41	.	ENBL
3	3		5		30JAN2014:08:11:07	30JAN2014:08:21:53	EXCSCS
3	4		5		30JAN2014:08:23:19	.	EXERR
3	5		5		30JAN2014:08:33:39	.	EXERR
3	6		5		30JAN2014:08:36:37	.	EXERR

scenario_pk	scenario_run_num	project_run_num	scenario_model_pk	scenario_abt_history_end_date	scenario_execution_start_dttm	scenario_execution_end_dttm	scenario_status_cd
3	7		5		30JAN2014:08:40:58	30JAN2014:08:41:04	EXCSCS
3	8		6		30JAN2014:08:44:46	.	EXERR

Part 3

Appendixes

Appendix 1

Global Parameters

Project-Specific Parameters

The project-specific parameters are stored in the PARAM_MSTR table and their values are stored in the PARAM_VALUE table. Both these tables are application data tables. For more information about these tables, see *SAS Customer Link Analytics: Data Reference Guide*.

Table A1.1 *Parameter Details*

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
log_clear	Specifies whether a new log file is created or log messages are appended in the existing log file.	N	new	
sia_aggr_trans_tbl_nm	Stores the name of the table that is produced as an output of the **Aggregated transactional data** enrichment category.	N	CLA_NO_CURR_D9	
sia_aggrtrns_nd_cat_cd	Stores the code for the **Aggregated transactional data** node-level data enrichment category.	N	NAGR	
sia_assoneignd_nd_cat_cd	Stores the code for the **Associations with neighboring roles** node-level data enrichment category.	N	NRLV	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_appl_debug_flg	Specifies whether a detailed log message is printed in SAS log files.	Y	Y	N
sia_appl_debug_options	Controls printing of detailed log messages in SAS log files depending on the value that you set for this parameter. You can specify one or more values that are mentioned below. Each value must be separated by a space. mlogic identifies the beginning and ending of macro execution, the values of macro parameters, and the values of conditional statements. mprint displays the SAS statements that are generated when a macro is run. symbolgen displays the results of resolving macro variable references. The details are logged in the file according to the combination of values that you specify for this parameter.	Y	mprint mlogic symbolgen	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_bsns_data_fldr_nm	Stores the name of folder of the metadata location that stores the business data. This folder is created in the following metadata location: `/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Projects/<Project ID>/Data Sources`.	N	Business Data	
sia_by_cluster_comm	Specifies that centrality measures are computed at community level.	N	COMMUNITY	
sia_by_cluster_net	Specifies that centrality measures are computed at network level.	N	NETWORK	
sia_chaq_var_tbl_nm	Stores the name of table that is produced as an output of the **Relation with churned and acquired nodes** data enrichment category.	N	CLA_CHACQ_VAR	
sia_chk_box_no	Stores the value N for check boxes.	N	N	
sia_chk_box_yes	Stores the value Y for check boxes.	N	Y	
sia_churnacqind_lnk_cat_cd	Stores the code for the **Churn and acquisition indicators** link-level data enrichment category.	N	LDER	
sia_churnacqind_nd_cat_cd	Stores the code for the **Churn and acquisition indicators** node-level data enrichment category.	N	NCQR	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_cla_der_ind_tbl_nm	Stores the name of the table that is produced as an output of the **Churn and acquisition indicators** data enrichment category.	N	CLA_DER_IND	
sia_cm_cd_auth	Stores the code that is assigned for the Authority centrality measure.	N	AUTHORITY	
sia_cm_cd_between	Stores the code that is assigned for the Betweenness centrality measure.	N	BETWEEN	
sia_cm_cd_close	Stores the code that is assigned for the Closeness centrality measure.	N	CLOSE	
sia_cm_cd_close_in	Stores the code that is assigned for the In-closeness centrality measure.	N	CLOSEIN	
sia_cm_cd_close_out	Stores the code that is assigned for the Out-closeness centrality measure.	N	CLOSEOUT	
sia_cm_cd_clust_coef	Stores the code that is assigned for the Clustering coefficient centrality measure.	N	CLUSTCOEF	
sia_cm_cd_dg	Stores the code that is assigned for the Degree centrality measure.	N	DEGREE	
sia_cm_cd_dg_in	Stores the code that is assigned for the In-degree centrality measure.	N	DEGREEIN	
sia_cm_cd_dg_out	Stores the code that is assigned for the Out-degree centrality measure.	N	DEGREEOUT	
sia_cm_cd_eigen	Stores the code that is assigned for the Eigenvector centrality measure.	N	EIGEN	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_cm_cd_hub	Stores the code that is assigned for the Hub centrality measure.	N	HUB	
sia_cm_cd_inflnce1	Stores the code that is assigned for the Influence 1 centrality measure.	N	INFLUENCE1	
sia_cm_cd_inflnce2	Stores the code that is assigned for the Influence 2 centrality measure.	N	INFLUENCE2	
sia_cm_round_precision	Stores the value for round precision of decimal point numbers.	Y	0.1	
sia_cmlvlstat_nd_cat_cd	Stores the code for the **Community-level statistics** node-level data enrichment category.	N	NCMV	
sia_comm_links_ds_flag	Stores the flag value for creating a community link data set in the Community Building workflow step. This data set describes the links between communities.	Y	N	Y
sia_comm_overlap_ds_flag	Stores the flag value for creating a community overlap data set in the Community Building workflow step. This data set describes the intensity of each node that belongs to multiple communities.	Y	N	Y
sia_comm_size_param_id	Stores the parameter ID for the community size used in the Community Building workflow step.	N	VALCOMMSIZE	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_creatrpt_prcd_cd	Stores the process code for creating the community report.	N	CTCOMMRPT	
sia_dataprep_prcs_cd	Stores the process code for the data enrichment process.	N	DATAPREP	
sia_datasrc_fldr_nm	Stores the name of folder of the metadata location that contains all the data sources. This folder is created in the following metadata location: **`/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Projects/<Project ID>`**.	N	Data Sources	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_de_job_exec_mode	Determines the mode of execution of the data enrichment process. Set up the execution mode as **Sequential** if you want SAS Customer Link Analytics to process the data enrichment process in the sequential order of the enrichment categories that you have selected. In this case, the processing of each individual enrichment category begins only after the processing of the previous category is complete. Select **Parallel** if you want SAS Customer Link Analytics to run the data enrichment process simultaneously for categories that are grouped together. In this case, the processing of enrichment categories that are grouped together runs in parallel followed by the processing of the next group of categories. In all, three such groups of categories are processed one after the other.	Y	Parallel	Sequential
sia_de_p1_tbl_nm	Stores the name of table that is produced as an output of the Data Extraction workflow step.	N	CLA_DE_P1	
sia_default_role_nm	Stores the name of the default role.	Y	Default	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_dflt_lnk_cat_cd	Stores the code for the default link-level data enrichment category, **Roles and Communities**.	N	LCLA	
sia_dflt_nd_cat_cd	Stores the code for the default node-level data enrichment category, **Roles and Communities**.	N	NCLA	
sia_dfltcomp_nd_cat_cd	Stores the code for the **Roles and communities over time** node-level data enrichment category.	N	NCMP	
sia_distance_type	Indicates the type of distance that is computed during the clustering procedure.	N	euclidean	
sia_dt_align_day	Specifies the date alignment for extracting the date in the number of days.	N	SAMEDAY	
sia_dt_align_month	Specifies date alignment for extracting the date in the number of months.	N	BEGIN	
sia_err_cd	Stores the error code value for an executed process.	N		
sia_explr_data_fldr_nm	Stores the name of the folder of the metadata location that stores the exploration data. This folder is created in the following metadata location: `/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Projects/<Project ID>/Data Sources`.	N	Exploration Data	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_flag_no	Stores N as the flag value.	N	N	
sia_flag_yes	Stores Y as the flag value.	N	Y	
sia_hpds2_commit_size	Stores the value of the commit parameter that is used in the HPDS2 procedure to load data into memory.	Y	10000000	user-specified
sia_imputation_method	Indicates the computation method that is used for the clustering procedure.	N	mean	
sia_in_degree_clmn_nm	Stores the column name of the In-degree centrality.	N	IN_DEGREE	
sia_in_degree_clmn_pk	Stores the primary key (PK) value of the In-degree column.	N	1	
sia_link_enrch_outtbl_nm	Stores the name of the link-level table that is produced as an output of the data enrichment process.	N	CLA_DP_LINK_LVL	
sia_link_lvl_lsr_optbltyp_cd	Stores the code for the type of link-level output table that is loaded into the SAS Customer Link Analytics LASR Analytic Server.	N	LKLSROP	
sia_link_lvl_optbltyp_cd	Stores the code of the output table for the link-level data enrichment table.	N	LKOP	
sia_lnf_p1_tbl_nm	Stores the name of the table that is produced as an output of the Link and Node Processing workflow step.	N	CLA_LNF_P1	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_lnk_wt_exp_param_id	Stores the parameter ID for the link weight expression.	N	LNKWTEXP	
sia_lknde_msr_max_up_lmt	Stores the maximum value for the upper limit that is defined in the Link and Node Processing workflow step.	N	10000000000	
sia_lknde_msr_min_low_lmt	Stores the minimum value for the lower limit that is defined in the Link and Node Processing workflow step.	N	0	
sia_load_lasr_ind	Identifies whether the enriched data will be loaded into the SAS Customer Link LASR Analytic Server during the batch execution of a project. The value 1 indicates that the data will be loaded into the server.	Y	1	0
sia_loaddata_prcs_cd	Stores the code for the process that copies the enriched node– level data to the SAS Customer Link LASR Analytic Server.	N	LDNDDATA	
sia_max_effect_no	Indicates the maximum number of variables that can be selected for variable reduction.	Y	40	
sia_max_iter_num	Indicates the maximum number of iterations that can be performed for the clustering procedure.	Y	100	
sia_navl	Stores the parameter ID for a null value.	N	#NAME?	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_ndattr_lnk_cat_cd	Stores the code for the **Node Attributes** link-level data enrichment category.	N	LATR	
sia_ndattr_nd_cat_cd	Stores the code for the **Node Attributes** node-level data enrichment category.	N	NATR	
sia_nod_attrb_tbl_typ_cd	Stores the code for the Node Attribute type of tables.	N	NDATTRB	
sia_node_enrch_outtbl_nm	Stores the name of the node-level table that is produced as an output of the data enrichment process.	N	CLA_DP_NODE_LVL	
sia_node_lvl_lsr_optbltyp_cd	Stores the code for the output table type of the node-level table that is loaded into the SAS Customer Link Analytics LASR Analytic Server.	N	NDLSROP	
sia_node_lvl_oplsr_tbl_nm	Stores the name of the node-level table that is produced as an output of loading data into SAS Customer Link Analytics LASR Analytic Server.	N	CLA_ND_LVL_LASR	
sia_node_lvl_optbltyp_cd	Stores the output table type code for a node-level enriched table.	N	NDOP	
sia_openrpt_prcs_cd	Stores the code for the open community report process.	N	OPNCOMMRPT	
sia_out_degree_clmn_nm	Stores the value of the column name of the Out-degree column.	N	OUT_DEGREE	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_out_degree_clmn_pk	Stores the PK value of the Out-degree column.	N	2	
sia_parallel_exec_mode	Stores the value for the parallel mode of execution of the data enrichment process.	Y	PARALLEL	
sia_param_id_btwnnormtyp	Stores the parameter ID for the approach that is used for computing betweenness.	N	BTWNNORMTYP	
sia_param_id_chkauth	Stores the parameter ID this is used for the **Authority** check box displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKAUTH	
sia_param_id_chkbtwn	Stores the parameter ID that is used for the **Betweenness** check box displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKBTWN	
sia_param_id_chkclscoeff	Stores the parameter ID that is used for the **Clustering coefficient** check box displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKCLSCOEFF	
sia_param_id_chkclsn	Stores the parameter ID that is used for the **Closeness** check box that is displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKCLSN	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_param_id_chkcommsegm	Indicates the parameter ID of the parameter that indicates whether the **Create community segments** check box is selected in the Centrality Measure Computation workflow step.	N	CHKCOMMSEG	
sia_param_id_chkcommsize	Stores the parameter ID that is used for checking community size in the Community Building workflow step.	N	CHKCOMMSIZE	
sia_param_id_chkdeg	Stores the parameter ID that is used for the **Degree** check box that is displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKDEG	
sia_param_id_chkdiam	Stores the parameter ID that is used for checking the diameter in the Community Building workflow step.	N	CHKDIAM	
sia_param_id_chkeigen	Stores the parameter ID that is used for the **Eigenvector** check box that is displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKEIGEN	
sia_param_id_chkhub	Stores the parameter ID that is used for the **Hub** check box that is displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKHUB	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_param_id_chkinfl	Stores the parameter ID that is used for the **Influence** check box that is displayed in the UI of the Centrality Measures Computation workflow step.	N	CHKINFL	
sia_param_id_clsnnopathtyp	Stores the parameter ID for the approach that is used for computing closeness of disconnected nodes.	N	CLSNNOPATHTYP	
sia_param_id_commbldaprch	Stores the parameter ID for the approach that is used for the community–building process.	N	COMMBLDAPRCH	
sia_param_id_commdiambywght	Stores the parameter ID for the approach that is used for computing community diameter in the summary report of the Community Building workflow step.	N	COMMDIAMBYWGHT	
sia_param_id_commsizediamrel	Stores the parameter ID for defining the relationship between community size and diameter in the Community Building workflow step.	N	COMMSIZEDIAMREL	
sia_param_id_complvltyp	Stores the parameter ID of another parameter that specifies whether centrality is computed by community or by network.	N	COMPLVLTYP	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_param_id_convergcriterion	Stores the parameter ID of the parameter that stores the value of the convergence criterion.	N	CONVERGCRITERION	
sia_param_id_datahst	Stores the parameter ID for data history for data extraction from the source event detail records table.	N	DATAHST	
sia_param_id_datahstday	Stores the parameter ID of the parameter that stores the number of days for which the history data needs to be extracted.	N	DATAHSTDAY	
sia_param_id_datatodt	Stores the parameter ID for the To Date column for data extraction from the source xDR table.	N	DATATODT	
sia_param_id_degtyp	Stores the parameter ID for the **Degree** check box that is displayed in the UI.	N	DEGTYP	
sia_param_id_eigenalgotyp	Stores the parameter ID for the algorithm that is used for computing the Eigenvector centrality.	N	EIGENALGOTYP	
sia_param_id_elbowcriterion	Stores the parameter ID of the parameter that stores the value of the elbow criterion.	N	ELBOWCRITERION	
sia_param_id_graphdirtyp	Stores the parameter ID of another parameter that specifies the graph direction type of the source data.	N	GRAPHDIRTYP	
sia_param_id_inclst	Stores the parameter ID for the inclusion list.	N	INCLST	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_param_id_lnkrmvratio	Stores the parameter ID for the link removal ratio in the community-building process.	N	LNKRMVRATIO	
sia_param_id_maxcluster	Stores the parameter ID of the parameter that stores the maximum number of clusters.	N	MAXCLUSTER	
sia_param_id_resvalentered	Stores the parameter ID for values entered for resolution lists in the Community Building workflow step.	N	RESVALENTERED	
sia_param_id_resvalselected	Stores the parameter ID for selected values of resolutions in the Community Building workflow step.	N	RESVALSELECTED	
sia_param_id_showcommbldresult	Stores the parameter ID for showing the result in the Community Building workflow step.	N	SHOWCOMMBLDRESULT	
sia_param_id_valcommsize	Stores the parameter ID for the value of the community size used in the Community Building workflow step.	N	VALCOMMSIZE	
sia_param_id_valdiam	Stores the parameter ID for the value of the diameter used in the Community Building workflow step.	N	VALDIAM	
sia_param_val_cd_commbld_bua	Stores the parameter ID for the bottom-up approach used in the Community Building workflow step.	N	BUA	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_param_val_cd_commbld_tda	Stores the parameter ID for the top-down approach used in the Community Building workflow step.	N	TDA	
sia_po_cd_algo	Stores the name of the community-building algorithm.	Y	PARALLEL_LABEL_PROP	
sia_po_graph_int_fmt	Specifies the internal graph format of the source data to be used by the OPTGRAPH procedure algorithms.	Y	THIN	FULL
sia_po_loglevel	Controls the amount of information that is displayed in the SAS log as a result of calling the OPTGRAPH procedure.	Y	1	0 2 3
sia_po_max_iter	Specifies the maximum number of iterations that are permissible in the algorithm of the community-building process.	Y	100	
sia_po_nthreads	Specifies the number of threads that the procedure can use.	Y	8	
sia_prcs_dt_colm_typ_cd	Stores the code for the table column type for the Process Date column.	N	PRCSDT	
sia_prm_val_grph_drctd	Stores the parameter ID for the directed graph data.	N	DIRECTED	
sia_prm_val_grph_undrctd	Stores the parameter ID for the undirected graph data.	N	UNDIRECTED	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_prm_val_xctn_md_dstrbtd	Stores the parameter ID for the distributed mode of execution.	N	DISTRIBUTED	
sia_prm_val_xctn_md_nndstrbtd	Stores the parameter ID for the non-distributed mode of execution.	N	NONDISTRIBUTED	
sia_ra_p1_tbl_nm	Stores the name of the table that is produced as an output of the Role Assignment workflow step.	N	CLA_RA_P1	
sia_rlnchurn_nd_cat_cd	Stores the code for the **Relation with churned and acquired nodes** node-level data enrichment category.	N	NDER	
sia_rpt_data_fldr_nm	Stores the name of the folder of the metadata location in which the community reports data is stored. This folder is created in the following metadata location: **`/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Projects/<Project ID>/Data Source`**.	N	Report Data	
sia_rpt_fldr_nm	Stores the name of the folder of the metadata location in which the community reports are stored. This folder is created in the following metadata location: **`/Shared Data/SAS Customer Link Analytics/Cust Link Analytics 6.5/Projects/<Project ID>`**.	N	Reports	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_rptvar_id_avgdenscomm	Stores the parameter ID for the community reporting variable, Average density of communities.	N	AVGDENSCOMM	
sia_rptvar_id_avgdiamcomm	Stores the parameter ID for the community reporting variable, Average diameter of communities.	N	AVGDIAMCOMM	
sia_rptvar_id_avgnumnodcomm	Stores the parameter ID for the community reporting variable, Average number of nodes in communities.	N	AVGNUMNODCOMM	
sia_rptvar_id_linkrmvd	Stores the parameter ID of the link and node processing reporting variable, Number of links removed.	N	LINKRMVD	
sia_rptvar_id_maxdenscomm	Stores the parameter ID for the community reporting variable, Maximum density of communities.	N	MAXDENSCOMM	
sia_rptvar_id_maxdiamcomm	Stores the parameter ID for the community reporting variable, Maximum diameter of communities.	N	MAXDIAMCOMM	
sia_rptvar_id_maxnumnodcomm	Stores the parameter ID for the community reporting variable, Maximum number of nodes in communities.	N	MAXNUMNODCOMM	
sia_rptvar_id_mindenscomm	Stores the parameter ID for the community reporting variable, Minimum density of communities.	N	MINDENSCOMM	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_rptvar_id_mindiamcomm	Stores the parameter ID for the community reporting variable, Minimum diameter of communities.	N	MINDIAMCOMM	
sia_rptvar_id_minnumnodcomm	Stores the parameter ID for the community reporting variable, Minimum number of nodes in communities.	N	MINNUMNODCOMM	
sia_rptvar_id_modnet	Stores the parameter ID for the community reporting variable, Modularity.	N	MODNET	
sia_rptvar_id_nodermvd	Stores the parameter ID of the link and node processing reporting variable, Number of nodes removed.	N	NODERMVD	
sia_rptvar_id_numcommnet	Stores the parameter ID for the community reporting variable, Number of communities in network.	N	NUMCOMMNET	
sia_rptvar_id_numdupsrctrn	Stores the parameter ID for the reporting variable, Duplicate records in transaction table that is used in the link and node processing workflow step.	N	NUMDUPSRCTRN	
sia_rptvar_id_numdupndeinc	Stores the parameter ID for the reporting variable, Duplicate nodes in the node inclusion list that is used in the link and node processing workflow step.	N	NUMDUPNDEINC	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_rptvar_id_numduplnkinc	Stores the parameter ID for the reporting variable, Duplicate nodes in the link inclusion list that is used in the link and node processing workflow step.	N	NUMDUPLNKINC	
sia_rptvar_id_numlink	Stores the parameter ID for the reporting variable, Number of links that is used in the link and node processing workflow step.	N	NUMLINK	
sia_rptvar_id_numnode	Stores the parameter ID for the reporting variable, Number of nodes that is used in the link and node processing workflow step.	N	NUMNODE	
sia_sequential_exec_mode	Stores the value for the sequential mode of execution of the data enrichment process.	Y	SEQUENTIAL	
sia_smd_err_key	Stores the error code value that is defined in the sia_smd_error_tbl table.	N		
sia_smd_error_tbl	Stores the name of the table that contains the error message details.	N	sia_smd_error	
sia_sql_ip_trace_flg	Specifies whether SQL trace messages are printed in the SAS log files.	Y	Y	N
sia_standardize_method	Indicates the standardization method that is used for the clustering procedure.	N	std	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_stop_criterion	Indicates the name of the stopping criterion that is used in the clustering procedure.	N	cluster_change	
sia_text_separator	Separates the texts in the strings.	N	:	
sia_var_sel_techniq ue	Indicates the technique that is used for variable selection.	N	VarianceAnalysis	
sia_variance_explai ned	Stores the percentage of variance that needs to be explained by the variable reduction process.	Y	0.95	
sia_wrkflw_status_ disable	Stores the code for the Disabled state of a workflow step.	N	DSBL	
sia_wrkflw_status_ edit	Stores the code for the Edited state of a workflow step.	N	EDT	
sia_wrkflw_status_ enable	Stores the code for the Enabled state of a workflow step.	N	ENBL	
sia_wrkflw_status_ error	Stores the code for the Error occurred state of a workflow step.	N	EXERR	
sia_wrkflw_status_ exectd	Stores the code for the Successfully executed state of a workflow step.	N	EXCSCS	
sia_wrkflw_status_i nprgs	Stores the code for the In progress state of a workflow step.	N	INPRGS	
sia_wrkflw_step_id _cmbld	Stores the code for the Community Building workflow step.	N	CMBLD	
sia_wrkflw_step_id _cntrmsr	Stores the code for the Centrality Measures Computation workflow step.	N	CNTRMSR	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_wrkflw_step_id_dxtr	Stores the code for the Data Extraction workflow step.	N	DXTR	
sia_wrkflw_step_id_lnkndpr	Stores the code for the Link and Node Processing workflow step.	N	LNKNDPR	
sia_wrkflw_step_id_rlass	Stores the code for the Role Assignment workflow step.	N	RLASS	
sia_xDR_aggr_type_day	Stores the aggregation type code for the daily aggregated source xDR table.	N	DY	
sia_xDR_aggr_type_fll	Stores the parameter ID for the fully aggregated source xDR table.	N	FLL	
sia_xDR_aggr_type_mth	Stores the aggregation type code for the monthly aggregated source xDR table.	N	MTH	
sia_xDR_aggr_type_wk	Stores the aggregation type code for the weekly aggregated source xDR table.	N	WEEK	
sia_xDR_dt_dtime_cd	Stores the parameter ID for the data type of the datetime variable in the source xDR table.	N	DATETIME	
sqlrc	Stores the return code of an SQL operation.	N		
tbl_clmn_type_cd_dgin	Stores the column code for the In-degree column.	N	DGIN	
tbl_clmn_type_cd_dgout	Stores the column code for the Out-degree column.	N	DGOUT	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
tbl_clmn_type_cd_frm	Stores the column type code for the From column in source table.	N	FRM	
tbl_clmn_type_cd_msr	Stores the column type code for the Measure column in the source table.	N	MSR	
tbl_clmn_type_cd_nd	Stores the column type code for the Node column in the source table.	N	NODE	
tbl_clmn_type_cd_to	Stores the column type code for the To column in the source table.	N	TO	
tbl_clmn_type_cd_trndt	Stores the column type code for the Transaction date column in the source table.	N	TRNDT	
tbl_cm_type_cd_cm	Stores the code for the table that is produced as an output of the Centrality Measures Computation workflow step.	N	CMSROP	
tbl_cm_type_cd_cs	Stores the code for the table that contains the community statistics.	N	COMSTOP	
tbl_cm_type_cd_de	Stores the code for the table that contains the output produced by the Data Extraction workflow step.	N	DETRANS	
tbl_cm_type_cd_lnf	Stores the code for the table that contains the output produced by the Link and Node Processing workflow step.	N	LNFTRANS	
tbl_type_cd_lnkinclist	Stores the table type code for the link inclusion list table.	N	LNKINCLLST	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
tbl_type_cd_ndinclist	Stores the table type code for the node inclusion list table.	N	NDINCLLST	
tbl_type_cd_trn	Stores the table type code for the transaction xDR table.	N	TRN	

Parameters for Viral Effect Analysis

The parameters that are required for viral effect analysis are stored in the SCENARIO_ PARAM_MSTR table and their values are stored in the SCENARIO_PARAM_VALUE table. The parameters that you configure are stored in the SCENARIO_ PARAM table. All these tables are application data tables. For more information about these tables, see *SAS Customer Link Analytics: Data Reference Guide*.

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_analyt_data_fldr_nm	Stores the name of the metadata folder that contains the analytical data required for viral effect analysis.	N	Analytics Data	
sia_default_proj_pk	Stores the default value for project PK for scenarios that are not linked to a project created in SAS Customer Link Analytics.	N	-999	
sia_em_project_nm	Name of the SAS Enterprise Miner project that is built for the current scenario. This name is generated automatically and you must not change it.	N		
sia_entity_cd_cm	Stores the code for the centrality measure entity.	N	CMSR	
sia_entity_cd_role	Stores the code for the role entity.	N	ROLE	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_event_count_cat_cd	Stores the code for the event-based ABT variable category.	N	EBCV	
sia_event_date_col_nm	Name of the column from the node attribute table that stores the event date.	Y		user-specified
sia_event_ind_cat_cd	Stores the code for the ABT variable category that computes the event indicator.	N	TRGT	
sia_event_usgmr_cat_cd	Stores the code for the usage-based ABT variable category.	N	EBUM	
sia_id_var_nm	Column that identifies a unique row of the node attribute table.	Y	NODE_ID	user-specified
sia_mod_abt_nm	Name of the modeling ABT that is built when the stored process for building an ABT is executed.	Y		user-specified
sia_mod_abt_param_nm	Stores the name of the parameter that contains the name of the modeling ABT.	N	sia_mod_abt_nm	
sia_mod_abt_tbl_typ_cd	Indicates the code that is assigned for a modeling ABT.	N	MDLABT	
sia_model_type	Type of model that is built for a given scenario using SAS Rapid Predictive Modeler. You can build basic, intermediate, or advanced types of models for a given scenario.	Y	Intermediate	Basic Advanced

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_node_attrib_tbl_nm	Name of the node attribute table that is used for a given scenario. This table name is stored in the form of libname.tablename. This table stores information about each node and its event date.	Y		user-specified
sia_obs_window_length	Length of the observation window in days. This length identifies whether a particular event occurred for the given node. For example, in this time period, the churn behavior of the nodes is observed.	Y	0	user-specified
sia_observ_win_event_nm	Stores the name of the table that contains the information about the events that occurred in the observation period.	N	observ_win_event	
sia_param_id_event_dt	Stores the parameter ID of the parameter that contains the event date column.	N	sia_event_date_col_nm	
sia_param_id_id_var_nm	Stores the parameter ID of the parameter that stores the column name of the unique entity ID in the event details table.	N	sia_id_var_nm	
sia_param_id_model_type	Stores the parameter ID of the parameter that stores the type of the model produced by SAS Rapid Predictive Modeler.	N	sia_model_type	
sia_param_id_node_attrb	Stores the ID of the parameter that stores the name of the event details table.	N	sia_node_attrib_tbl_nm	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_param_id_obs_wndo_length	Stores the ID of the parameter that stores the length of the observation window.	N	sia_obs_window_length	
sia_param_id_perf_wndo_length	Stores the ID of the parameter that stores the length of the performance window.	N	sia_perf_window_length	
sia_param_id_prod_id	Stores the ID of the parameter that stores the product IDs.	N	sia_prod_id	
sia_param_id_prod_id_colm	Stores the ID of the parameter that stores the product ID column name.	N	sia_prod_id_colm_nm	
sia_param_id_proj_locn	Stores the ID of the parameter that stores the location of a SAS Enterprise Miner project.	N	sia_project_location	
sia_param_id_proj_name	Stores the ID of the parameter that stores the name of a SAS Enterprise Miner project.	N	sia_em_project_nm	
sia_param_id_scen_end_dt	Stores the ID of the parameter that stores the end date of the scenario data.	N	sia_scen_src_end_dttm	
sia_param_id_scen_start_dt	Stores the ID of the parameter that stores the start date of the scenario data.	N	sia_scen_src_start_dttm	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_perf_window_length	This length identifies whether a particular event occurred for the given node. For example, in this time period, the impact of nodes (that have churned in the observation window) on the remaining nodes is observed. The impact is studied in this window. Also, the impact is considered in terms of whether the remaining nodes churned in the performance window. SAS Customer Link Analytics computes the length of the performance window while building the modeling ABT.	Y	0	user-specified
sia_prev_abb_nm	Stores the abbreviation used for the reporting column that indicates a previous value.	N	PREV	
sia_prod_id	Stores the product ID for which the scenario is defined. This parameter is applicable for viral product adoption.	Y		user-specified
sia_prod_id_colm_nm	Stores the column name of the event details table that stores the product IDs. This parameter is applicable for viral product adoption.	Y		

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_project_location	Location on the physical file system in which the SAS Enterprise Miner project workspace that is created for each model created by SAS Rapid Predictive Modeler is stored.	Y		user-specified
sia_scdatasrc_fldr_nm	Indicates the name of the folder that is created in the SAS metadata location for viral effect analysis. This folder further contains various subfolders.	N	Data Sources	
sia_scen_src_end_dttm	End date for extracting source data required for ABT building.	Y		
sia_scen_src_start_dttm	Start date for extracting source data required for ABT building.	Y		
sia_scen_type_churn	Indicates the code that is assigned for the **Churn** scenario type.	N	CHRN	
sia_scen_type_prda	Indicates the code that is assigned for the **Product adoption** scenario type.	N	PRDA	
sia_scmodl_fldr_nm	Indicates the name of the folder that is created in physical file system for viral effect analysis. This folder stores the model that the SAS Rapid Predictive Modeler macro creates and registers in SAS metadata.	N	Models	
sia_scored_abt_tbl_typ_cd	Indicates the code that is assigned for a scored ABT.	N	SCRDABT	
sia_scoring_abt_tbl_typ_cd	Indicates the code that is assigned for a scoring ABT.	N	SCRABT	

Parameter ID	Description	Is Editable	Default Value	Other Possible Values
sia_scr_data_fldr_nm	Indicates the name of the folder that is created in the SAS metadata location for each viral effect analysis scenario. This folder stores the output that the scoring process produces.	N	Score Data	
sia_scrpt_data_fldr_nm	Indicates the name of the folder that is created in the SAS metadata location for each viral effect analysis scenario. This folder stores the data that is required for generating SAS Visual Analytics reports.	N	Report Data	
sia_scrpt_fldr_nm	Indicates the name of the folder that is created in the SAS metadata location for each viral effect analysis scenario. This folder stores the SAS Visual Analytics reports that are created to analyze the results of viral effect analysis.	N	Reports	

Appendix 2

Quality Checks for Source Data

Inclusion Lists

If you intend to use an inclusion list in the Data Extraction workflow step, make sure that the associated table does not include duplicate key columns. That is, for a node inclusion list, make sure that it does not contain duplicate node IDs. Similarly, for a link inclusion list, make sure that it does not contain duplicate links.

Node Attribute Tables

If you intend to include a node attribute table in a source data profile, you must verify that the node ID is unique in this table. Otherwise, when the business user or the network analyst runs the data enrichment process for the node-level or link-level **Node attributes** data enrichment category, the output tables will contain duplicate records for the same node ID. When this information is used for example to enhance campaign definitions or explore communities, it might result in incorrect business implications.

Appendix 3

Updating Host Name References

Redeploying the Loop Job

If you run the **Update Host Name References** option that is available in the SAS Deployment Manager, you have to redeploy the sialoopjob job. For more information, see “Deploy the Loop Job for Data Enrichment” on page 26.

Creating a New SAS Customer Link Analytics LASR Analytic Server

If you run the **Update Host Name References** option that is available in the SAS Deployment Manager, you have to create a new SAS Customer Link Analytics LASR Analytic Server in the metadata.

To create a new server:

1. On the **Plug-ins** tab of SAS Management Console, right-click **Server Manager**, and then select **New Server**.
2. Select **SAS LASR Analytic Server** from the **SAS Servers** list. Click **Next**.
3. Enter the name and description of the server. Click **Next**.
4. Specify the same value for each of the server properties that you configured for the previous server. Click **Next**.
5. Specify the same value for each of the connection properties that you configured for the previous server. Click **Next**.

   Note: In the previous wizard page, if you selected **No** for **Single machine server**, then specify the host name on which you have installed the SAS Customer Link Analytics LASR Analytic Server grid. Otherwise, specify the host name of the server that you are defining.
6. Set the same metadata permissions that you defined for the previous server.
7. Click **Finish**.
8. From **Data Library Manager**, right-click the **Customer Link Analytics LASR** library.
9. On the **Data Server** tab, from the **Database Server** list, select the new server that you created.
10. Click **OK**.

Appendix 4

Troubleshooting

Troubleshooting Error Messages in the Log File

The following table lists the errors that you might encounter when you run the Community Building and Centrality Measures Computation workflow steps in distributed mode. You can review the probable reason that is mentioned and take appropriate action to resolve the problem. These errors are logged in a log file. For details, see “Log Files for Projects and Scenarios” on page 46.

Table A4.1 *Error Messages*

Error Message in Log File	Probable Reason
bash: /opt/v940m1/laxno/TKGrid1/tkmpirsh.sh: No such file or directory time-out waiting for grid connection. ERROR: Failed to enumerate available compute nodes in the distributed computing environment. ERROR: Failed to open TKGrid library. ERROR: The bridge for SAS High-Performance Analytics encountered an internal error.	The SAS High-Performance Analytics grid installation location is not specified correctly.
Unable to open connection: Cannot resolve address. Time-out waiting for grid connection. ERROR: Failed to enumerate available compute nodes in the distributed computing environment. ERROR: Failed to open TKGrid library. ERROR: The bridge for SAS High-Performance Analytics encountered an internal error.	The SAS High-Performance Analytics grid server name is not specified correctly.
ERROR: Connect to HPA failed, connect_rc = -1 ip = 127.0.0.1 port = -26755. Double check your GRIDMODE setting. ERROR: GRIDdata BulkOperations errort_rc = -1 ip = 127.0.0.1 port = -26755. Double check your GRIDMODE setting. ERROR: Unable to add rows to table.rt_rc = -1 ip = 127.0.0.1 port = -26755. Double check your GRIDMODE setting.	The SAS High-Performance Analytics grid mode is not set up correctly.
Unable to use key file “<user home directory>/.ssh/id_rsa” (unable to open file) Time-out waiting for grid connection. ERROR: Failed to enumerate available compute nodes in the distributed computing environment. ERROR: Failed to open TKGrid library. ERROR: The bridge for SAS High-Performance Analytics encountered an internal error.	The .SSH key is not set up correctly.

Troubleshooting the Performance of the Data Extraction Workflow Step

Problem statement

The source transactional data is located in the same Teradata or Hadoop (Hive) database in which the data that is extracted in the Data Extraction workflow step is stored. However, the workflow step execution takes longer.

Suggested solution

This problem might occur if the Data Extraction workflow step is not running as an in-database process.

Perform the following steps to resolve the problem:

1. Check the properties of the external library that you have created for the source data.
2. Compare these properties with the properties of the following libraries depending on the database type:
   - For Teradata, make sure that these properties are the same as the properties of the sia_bdop and sia_bdim libraries.
   - For Hadoop (Hive), make sure that these properties are the same as the properties of the sia_bdm_hive library.

   If they are not, change the properties of the external library as required.
3. Confirm that the Data Extraction workflow step runs as an in-database process and its execution time shows a performance gain.

Tuning Recommendation for Using PostgreSQL

If you have opted for a middle-tier cluster for deployments that have multiple cluster nodes, then you need to configure the connection tuning parameter for PostgreSQL. To implement this recommendation for a large database, in the default postgresql.conf file, change the value of the max_connections parameter to 512. The postgresql.conf file is available in the following location: `<SAS configuration directory>/Lev1/WebInfrastructurePlatformDataServer/data`. A restart is required after you change this value.

Troubleshooting the Problem of Insufficient Memory

Problem statement

While operating in non-distributed mode, the Community Detection or the Centrality Measures Computation workflow step might fail because insufficient memory was logged in the log file.

Suggested solution

To resolve this out-of-memory problem, open the sasv9.cfg file, which is available in the following location: `<SAS Home>/SASFoundation/9.4/nls/en`. Verify the values that you have specified for the following SAS system options:

- WORK
- MEMSIZE
- SORTSIZE

Change the value of each of these system options depending on the size of the data that is being processed.

Troubleshooting Memory Issues for Parallel Sessions of the Data Enrichment Process

Problem statement

There is a lack of available memory when the data enrichment process runs in parallel mode. This problem can arise if the business data is stored in the SAS database.

Suggested solution

To resolve the memory availability issue, you need to control the number of sessions that run in parallel.

To control the number of sessions that run in parallel, complete the following steps:

1. Connect to SAS Data Integration Studio with administrative privileges.
2. On the **Folders** tab, expand **Products** ▸ **SAS Customer Link Analytics** ▸ **Cust Link Analytics 6.5** ▸ **Jobs**.
3. Double-click the **sialoopjob** job.
4. Double-click the **sia_inner_loopjob** transform.
5. Right-click **loop_parallel_cla** and select **Properties**.
6. On the **Loop Options** tab, select any one of the following options for **Maximum number of concurrent processes**:

One process for each available CPU node
Indicates that a single session runs on each available CPU node.

Use this number
Indicates the number of sessions that are running in parallel. Specify the exact number of sessions that you want to run in parallel. Make sure that the value that you enter is not greater than 6.

7 Redeploy the sialoopjob job. For more information, see "Deploy the Loop Job for Data Enrichment" on page 26.

Troubleshooting the Data Enrichment Processing Time

Problem statement

The data enrichment process is running for a long time and the project status remains as **Data enrichment in progress**.

Suggested solution

The data enrichment process could be running so long because one of the servers might have stopped running. To verify this possibility, complete the following steps:

1 Note the project ID of the project for which you are running the data enrichment process.

2 For this project ID, search for the record in the project_process_status application data table for which the value of the process_cd column is DATAREP.

3 Change value of the process_status_cd column to ENBL.

4 Rerun the data enrichment process.

5 Check the status of the process. If the process still continues to run for a long time, verify if any of the servers such as the SAS Workspace Server or the SAS Stored Process Server have stopped running. If so, then restart the server and repeat steps 2 through 4.

Troubleshooting the Return Code Error during Data Enrichment Execution in the Hadoop (Hive) Environment

Problem statement

While executing the data enrichment process in the Hadoop (Hive) environment, the following error can be encountered:

```
ERROR: java.sql.SQLException: Error while processing
statement: FAILED: Execution Error, return code 2 from
org.apache.hadoop.hive.ql.exec.mr.MapRedTask

ERROR: Unable to execute Hadoop query.

ERROR: java.sql.SQLException: Error while processing
statement: FAILED: Execution Error, return code 2 from
org.apache.hadoop.hive.ql.exec.mr.MapRedTask

ERROR: Unable to execute Hadoop query.
```

Suggested solution

You can try to resolve this problem by increasing the default values of the following YARN parameters:

- Map Task Maximum Heap Size (mapreduce.map.java.opts.max.heap)
- Map Task Memory (mapreduce.map.memory.mb)

Troubleshooting Multi-User Access of the SAS Customer Link Analytics LASR Analytic Server in the Hadoop (Hive) Environment

Problem statement

A metadata user who has administrative rights might have started the SAS Customer Link Analytics LASR Analytic Server. However, if another user tries to access the server while processing the Community Building workflow step, then the workflow step fails to execute. This can occur only in a Hadoop (Hive) environment.

Suggested solution

To resolve this problem, make sure that the administrative user starts the SAS Customer Link Analytics LASR Analytic Server with the following option: `SERVERPERMISSIONS=764`.

Troubleshooting the Failure of Loading Data into SAS Customer Link Analytics LASR Analytic Server for a Multi-Machine Deployment

Problem statement

The process of loading data into the SAS Customer Link Analytics LASR Analytic Server fails if both the SAS Customer Link Analytics LASR Analytic Server and the SAS Customer Link Analytics Server are deployed on different Application Server Contexts.

Suggested solution

If the SAS Customer Link Analytics LASR Analytic Server and the SAS Customer Link Analytics Server are deployed on different Application Server Contexts, then add the additional server context of the SAS Customer Link Analytics Server to the SAS Customer Link Analytics LASR Analytic Server library. To do so, complete the following steps:

1. Open SAS Management Console.
2. On the **Plug-ins** tab, expand **Environment Management** ▸ **Data Library Manager** ▸ **Libraries**.
3. Right-click the **Customer Link Analytics LASR** library, and select **Properties**.
4. On the **Assign** tab, from the **Available servers** list, select the server context that you specified for the SAS Customer Link Analytics Server, and add it to the **Selected servers** list.
5. Click **OK**.

Troubleshooting the Validation Failure of the SAS Connect Server and Others

Problem statement

After you install SAS Customer Link Analytics and proceed to validate the SAS Connect Server and other servers that are available under **SASApp** in SAS Management Console, the validation fails.

Suggested solution

During the validation, the SAS Connect Server or the other servers try to connect to the SAS Customer Link Analytics LASR Analytic Server library. However, because the SAS Customer Link Analytics LASR Analytic Server has not been started yet, the validation fails. Therefore, you must validate the servers that are available under SASApp only after you have successfully started the SAS Customer Link Analytics LASR Analytic Server.

Troubleshooting the Failure to Import Tables from a Customized Hadoop (Hive) Library

Problem statement

SAS Customer Link Analytics fails to import tables from the Hadoop (Hive) library if the default location of the temporary directory for Hadoop (Hive) is changed.

Suggested solution

The default location of the temporary directory for Hadoop (Hive) is **/tmp**. If you change this location in the Hadoop (Hive) setup, then you have to modify the LIBNAME statement in the sia_post_init macro. In the Windows environment, this macro is available in the following location: **<*SAS Home*>/SASFoundation/9.4/clasvr/ucmacros**. For example, you change the temporary directory to **/tmphadoop**. In this case, you have to add the following code to the LIBNAME statement of the Hadoop (Hive) library:

```
HDFS_TEMPDIR="/tmphadoop"
```

Troubleshooting In-Database Query Processing during Data Enrichment Execution in the Hadoop (Hive) Environment

Problem statement

In the Hadoop (Hive) environment, while you are executing the data enrichment process on larger volumes of data, queries might not execute inside the database. This might increase the processing time of data enrichment.

Suggested solution

You can resolve this problem by increasing the default value of the parameter mapred.child.java.opts (a maximum memory of the child Java process).

Troubleshooting the Failure of Processes Due to Too Many Concurrent Connections to the Hive Server

Problem statement

Some of the processes such as data enrichment or building modeling ABT for viral effect analysis might fail to execute. This might happen if there are too many concurrent connections to the Hive Server while these running processes. For example, you might encounter this problem while running data enrichment for all categories or while building the modeling ABT for viral effect analysis with more number of roles.

Suggested solution

In order to resolve this problem, restart the Hive Server and re-run these processes.

Troubleshooting the Reporter Node Failure for the Chinese Locale

Problem statement

While you are building an analytical model using SAS Rapid Predictive Modeler in the Chinese locale (zh_CN) environment, the model building process fails when the reporter node is executed.

Suggested solution

The error occurs because DBCS parentheses are used instead of ASCII parentheses while defining error messages for SAS Enterprise Miner.

To resolve the error:

1 In Base SAS, define a library that points to the `<SAS Home Path>\SASFoundation\9.4\dmine\sashelp` location in your environment.

2 Open the Dmine data set that is available in the library. Make sure that you open the data set in Edit mode.

3 Search for the following values for the zh_CN locale and replace the values as mentioned in the following table.

Note: Make sure that you use the ASCII parentheses while replacing the text.

Table A4.2 Search and Replace Column Key Values

Search Text	Replace Text
rpt_rptlift10_vlabel	\u63d0\u5347\u5ea6(10%%)
rpt_rptcumlift10_vlabel	\u7d2f\u79ef\u63d0\u5347\u5ea6(10%%)
rpt_rptcaptresp10_vlabel	\u6355\u83b7\u54cd\u5e94(10%%)
rpt_rptcumcaptresp10_vlabel	\u7d2f\u79ef\u6355\u83b7\u54cd\u5e94(10%%)

4 Save the changes.

Troubleshooting the Display of All Columns of a Table

Problem statement:

In certain scenarios, tables that are available in the UI might not display all the columns. For example, in the Projects workspace, the Project list might display only two columns, **Status** and **Date Modified**.

Suggested solution:

To view all the columns of a table:

1 On the toolbar, click . The Manage Columns window appears.

2 From the **Available columns** list, move all the column names to the **Displayed columns** list. Alternatively, you can move only those columns that you want to view in the table.

3 Click **OK**.

Recommended Reading

SAS Customer Link Analytics is supported by the following documents:

- *SAS Customer Link Analytics: User's Guide* is written for users who want to create projects in SAS Customer Link Analytics and run workflow steps.
- *SAS Customer Link Analytics: Data Reference Guide* is written for users who want to understand the details about application data and business data tables.
- *SAS Customer Link Analytics: Upgrade and Migration Guide* is written for users who want to upgrade or migrate to SAS Customer Link Analytics 6.5.

Other relevant documents include the following:

- *SAS Intelligence Platform: Installation and Configuration Guide* is for system administrators who need to install and configure SAS products that use the metadata server.
- *SAS High-Performance Analytics Infrastructure: Installation and Configuration Guide* is for system administrators who need to install and configure SAS High-Performance Analytics Infrastructure.

For a complete list of SAS publications, go to sas.com/store/books. If you have questions about which titles you need, please contact a SAS Representative:

SAS Books
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-0025
Fax: 1-919-677-4444
Email: sasbook@sas.com
Web address: sas.com/store/books

Glossary

ABT variable
See analytical base table variable.

analytical base table
a highly denormalized data structure that is designed to build an analytical model or to generate scores based on an analytical model.

analytical base table variable
a column in an analytical base table that is used to build a statistical model to predict defaults.

analytical model
a statistical model that is designed to perform a specific task or to predict the probability of a specific event.

box plot
a graphical display of five statistics (the minimum, lower quartile, median, upper quartile, and maximum) that summarize the distribution of a set of data. The lower quartile (25th percentile) is represented by the lower edge of the box, and the upper quartile (75th percentile) is represented by the upper edge of the box. The median (50th percentile) is represented by a central line that divides the box into sections. The extreme values are represented by whiskers that extend out from the edges of the box.

centrality measure
in graph theory and network analysis, a factor that indicates the relative importance of a vertex within a graph. A few examples of centrality measures are Degree, Closeness, Betweenness, and Eigenvector.

community
a group of nodes in a network that are more densely connected internally than with the rest of the network. A network can contain one or more communities.

data store
a table, view, or file that is registered in a data warehouse environment. Data stores can contain either individual data items or summary data that is derived from the data in a database.

geodesic
the shortest distance between a pair of nodes.

grid
a collection of networked computers that are coordinated to provide load balancing of multiple SAS jobs, accelerated processing of parallel jobs, and scheduling of SAS workflows.

grid host
the machine to which the SAS client makes an initial connection in a SAS High-Performance Analytics application.

link
in a network diagram, a line that represents a relationship between two nodes.

locked-down server
a SAS server that is configured with the LOCKDOWN system option, so that the server can access only designated host resources.

model scoring
the process of applying a model to new data in order to compute outputs.

network
a collection of one or more communities.

node
in a network diagram, a dot or point that represents an individual actor within the network.

outlier
a data point that differs from the general trend of the data by more than is expected by chance alone. An outlier might be an erroneous data point or one that is not from the same sampling model as the rest of the data.

project
the named collection of activities and reports to implement a business strategy for addressing a business pain. For example, a project can be created for reducing churn of highly profitable customers in the North region.

quartile
any of the three points that divide the values of a variable into four groups of equal frequency, or any of those groups. The quartiles correspond to the 25th percentile, the 50th percentile (or median), and the 75th percentile.

scoring
See model scoring.

SMP
See symmetric multiprocessing.

symmetric community
a community in which each central node is symmetrically linked to other nodes of the community.

symmetric multiprocessing
a type of hardware and software architecture that can improve the speed of I/O and processing. An SMP machine has multiple CPUs and a thread-enabled operating system. An SMP machine is usually configured with multiple controllers and with multiple disk drives per controller.

transactional data
timestamped data collected over time at no particular frequency. Some examples of transactional data are point-of-sale data, inventory data, call center data, and trading data.

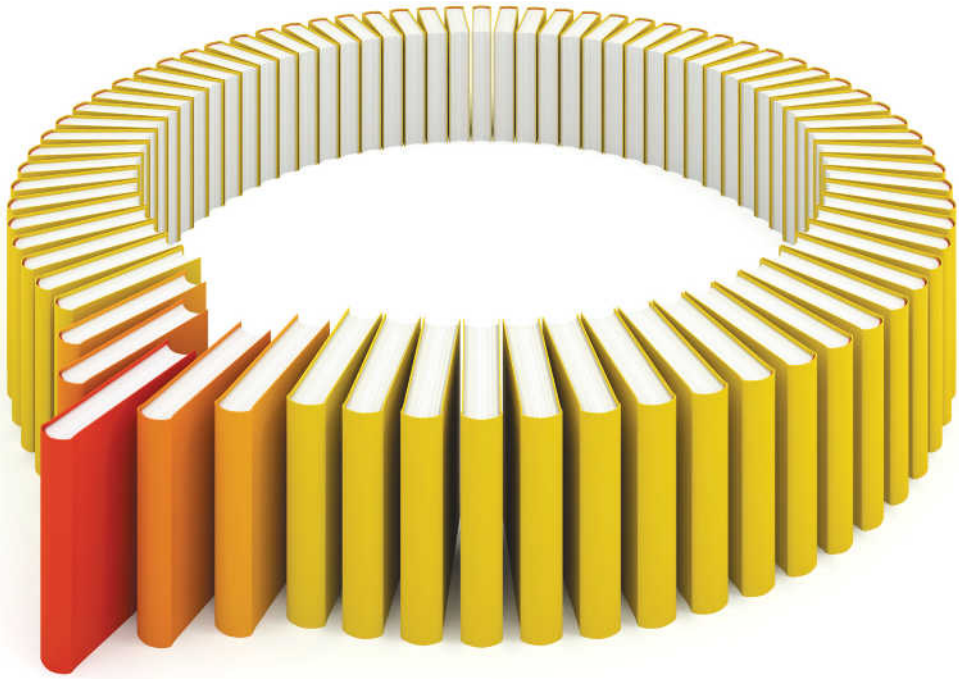
transactional measure
in a transactional table, a type of column that contains an aggregated value. For example, "call duration" is a transactional measure for a communications network, and "number of likes" is a transactional measure for a social network.

whisker
a vertical line on a box plot that represents values larger than the third quartile or smaller than the first quartile but within 1.5 interquartile ranges of the box.

workflow
a series of tasks, together with the participants and the logic that is required to execute the tasks. A workflow includes policies, status values, and data objects.

workflow diagram
a diagram that indicates the order in which activities of a project are to be performed.

workflow step
each individual activity of a project that is depicted in a workflow diagram.



Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore
for additional books and resources.