THE POWER TO KNOW.

SAS® Customer Analytics for Communications 5.3

Administrator's Guide

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2012. *SAS® Customer Analytics for Communications 5.3: Administrator's Guide*. Cary, NC: SAS Institute Inc.

SAS® Customer Analytics for Communications 5.3: Administrator's Guide

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, February 2012

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at **support.sas.com/publishing** or call 1-800-727-3228.

Contents

About This Book . . . . . *v*
Recommended Reading . . . . . *vii*

PART 1 Installing and Configuring SAS Customer Analytics for Communications 1

Chapter 1 • Architecture Overview **3**
SAS Customer Analytics for Communications Architecture 3
Data Flow in SAS Customer Analytics for Communications 7
SAS Customer Analytics for Communications Solution Flow 8

Chapter 2 • Installation and Configuration **11**
Pre-Installation Tasks 11
Default File Locations 13
Installation Instructions 13
Validation Instructions 15
Post-Installation Tasks 17
Unconfiguring SAS Customer Analytics for Communications 22

PART 2 Data Management 25

Chapter 3 • Solution-Specific ETLs **27**
Prerequisite Tasks 27
Populating the Base Tables 27
Configuration Job 32
Writeback ETL Jobs 32

PART 3 Application Management 37

Chapter 4 • Configuring SAS Customer Analytics for Communications **39**
Configuring SAS Customer Analytics for Communications 40
Creating a New Subset Map 43
Creating a Subject Group 49
Outcome-Based Filtering 51
Build-Date Cap 52
Defining Implicit Subset Criterion 52
Creating an Implicit Variable 54
Creating a Post-Action Macro 56
Restarting the Server-Side Web Application 58

Chapter 5 • Performing Middle-Tier Administrative Tasks **59**
Middle-Tier Administration Overview 59

Roles and Capabilities . . . 59
Change the Owner of a Project . . . 62
Working with Software Component Properties . . . 63
Configuring the Logging Folder . . . 66

Chapter 6 • Scoring . . . **67**
The Scoring Process . . . 67

PART 4 Appendixes 75

Appendix 1 • Parameter Configuration . . . **77**

Appendix 2 • Predefined Subset Maps . . . **83**

Appendix 3 • Load Order Sequence . . . **89**

Appendix 4 • Troubleshooting . . . **93**
Troubleshooting a Modeling ABT Building Process . . . 93
Troubleshooting the Scoring Process . . . 95
Troubleshooting the Nonexistence of the APDM Library . . . 96

Glossary . . . **99**
Index . . . **101**

About This Book

Audience

This document is primarily intended for administrators who will perform the initial installation and configuration of SAS Customer Analytics for Communications and also perform the administrative tasks on a regular basis. The document gives an overview of the SAS Customer Analytics for Communications architecture. It also explains various architecture components and the interactions between them. In addition, this document provides the data flow diagram. This diagram explains the data-processing infrastructure of SAS Customer Analytics for Communications. The solution-specific administrative tasks are detailed in the respective chapters.

Prerequisites

Before you administer SAS Offer Optimization for Communications, make sure that you are familiar with the following concepts:

basic concepts and components of the SAS Intelligence Platform
: For details, see *SAS Intelligence Platform: Overview*, which is located at **http://support.sas.com/documentation/cdl/en/biov/63145/PDF/default/biov.pdf**.

the SAS environment
: For details, see *SAS Intelligence Platform: System Administration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/bisag/60945/PDF/default/bisag.pdf**.

the SAS applications servers that are required for particular content
: For details, see *SAS Intelligence Platform: Application Server Administrative Guide*, which is located at **http://support.sas.com/documentation/cdl/en/biasag/61237/PDF/default/biasag.pdf**.

security concepts
: You should be familiar with the authentication and authorization concepts. You should also know how to manage access in the metadata layer. In addition, you should know how to create and manage user and group definitions in metadata. For details, see *SAS Intelligence Platform: Security Administration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/bisecag/61133/PDF/default/bisecag.pdf**.

the middle-tier environment
: You should know how to configure the Application server.

SAS products
: You should know the basic procedures for using the applications that you plan to administer. For example, if you are responsible for administering SAS Web Report Studio, then you should know how to log on, navigate, and create reports in SAS Web Report Studio.

server context
: You should have complete information about the SAS Application Server context. A SAS Application Server knows its server context (the context in which it is being used) and makes decisions based on that knowledge. For example, a client such as SAS Data Integration Studio is assigned a default SAS Application Server. When the client generates code, it submits the code to that application server. The application server determines what type of code is being submitted and directs it to the correct server. That is, if the code is a typical SAS code that can be run in SAS Display Manager, the code is executed by the application server's workspace server. In addition, data-related objects such as SAS libraries, database libraries, and OLAP schemas can be assigned to a SAS application server. After this assignment, a client might need to access data in a particular library or OLAP schema. The client then uses a server component that belongs to the application server to which the library or schema is assigned.

Document Conventions

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

Convention	Description
<SAS Home>	Represents the path to the folder where SAS is installed. For example, on a Windows computer, this 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 can be **C:/SAS/Config**.

Recommended Reading

When you refer to this guide, make sure that you read the following documents in the sequence mentioned below:

- *SAS Communications Analytics Architecture: Data Reference Guide*
- *SAS Communications Analytics Architecture: Administrator's Guide*
- *SAS Customer Analytics for Communications: Data Reference Guide*

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

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-3228
Fax: 1-919-677-8166
E-mail: sasbook@sas.com
Web address: support.sas.com/bookstore

Part 1

Installing and Configuring SAS Customer Analytics for Communications

Chapter 1
Architecture Overview *3*

Chapter 2
Installation and Configuration *11*

Chapter 1

Architecture Overview

SAS Customer Analytics for Communications Architecture **3**
Overview 3
Application Architecture 4
Data Flow in SAS Customer Analytics for Communications **7**
SAS Customer Analytics for Communications Solution Flow **8**

SAS Customer Analytics for Communications Architecture

Overview

The SAS Customer Analytics for Communications architecture is designed to efficiently process a large volume of data. At the same time, the architecture enables the solution to use this data to support user-driven workflow through the application user interface. SAS Customer Analytics for Communications has an n-tier architecture that separates the workflow-related activities from data-intensive process routines and distributes functionality across computer resources that are best suited for these tasks.

You can scale the architecture to meet the demands of your workload. For a large organization, the tiers can be installed across a multitude of 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 Analytics for Communications for small enterprises, then you can install all the tiers on a single machine.

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

Data Tier
: The data tier stores your enterprise data. This data is stored in SAS tables in appropriate libraries and is used in various analysis-through-client programs.

Server Tier
: The server tier of SAS Customer Analytics for Communications consists of data routines and SAS servers that process your enterprise data based on requests from client programs (through middle-tier services) and other programs.

Middle Tier

The middle tier of SAS Customer Analytics for Communications provides an environment in which the SAS Customer Analytics for Communications client, along with other business intelligence Web applications, such as SAS Web Report Studio and SAS Information Delivery Portal, 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 applications depend on the servers that are deployed on the server tier to process, query on, and analyze data.

Client Tier

Clients in SAS Customer Analytics for Communications include Web-based and desktop user interface content and applications. These clients provide access to content, appropriate query and reporting interfaces, and business intelligence functionality, including advanced design and analysis tasks for all information consumers in your enterprise.

Application Architecture

The following diagram describes the components in the SAS Customer Analytics for Communications architecture.

Figure 1.1 *SAS Customer Analytics for Communications Architecture*

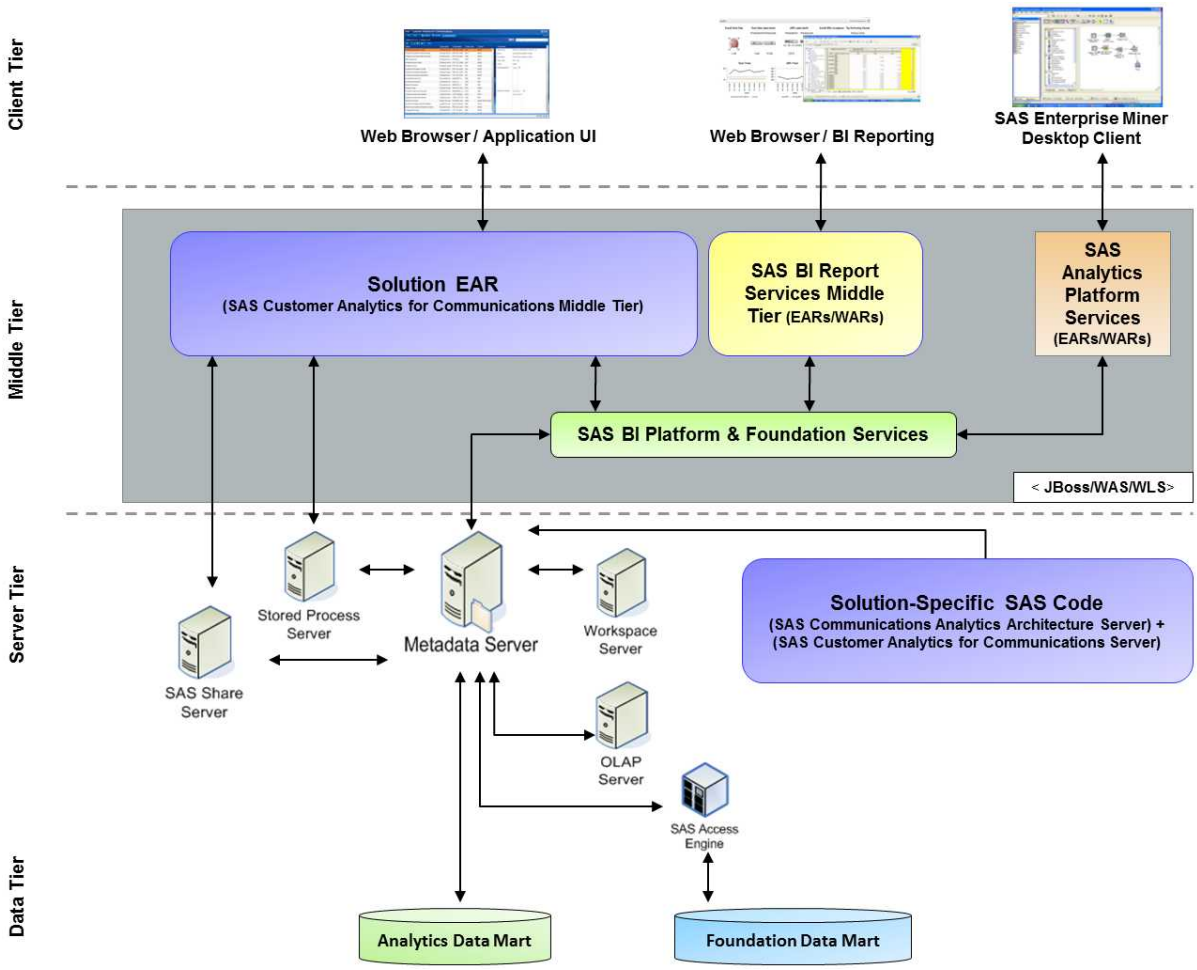


Table 1.1 *Data Tier Components*

Component	Function
Foundation Data Mart	Stores your enterprise data (and history) that is classified into reference data, dimensions, and transaction summary facts. This data is used by the business solutions to support analysis at customer, product, and offer level. This data is also used to generate business intelligence reports and analyze the best offer recommendations. For information about the entities and attributes, see *SAS Communications Analytics Architecture: Data Reference Guide*.
Analytics Data Mart	Consists of base data structures that are used to build analytical models. These models are used for scoring or segmenting customers. The solution has a defined set of derived, behavior, and descriptive variables. These variables are initially used to configure the model and later as an input to the scoring code that is provided by the model. For information about the entities and attributes, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Table 1.2 *Server Tier Components*

Component	Function
Solution-specific SAS code	Represents the data processing SAS routines that are packaged as a part of the solution. These routines perform distinct data operations based on client requests and other SAS routines (in batch mode).
SAS Metadata Server	Is a multi-user-centralized resource for storing, managing, and delivering metadata for all SAS applications across your enterprise.
SAS Workspace Server	Provides access to SAS software features such as SAS language, SAS libraries, the server file system, results content, and formatting services-execution environments for solution data routines.
SAS Stored Process Server	Responds to client requests to execute solution-specific stored processes.
SAS OLAP Server	Provides access to solution-specific cubes.
SAS Share Server	Facilitates concurrent access to SAS data sets from disparate systems.

Table 1.3 *Middle-Tier Components*

Component	Function
SAS BI Platform and Foundation Services	Consists of SAS Shared Services, SAS Remote Services, Java Platform Services, and SAS Web Infrastructure Platform. For details, see *SAS Intelligence Platform: Overview*, which is located at **http://support.sas.com/documentation/cdl/en/biov/63145/PDF/default/biov.pdf**.
SAS BI Report Services Middle Tier	Provides an execution environment for the following business intelligence applications: • SAS Web Report Studio • SAS Information Delivery Portal • SAS BI Dashboard • SAS BI Portlets
SAS Analytics Platform	Provides a common application framework for analytical applications such as SAS Enterprise Miner and SAS Forecast Server. For details, see *Administrator's Guide for SAS Analytics Platform*, which is located at **http://support.sas.com/documentation/onlinedoc/apcore/admin15.pdf**.
SAS Customer Analytics for Communications Middle Tier	Consists of solution-specific services that interact with the client interface to accept user requests (query analysis or data processing) and respond to them with the help of the server tier.

Table 1.4 *Client-Tier Components*

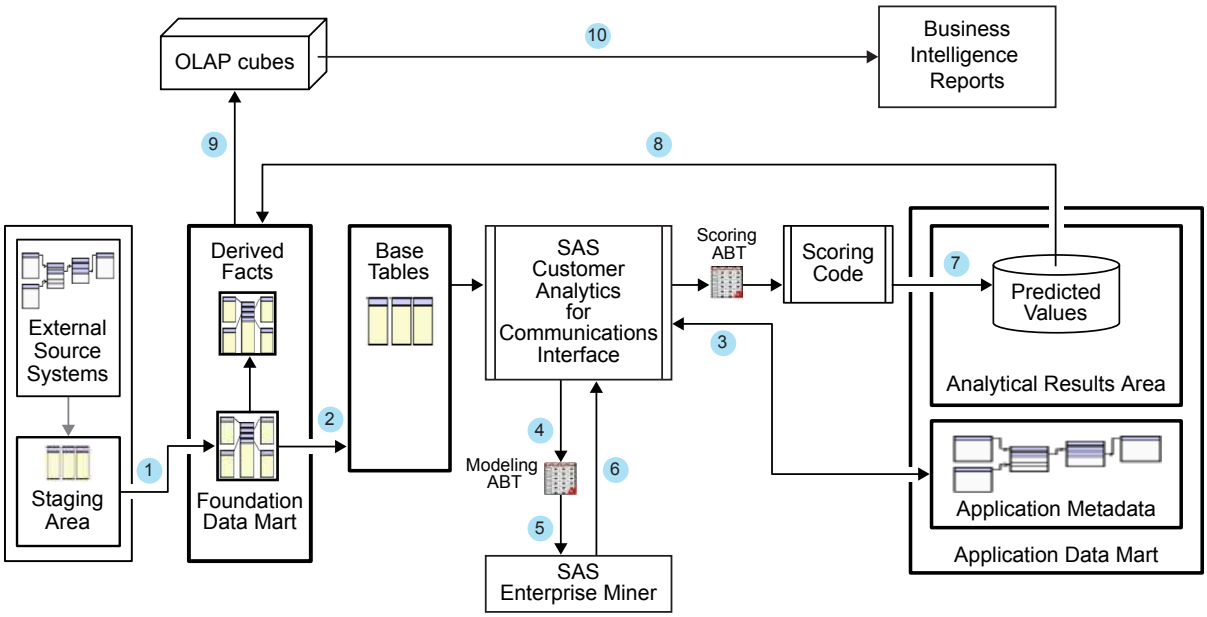
Component	Function
SAS BI Reporting Interface	Consists of the following client interfaces: • SAS OLAP Cube Studio • SAS Information Map Studio • SAS Web Report Studio • SAS Information Delivery Portal • SAS BI Dashboard • SAS BI Portlets For information and usage of these clients, see *SAS Intelligence Platform: Overview*, which is located at **http://support.sas.com/documentation/cdl/en/biov/63145/PDF/default/biov.pdf**.

Component	Function
SAS Enterprise Miner for Desktop and SAS Enterprise Guide	For information and usage of these clients, see *SAS Intelligence Platform: Overview*, which is located at `http://support.sas.com/documentation/cdl/en/biov/63145/PDF/default/biov.pdf`.
SAS Customer Analytics for Communications Web User Interface	Is the Web-based user interface to support the end-to-end analytical workflow from building modeling ABT to scoring. For details, see *SAS Customer Analytics for Communications: User's Guide*.

Data Flow in SAS Customer Analytics for Communications

The following diagram illustrates the data marts that form the solution information model. The diagram also illustrates the data flow across various data marts and the internal working of SAS Customer Analytics for Communications.

Figure 1.2 *Data Flow in SAS Customer Analytics for Communications*



1 Denotes the extract, transform, and load (ETL) jobs that populate the Foundation data mart from the external source systems. These ETL jobs are not pre-packaged with the solution.

2 Denotes the ETL jobs that populate the base tables from the Foundation data mart. These ETL jobs are pre-packaged with the solution. The data in the base tables is used as the source for building modeling and scoring analytical base tables (ABTs). For details, see "Populating the Base Tables" on page 27.

3 SAS Customer Analytics for Communications stores and maintains the application metadata in the Application data mart. The metadata of the various objects, such as source tables, projects, ABTs, variables, and scoring templates, is maintained in various data sets in the Application data mart.

4 SAS Customer Analytics for Communications uses the base tables data to build modeling ABTs. The modeling ABTs are registered with the SAS metadata.

5 SAS Enterprise Miner uses the registered modeling ABTs to create SAS Enterprise Miner models. These models are registered with the SAS metadata.

6 When you capture a model's information in SAS Customer Analytics for Communications, the solution reads the model metadata (such as significant variables, outcome variable, source code, and so on) and stores the information in the Application data mart. When you create a scoring template, the solution reads the scoring code from the model metadata.

7 When you run a scoring job, it uses the base tables data as source and creates a scoring ABT that contains the significant variables. Then, the scoring job applies the SAS Enterprise Miner scoring code on the scoring ABT, and writes the analytical results (for example, the predicted values or segments) to certain tables in the Application data mart. These Application data mart tables are also referred to as the analytical results area.

8 Denotes the ETL jobs that use the information stored in the analytical results area to populate the fact tables and the dimension tables of the Foundation data mart.

9 When you build the OLAP cubes, the OLAP cubes are populated with data from the fact tables and the dimension tables the Foundation data mart.

10 The OLAP cubes are used as source for generating business intelligence reports. You can view these reports through SAS Web Report Studio. For details, see the *SAS Communications Analytics Architecture: User's Guide*.

SAS Customer Analytics for Communications Solution Flow

The SAS Customer Analytics for Communications solution flow contains the following steps:

1. Populate the Foundation data mart with data from the external source system. To populate the Foundation mart with data from the external source system, you need to create ETL jobs. For details, see the *SAS Communications Analytics Architecture: Administrator's Guide*.
2. Populate the base tables with data from the Foundation data mart. You can use the prepackaged ETL jobs to do this. For details, see "Populating the Base Tables" on page 27.
3. Review the default configuration of the application and modify it, if required.

   SAS Customer Analytics for Communications has number of predefined libraries, source tables, time periods for behavioral variables, subset maps, and subject groups. These objects are preconfigured for use in the application. If required, you can modify these objects according to your requirements. You can also define and configure new objects. For details, see Chapter 4, "Configuring SAS Customer Analytics for Communications," on page 39.

4. Log on to SAS Customer Analytics for Communications as a business analyst and complete these tasks:
   a. Create a project for your business problem. For more information, see the *SAS Customer Analytics for Communications: User's Guide*.
   b. Define a subset criterion to define your target population—the population on which you want to perform your analysis. If you want to perform your analysis on the entire population, you can skip this step. For more information, see the *SAS Customer Analytics for Communications: User's Guide*.
   c. Define a modeling analytical base table (ABT). For more information, see the *SAS Customer Analytics for Communications: User's Guide*.
5. Log on to SAS Enterprise Miner, create an analytical model, and register the model with the SAS Metadata Server.
6. Log on to SAS Customer Analytics for Communications as a business analyst, and complete the following tasks. For more information, see the *SAS Customer Analytics for Communications: User's Guide*.
   a. Capture the details of the model that you created in SAS Enterprise Miner and registered with the SAS Metadata Server.
   b. Publish the model for scoring.
   c. Create a scoring template for the published model.
7. Populate the Application data mart by running the scoring job. For details, see "The Scoring Job" on page 68.
8. Populate the Foundation data mart from the Application data mart by using the prepackaged ETL jobs. For details, see "Writeback ETL Jobs" on page 32.
9. Run the OLAP cubes. For more information, see the *SAS Communications Analytics Architecture: Administrator's Guide*.
10. Log on to SAS Web Report Studio and view business intelligence reports. For more information, see the *SAS Communications Analytics Architecture: User's Guide*.

Chapter 2

Installation and Configuration

Pre-Installation Tasks **11**
Verify System Requirements 11
Complete the Pre-Installation Tasks for SAS Intelligence Platform 12
Complete the Pre-Installation Tasks for SAS Communications Analytics Architecture 12
Obtain a Deployment Plan 12
Create a SAS Software Depot 12
Determine the Location of the SAS Environment URL 12

Default File Locations **13**

Installation Instructions **13**

Validation Instructions **15**
Verify the Installation of SAS Customer Analytics for Communications 15
Verify the Metadata Layout 15
Verify Roles and Capabilities 16
Verify the Predefined Libraries 17

Post-Installation Tasks **17**
Overview of Post-Installation Tasks 17
Create Users for SAS Enterprise Miner 18
Verify Operating System Users and Grant Permissions 19
Define Users and Assign Roles 20
Configure Parameters 20
Deploy the Master Loop Job 20
Populate the Reference Tables 21

Unconfiguring SAS Customer Analytics for Communications **22**
Prerequisite Tasks 22
Remove SAS Customer Analytics for Communications 23
Post-Unconfiguration Tasks 23

Pre-Installation Tasks

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 Analytics for Communications. You can access the document from the following location: **http://support.sas.com/resources/sysreq/**.

Complete the Pre-Installation Tasks for SAS Intelligence Platform

Before you begin to install SAS Intelligence Platform and SAS Customer Analytics for Communications, read the *SAS Intelligence Platform: Installation and Configuration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf**. SAS Customer Analytics for Communications is designed to work with SAS Intelligence Platform. Therefore, reading the documentation will help you understand the pre-installation tasks and guide you through a typical installation of SAS Intelligence Platform.

Complete the Pre-Installation Tasks for SAS Communications Analytics Architecture

Before you install SAS Customer Analytics for Communications, make sure that you complete the pre-installation instructions that are detailed in *SAS Communications Analytics Architecture: Administrator's Guide*. For information about how to access SAS Communications Analytics Architecture documentation, see **http://support.sas.com/documentation/onlinedoc/securedoc/index_caa.html**.

Obtain a Deployment Plan

Before you can install SAS Customer Analytics for Communications, 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 the *SAS Intelligence Platform: Installation and Configuration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf**.

Create a SAS Software Depot

Download the software that is listed in your SAS Software Order with the SAS Download Manager. This creates a SAS Software Depot, which includes the SAS installation data (SID) file. The SID file is used by the SAS system 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 the SAS Software Depot, you can then use the SAS Deployment Wizard to install your software. For more information, see "Creating a SAS Software Depot" in the *SAS Intelligence Platform: Installation and Configuration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf**.

Determine the Location of the SAS Environment URL

During deployment, you are prompted by the SAS Deployment Wizard to specify a URL location of the SAS environment file, named sas-environment.xml (for example,

http://<your HTTP server>/sas-environment.xml). This file defines a set of SAS deployments at your site for client applications to use. The sas-environment.xml file does not need to physically exist at the URL location that you specify in the SAS Deployment Wizard before beginning the SAS installation.

However, knowing the intended value of this URL is important because every client installation is prompted for this value. If you do not specify the URL when SAS Customer Analytics for Communications is installed, then as a post-installation task, you must manually edit a file on every client machine to specify this URL. Therefore, it is beneficial to decide on a value for this URL during your planning process so that it can be provided to administrators who might be performing an installation.

For more information about the structure of this file, see "Configuring the SAS Environment File" in the *SAS Intelligence Platform: Middle-Tier Administration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/bimtag/64207/PDF/default/bimtag.pdf**.

Default File Locations

The 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 configuration files are stored in a default location referred to as <SAS configuration directory>. For example, on a Windows machine, <SAS configuration directory> is **C:/SAS/Config**.

Note: You can deploy up to 10 configurations of the SAS products. The SAS Deployment Wizard specifies each configuration under a <Level> folder. For example, if you deploy a level 2 configuration, the default configuration directory is **C:/SAS/Config/Lev2**.

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

Table 2.1 *Default Locations*

Location Name	Windows Path	UNIX Path
<SAS Home>	**C:/Program Files/SASHome**	**../SASHome**
<SAS configuration directory>	**C:/SAS/Config**	**../SAS/Config**

Installation Instructions

SAS Customer Analytics for Communications works with SAS Intelligence Platform, which is also installed and configured when you deploy SAS Customer Analytics for

Communications. You use the SAS Deployment Wizard to install and configure your software. For more information, see the *SAS Deployment Wizard: User's Guide*, which is located at **http://support.sas.com/documentation/installcenter/en/ikdeploywizug/62130/PDF/default/user.pdf**

This section explains installation instructions that are specific to SAS Customer Analytics for Communications. For complete installation and configuration instructions, see the *SAS Intelligence Platform: Installation and Configuration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf**.

If you are installing SAS Customer Analytics for Communications along with SAS Communications Analytics Architecture, then make sure that you refer to the solution-specific installation and configuration instructions that are explained in the *SAS Communications Analytics Architecture: Administrator's Guide*.

To install SAS Customer Analytics for Communications and other relevant SAS products by using the SAS Deployment Wizard, complete these steps:

1. Log on to the machine on which you want to install SAS Customer Analytics for Communications.

   On a Windows machine, log on as a user who is a member of the Administrators group. On a UNIX machine, log on as a SAS user (for example, sas) that you defined when performing the SAS Intelligence Platform pre-installation tasks.

   Note: It is recommended that you do not log on as root to perform an installation on a UNIX machine.

2. Start the SAS Deployment Wizard from your SAS Software Depot. For example, on a Windows machine, double-click the setup.exe file, which is located in your SAS Software Depot folder.

3. Open the *SAS Intelligence Platform: Installation and Configuration Guide*, which is located at **http://support.sas.com/documentation/cdl/en/biig/62611/PDF/default/biig.pdf**.

   TIP On the Select Deployment Step and Products to Install page of the wizard, make sure that you select **SAS Customer Analytics for Communications Server** from the **Product** list.

4. (Optional) If you are installing SAS Customer Analytics for Communications along with SAS Communications Analytics Architecture, see the *SAS Communications Analytics Architecture: Administrator's Guide* for the solution-specific installation instructions.

5. Perform step-by-step installation and configuration as explained in the respective guides.

6. On the Deployment Summary page, review the list of products that you are about to install, and then click **Start**.

7. On the Deployment Complete page, the SAS Deployment Wizard indicates that the installation and configuration of your SAS software is complete. Each software component should have a check mark beside it in the list. Click **Next**.

8. On the Additional Resources page, review the manual configuration instructions in the instructions.html file. This file is saved to your SAS configuration directory during the deployment (for example, **C:/SAS/Config/Lev1/Documents/instructions.html**). You can also view the list of the links for additional resources about your deployment. You can print the list of the links for future reference.

9. Click **Finish** to close the SAS Deployment Wizard.

   Note: On a UNIX platform, certain files require setuid permissions. To set the setuid permissions, run the setuid.sh script, which is located in the **`/usr/local/SASHome/SASFoundation/9.3/utilities/bin`** directory.

Validation Instructions

Verify the Installation of SAS Customer Analytics for Communications

To verify whether SAS Customer Analytics for Communications has been installed properly, complete these steps:

1. Enter the application URL in the address field of your Web browser, and then press ENTER. For example, you might enter **`http://server01.abc.com:8080/SASCustAnalyticsforCommunications/`**.
2. Log on to SAS Customer Analytics for Communications as a user who has all the capabilities of the Business Analysis role. For details, see "Roles and Capabilities" on page 59.
3. You should see the following workspaces:
   - Projects
   - Scoring

   Note: You see the Administrative workspace only if you log on to the application as an administrator.
4. In the Projects workspace, check that you can create a project. After you create a project, you can open it and define its components such as a subset criterion, an analytical base table (ABT), and variables. For instructions on how to create a project and define its components, see *SAS Customer Analytics for Communications: User's Guide*.

   Note: Installing SAS Customer Analytics for Communications does not automatically populate the base tables. Base tables are populated when you run the ETL jobs. For details, see "Populating the Base Tables" on page 27. Therefore, unless you populate the base tables with the required data, you cannot perform the tasks that access the base tables. These tasks include the following:

   - build a modeling ABT
   - view the number of members in the target population defined through a subset criterion (the Show Count operation)

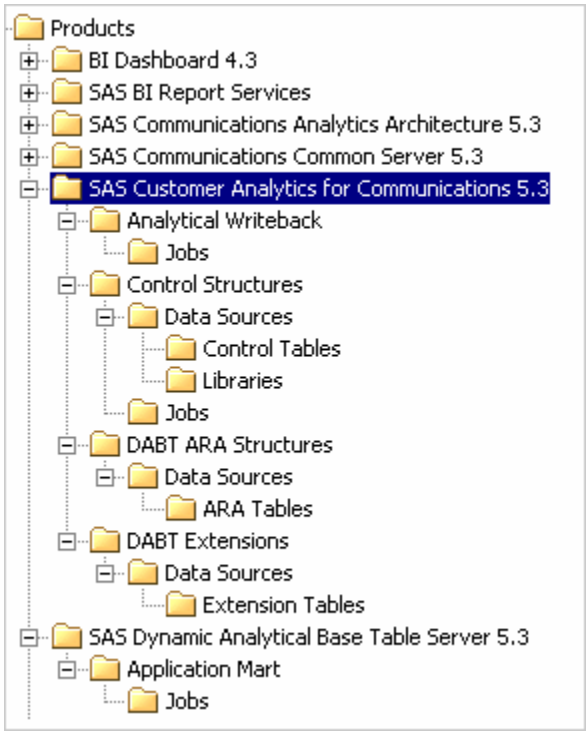
Verify the Metadata Layout

To verify that all the required metadata components are created successfully, complete these steps:

1. Log on to SAS Management Console with a certain profile.

2. On the **Folders** tab, expand **Products** ⇨ **SAS Customer Analytics for Communications 5.3**.
3. Confirm that the following subfolders are created in each of these folders.

Figure 2.1 *Metadata Layout*



4. Close SAS Management Console.

Verify Roles and Capabilities

To verify that appropriate roles and capabilities are created, complete the following steps:

1. Log on to SAS Management Console with the profile of an administrator.
2. On the **Plug-ins** tab, expand **Environment Management** and select **User Manager**.
3. In the right pane, make sure that the following roles are available:
   - Cust Analytics Communications: Administration
   - Cust Analytics Communications: Business Analysis
   - Cust Analytics Communications: Data Analysis
4. Right-click on any one of the roles and select **Properties**.
5. On the **Capabilities** tab, make sure that the role-specific capabilities are selected for the role. For details, see "Roles and Capabilities" on page 59.
6. Close SAS Management Console.

Verify the Predefined Libraries

To confirm that the predefined libraries are created appropriately, complete these steps:

1. Log on to SAS Management Console with the profile of an administrator.
2. On the **Plug-ins** tab, expand **Environment Management** ⇨ **Data Library Manager** ⇨ **Libraries**.
3. Make sure that the following libraries are created:
   - CFDBASE
   - cs_apdm_base
   - cs_apdm_remote
   - dabtctrl
4. Close SAS Management Console.
5. Go to the **`<SAS configuration directory>/Lev1/<SAS Application Server context name>`** folder.
6. Depending on whether the operating system is Windows or UNIX, run the sas.bat or the sas.sh file respectively. For example, on the Windows machine, run the **`C:/SAS/Config93/Lev1/SASApp/sas.bat`** file.
7. In the SAS Explorer, double-click **Libraries** and verify that the following library names are displayed:
   - APDM
   - CFDBASE
   - dabtctrl

   Note: If the APDM library does not appear in the list of libraries, you have to add certain library name statements in the SASApp autoexec file and the SAS Share server autoexec file. For details, see "Troubleshooting the Nonexistence of the APDM Library" on page 96.
8. Close Base SAS.

Post-Installation Tasks

Overview of Post-Installation Tasks

If the installation is successful, the SAS Deployment Wizard produces an HTML file named instructions.html. Follow the post-installation tasks in this file to complete the configuration of the server tier and the middle tier. The instructions.html file also describes when to perform the tasks that are outlined in this chapter.

Additional information about the instructions.html file and supplemental instructions are available in the SAS Intelligence Platform documentation.

Create Users for SAS Enterprise Miner

Overview

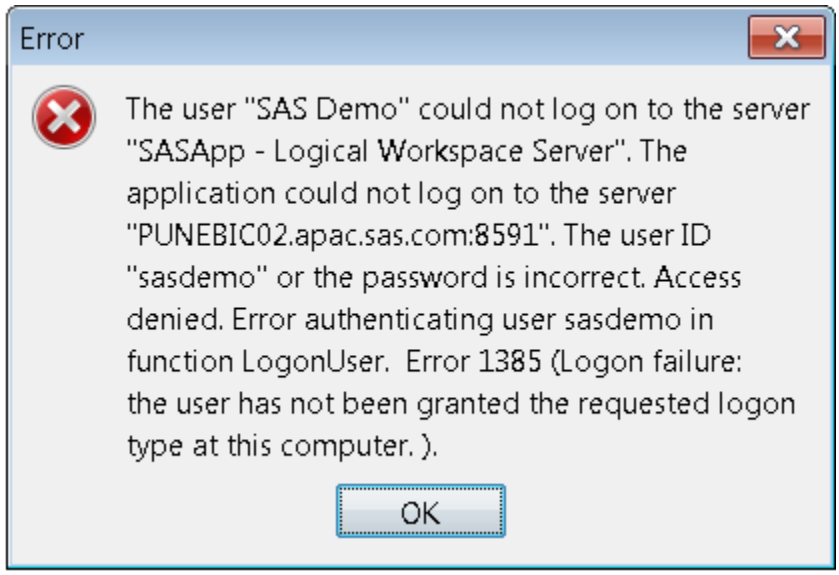
Analytical models for customer churn, customer segmentation, cross-sell and up-sell, and so on, are created using SAS Enterprise Miner. Therefore, you must create a user in the operating system for SAS Enterprise Miner. Also, you have to add this user ID in the SAS metadata by using SAS Management Console and grant privileges that are related to SAS Enterprise Miner.

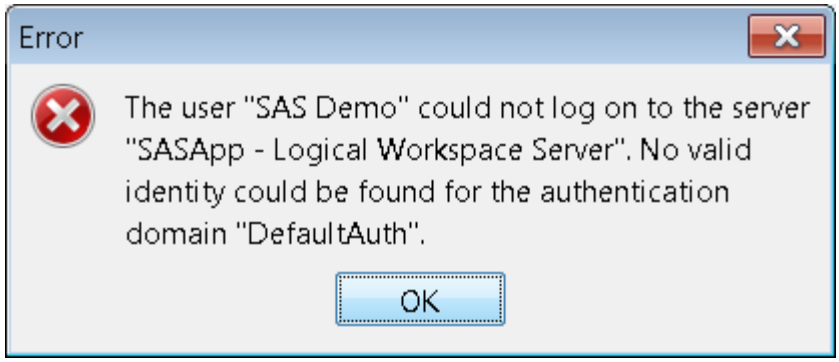
Create Operating System User for SAS Enterprise Miner

On a Windows machine, to create a user and add security policies, perform these steps:

1. Create a user in the operating system.
2. Go to **Control Panel** ⇨ **System and Security** ⇨ **Administrative Tools**.
3. Double-click **Local Security Policy**.
4. In the left pane, expand **Local Policies**, and then select **User Rights Assignment**.
5. Add the user ID that you have created to the following security policies:
   - act as part of the operating system
   - log on as a batch job

   Note: If you do not assign the security policies to the user, you might encounter the following errors when you work in SAS Enterprise Miner:





Create User in SAS Management Console for SAS Enterprise Miner

Note: It is recommended that the user who creates projects in SAS Customer Analytics for Communications is the same as the SAS Enterprise Miner user who creates and registers the analytical models for these projects.

To create a user for SAS Enterprise Miner, complete these steps:

1. Open SAS Management Console with the default profile of sasadm user.
2. On the **Plug-ins** tab, expand **Environment Management**.
3. Right-click **User Manager**, and then select **New** ⇨ **User**. The New User Properties dialog box appears.
4. On the **General** tab, enter the appropriate name and description of the user, and then click **OK**.
5. Select the **Groups and Roles** tab.
   a. Select **Metadata Server: Unrestricted** from the **Available Groups and Roles** list and move it to the **Member of** list.
   b. Click **OK**.
6. Select the **Accounts** tab.
   a. Click **New**. The New Login Properties dialog box appears.
   b. Enter appropriate **User ID** and **Password**.

      Note: Make sure that the user ID that you create is the same as the user ID of the operating system user that you have created. For details, see "Create Operating System User for SAS Enterprise Miner" on page 18.
   c. Select **DefaultAuth** as the **Authentication Domain**.
   d. Click **OK**.
7. Close SAS Management Console.

Verify Operating System Users and Grant Permissions

Verify that a user group named SAS Server Users exists at the operating system level. This user group is created as a part of the SAS installation. Make sure that this group includes the sassrv user (a SAS Server user).

Grant the SAS Server Users group Full Control permission (on Windows) or 775 permission (on UNIX) for the following folders and their subfolders:

- **<SAS configuration directory>/Lev1/AppData/SASCommunicationsCommonSvr**
- **<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsCommServer**
- **<SAS configuration directory>/Lev1/AppData/SASDynamicAnalyBaseTblSvr**
- **<SAS configuration directory>/Lev1/Applications/SASCommunicationsCommonSvr5.3**
- **<SAS configuration directory>/Lev1/Applications/SASCustAnalyticsCommServer5.3**
- **<SAS configuration directory>/Lev1/Applications/SASDynamicAnalyBaseTblSvr5.3**

Define Users and Assign Roles

You need to define users who can log on to SAS Customer Analytics for Communications and perform tasks based on their assigned roles. SAS Customer Analytics for Communications ships with three predefined roles: Business Analysis, Data Analysis, and Administration. A predefined set of capabilities is available for each role.

You can define users and assign roles in SAS Management Console. For more information, see the *SAS 9.3 Management Console: Guide to Users and Permissions*. This document is available at **http://support.sas.com/documentation/cdl/en/mcsecug/63190/PDF/default/mcsecug.pdf**.

Configure Parameters

SAS Customer Analytics for Communications has a set of parameters that are defined in the DABT_CONFIG_PARAM_TBL table (in the APDM library). These parameters are essential to ensure proper functioning of the solution. During the installation, these parameters are assigned certain default values. Review these default values and modify them, as required. For details, see "Parameters List" on page 77.

Deploy the Master Loop Job

The MasterLoopDABTJob invokes the jobs that populate the ABT during the ABT building process.

To deploy the MasterLoopDABTJob job, complete these steps:

1. Log on to SAS Data Integration Studio and connect to an appropriate profile.
2. On the **Folders** tab, expand **Products** ⇨ **SAS Dynamic Analytical Base Table Server 5.3** ⇨ **Application Mart** ⇨ **Jobs**.
3. Right-click **MasterLoopDABTJob**, and then select **Scheduling** ⇨ **Deploy**.
4. In the Deploy a job for scheduling dialog box, from the **Batch Server** list, select **<Application server> - SAS DATA Step Batch Server**.

   Note: **<Application server>** represents the application server on which you installed SAS Customer Analytics for Communications. For example, if you

installed SAS Customer Analytics for Communications on SASApp, select **SASApp – SAS DATA Step Batch Server**.

5. Click **New** to define the deployment directory. The New directory dialog box appears.
   a. In the **Name** field, type a name for the deployment directory. For example, you can type **`dabt_deployedjobs`**.
   b. In the **Directory** field, type **`<SAS configuration directory>/Lev1/Applications/SASDynamicAnalyBaseTblSvr5.3/deployed_jobs/dabt`**. Alternatively, you can click **Browse** to navigate to the deployment directory.
   c. Click **OK**.
6. Do not change the default values in the **Deployed Job Name** field and the **Location** field.
7. Click **OK**.

   If the MasterLoopDABTJob job is deployed successfully, a file named MasterLoopDABTJob.sas is created in the deployment directory (**`<SAS configuration directory>/Lev1/Applications/SASDynamicAnalyBaseTblSvr5.3/deployed_jobs/dabt`**).
8. Edit the MasterLoopDABTJob.sas file and make the following changes:

   Note: You must edit the MasterLoopDABTJob.sas file and make these changes every time you deploy or redeploy the MasterLoopDABTJob job.
   a. Search for the line of code beginning with the following declaration: **`%let etls_controlName`**

      You will find two occurrences of this line of code.
   b. On the first occurrence, replace the value of the etls_controlName variable with the following: **`&outer_loop_remote_session_prefix.;`**
   c. On the second occurrence, replace the value of the etls_controlName variable with the following: **`&inner_loop_remote_session_prefix.;`**
9. Save the changes, and then close the MasterLoopDABTJob.sas file.
10. Close SAS Data Integration Studio.

Populate the Reference Tables

The value of the LEVEL_CD column of the LEVEL_MASTER table (in the APDM library) must be same as the value of the ANALYTICAL_MODEL_LEVEL_CD column of the ANALYTICAL_MODEL_LEVEL table (in the CFDREF library). Also, the value of the ANALYTICAL_MODEL_TYPE_CD column of the ANALYTICAL_MODEL_TYPE table (in the CFDREF library) must be same as the first two letters of the PURPOSE_CD column of the PURPOSE_MASTER table (in the APDM library).

To populate these tables with appropriate data based on these constraints, complete these steps:

1. Make sure that the ANALYTICAL_MODEL_LEVEL and ANALYTICAL_MODEL_TYPE tables do not contain any data.

2. Go to the **<SAS configuration directory>/Lev1/<SAS Application Server context name>** folder.
3. Depending on whether the operating system is Windows or UNIX, run the sas.bat or the sas.sh file respectively. For example, on the Windows machine, run the **C:/SAS/Config93/Lev1/SASApp/sas.bat** file.
4. View the pre-assigned libraries such as CFDREF, CFDDIM, and so on, in the SAS Explorer.
5. Open the cac_post_install_wrapper.sas macro. This macro is located in the following folder:

   Windows
   <SAS Home>/SASFoundation/9.3/cacsrv/sasmisc/controlscripts.

   UNIX
   <SAS Home>/SASFoundation/9.3/misc/cacsrv/controlscripts
6. Verify the arguments.

   ```
   %cac_post_install_wrapper(ref=CFDREF);
   ```
7. Enter appropriate values for the macro arguments. Ensure that the correct library reference for the reference library is provided as an argument to this macro.
8. Click **Save**.
9. Click **Run**.
10. To ensure successful execution of the INSERT scripts, view the log.
11. Close Base SAS.

Unconfiguring SAS Customer Analytics for Communications

Prerequisite Tasks

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

1. Create a backup of the following folders:
   - **<SAS configuration directory>/Lev1/AppData/SASCommunicationsCommonSvr**
   - **<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsCommServer**
   - **<SAS configuration directory>/Lev1/AppData/SASDynamicAnalyBaseTblSvr**
2. Create a backup of the application metadata if you have made any customizations that you want to save for later use. To do so, Start SAS Management Console, and then open the appropriate connection profile to connect to the desired metadata server.

3. On the **Folders** tab, expand **SAS Folders** ⇨ **Products**.
4. Create a backup of the following folders:
   - **SAS Communications Common Server 5.3**
   - **SAS Customer Analytics for Communications 5.3**
   - **SAS Dynamic Analytical Base Table Server 5.3**
   - **User Folders**
5. Close SAS Management Console.
6. Stop the Web application server (JBoss, WebSphere, or WebLogic, as applicable).

Remove SAS Customer Analytics for Communications

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

- SAS Customer Analytics for Communications Mid-Tier
- SAS Customer Analytics for Communications Server
- SAS Communications Common Server

Post-Unconfiguration Tasks

After you have removed the software components of SAS Customer Analytics for Communications, 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, expand **SAS Folders** ⇨ **Products**.
3. Delete the following folders:
   - **SAS Communications Common Server 5.3**
   - **SAS Customer Analytics for Communications 5.3**
   - **SAS Dynamic Analytical Base Table Server 5.3**
4. Close SAS Management Console.
5. Delete the following folders:
   - **`<SAS configuration directory>/Lev1/AppData/SASCommunicationsCommonSvr`**
   - **`<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsCommServer`**
   - **`<SAS configuration directory>/Lev1/AppData/SASDynamicAnalyBaseTblSvr`**

Part 2

Data Management

Chapter 3
Solution-Specific ETLs *27*

Chapter 3

Solution-Specific ETLs

Prerequisite Tasks 27
Populating the Base Tables 27
Configuration Job 32
Writeback ETL Jobs 32

Prerequisite Tasks

Before you proceed with performing the tasks that are mentioned in this chapter, make sure that the following tasks are complete:

1. Perform all the tasks that are detailed in the *SAS Communications Analytics Architecture: Administrator's Guide*. Especially, make sure that you have run all the extract, transform, and load (ETL) jobs. These are prerequisite jobs for running the ETL jobs that are required for SAS Customer Analytics for Communications.
2. Perform all the post-installation tasks that are detailed in this guide. For details, see "Post-Installation Tasks" on page 17.

Populating the Base Tables

In order to populate the base tables, you have to run the ETL jobs for analytics. These jobs extract data from the Foundation data mart and then load this data in the base tables.

The following table lists the jobs that you have to run for managing data that is required for the analytical components. For the recommended sequence in which you should run these jobs, see "Load Order Sequence for Bill-Monthly Jobs" on page 89 and "Load Order Sequence for Weekly Jobs" on page 90.

Table 3.1 *ETL Jobs for Analytics — Prepaid*

Job Name	Purpose	Primary Source Table	Target Table
cfd_pre_pd_cust_interaction_b_job	Loads interaction-related data (that is, count of inquiries and complaints for prepaid customers). This job loads data for the completed week.	• CUST_INQUIRY_F • CUST_COMPLAINT_F	PRE_PD_CUST_INTERACTION_B
cfd_pre_pd_cust_loyalty_b_job	Loads loyalty-related data such as loyalty points earned and loyalty bonus points earned for prepaid customers. This job loads data for the completed week.	LOYALTY_ACTIVITY_F	PRE_PD_CUST_LOYALTY_B
cfd_pre_pd_equip_activity_b_job	Loads information about status and their corresponding change reason for various equipment that is owned by prepaid customers. This job loads data for the completed week.	SUBSCRP_EQUIP_ACTIVITY_F	PRE_PD_EQUIP_ACTIVITY_B
cfd_pre_pd_service_activity_b_job	Loads information about various services and their status that are owned by prepaid customers. This job loads data for the completed week.	SUBSCRP_SERVICE_ACTIVITY_F	PRE_PD_SERVICE_ACTIVITY_B
cfd_pre_pd_subscrp_bucket_drvd_b_job	Loads various bucket-level customer variables that are precalculated for analytical processing such as calculating the duration between first recharge from activation. This job loads data for the completed week.	• PREPAID_NON_USAGE_CHARGE_F • PREPAID_USAGE_CHARGE_F • SUBSCRP_BUCKET_USAGE_SNPSHT_F	PRE_PD_SUBSCRP_BUCKET_DRVD_B
cfd_pre_pd_subscrp_bucket_usage_b_job	Loads a daily snapshot of the recharge closing balance and usage decrement value for prepaid customers. This job loads data for the completed week.	SUBSCRP_BUCKET_USAGE_SNPSHT_F	PRE_PD_SUBSCRP_BUCKET_USAGE_B
cfd_pre_pd_subscrp_interaction_b_job	Loads interaction information at subscription level for prepaid customers. This job loads data for the completed week.	• CUST_INQUIRY_F • CUST_COMPLAINT_F	PRE_PD_SUBSCRP_INTERACTION_B

Job Name	Purpose	Primary Source Table	Target Table
cfd_pre_pd_subscrp_loyalty_b_job	Loads information about loyalty points for prepaid customers at a subscription level. This job loads data for the completed week.	LOYALTY_ACTIVITY_F	PRE_PD_SUBSCRP_LOYALTY_B
cfd_pre_pd_subscrp_usage_drvd_b_job	Loads various precalculated variables at subscription level. These variables are required for analytical processing such as calculating duration of last recharge from first usage. This job loads data for the completed week.	• PREPAID_NON_USAGE_CHARGE_F • PREPAID_USAGE_CHARGE_F	• PRE_PD_SUBSCRP_USAGE_DRVD_B • USAGE_SUMMARY_F • PAYMENT_F
cfd_pre_pd_usage_b_job	Loads usage information for prepaid customers. This job loads data for the completed week.	• USAGE_SUMMARY_F • EVENT_FAILURE_SUMMARY_F	PRE_PD_USAGE_B
cfd_pre_pd_usage_recharge_b_job	Loads recharge information for prepaid customers. This job loads data for the completed week.	• PAYMENT_F • PREPAID_USAGE_CHARGE_F	PRE_PD_USAGE_RECHARGE_B
cfd_pre_pd_cust_offer_snpsht_b_job	Loads the latest details of the all the offers that a prepaid customer owns.	• PRE_PD_CUST_TMP • OFFER_D	PRE_PD_CUST_OFFER_SNPSHT_B
cfd_pre_pd_cust_snpsht_b_job	Loads the details about prepaid customers. These details are related to current and previous offer bundle and subscription status (count of activation and deactivation).	• PRE_PD_CUST_TMP • OFFER_BUNDLE_D,SUBSCRP_D	PRE_PD_CUST_SNPSHT_B
cfd_pre_pd_usage_summary_b_job	Loads summarized usage-related information for prepaid customers. This job loads data for a completed week.	• USAGE_SUMMARY_F • EVENT_FAILURE_SUMMARY_F	PRE_PD_USAGE_SUMMARY_B

Table 3.2 *ETL Jobs for Analytics — Postpaid*

Job Name	Purpose	Primary Source Table	Target Table
cfd_pst_pd_bill_usage_b_job	Loads information about billing-usage data for the postpaid customers.	BILL_USAGE_F	PST_PD_BILL_USAGE_B

Job Name	Purpose	Primary Source Table	Target Table
cfd_pst_pd_cust_acct_snpsht_b_job	Loads derived information about customers' accounts for postpaid customers. This job loads data for the completed billing cycle.	CUST_ACCT_BALANCE_SNPSHT_F	PST_PD_CUST_ACCT_SNPSHT_B
cfd_pst_pd_cust_bill_b_job	Loads bill-related information for postpaid customers. This job loads data for the completed billing cycle.	BILL_F	PST_PD_CUST_BILL_B
cfd_pst_pd_cust_bill_nonusage_b_job	Loads bill-non-usage-related information for postpaid customers. This job loads data for the completed billing cycle.	BILL_NON_USAGE_F	PST_PD_CUST_BILL_NONUSAGE_B
cfd_pst_pd_cust_interaction_b_job	Loads interaction-related information for postpaid customers. This job loads data for the completed billing cycle.	• CUST_INQUIRY_F • CUST_COMPLAINT_F	PST_PD_CUST_INTERACTION_B
cfd_pst_pd_cust_loyalty_b_job	Loads information about loyalty points for postpaid customers. This job loads data for the completed billing cycle.	LOYALTY_ACTIVITY_F	PST_PD_CUST_LOYALTY_B
cfd_pst_pd_equip_activity_b_job	Loads information related to the status and their corresponding change reason for various equipment that is owned by postpaid customers. This job loads data for the completed billing cycle.	SUBSCRP_EQUIP_ACTIVITY_F	PST_PD_EQUIP_ACTIVITY_B
cfd_pst_pd_payment_base_b_job	Loads payment-related information about postpaid customers. This job loads data for the completed billing cycle.	PAYMENT_F	PST_PD_PAYMENT_B
cfd_pst_pd_payment_drvd_b_job	Loads payment-related-derived information about postpaid customers. This job loads data for the completed billing cycle.	• BILL_X_PAYMENT_BRIDGE • BILL_F	PST_PD_PAYMENT_DRVD_B

Job Name	Purpose	Primary Source Table	Target Table
cfd_pst_pd_service_activity_b_job	Loads information about various services and their statuses that are owned by postpaid customers. This job loads data for the completed billing cycle.	SUBSCRP_SERVICE_ACTIVITY_F	PST_PD_SERVICE_ACTIVITY_B
cfd_pst_pd_subscrp_bill_nonusage_b_job	Loads bill-non-usage-related information about postpaid customers at subscription level. This job loads data for the completed billing cycle.	BILL_NON_USAGE_F	PST_PD_SUBSCRP_BILL_NONUSAGE_B
cfd_pst_pd_subscrp_interaction_b_job	Loads subscription-level interaction information about postpaid customers. This job loads data for the completed billing cycle.	• CUST_INQUIRY_F • CUST_COMPLAINT_F	PST_PD_SUBSCRP_INTERACTION_B
cfd_pst_pd_subscrp_loyalty_b_job	Loads the loyalty points for subscriptions of postpaid customers. This job loads data for the completed billing cycle.	LOYALTY_ACTIVITY_F	PST_PD_SUBSCRP_LOYALTY_B
cfd_pst_pd_usage_b_job	Loads usage-related information about postpaid customers. This job loads data for the completed billing cycle.	• USAGE_SUMMARY_F • EVENT_FAILURE_SUMMARY_F	PST_PD_USAGE_B
cfd_pst_pd_cust_offer_snpsht_b_job	Loads the latest details of the all the offers that a postpaid customer owns.	• CUST_X_BILL_CYCLE_BRIDGE_TMP • OFFER_D	PST_PD_CUST_OFFER_SNPSHT_B
cfd_pst_pd_cust_snpsht_b_job	Loads the details about postpaid customers. These details are related to current and previous offer bundle and subscription status (count of activation and deactivation).	• CUST_X_BILL_CYCLE_BRIDGE_TMP • OFFER_BUNDLE_D • SUBSCRP_D	PST_PD_CUST_SNPSHT_B
cfd_pst_pd_usage_summary_b_job	Loads summarized usage-related information about postpaid customers. This job loads data for a completed billing cycle.	• USAGE_SUMMARY_F • EVENT_FAILURE_SUMMARY_F	PST_PD_USAGE_SUMMARY_B

Configuration Job

You have to run the cfd_input_mart_load_dates_job configuration job in SAS Data Integration Studio after you run the ETLs that populate the base tables.

Table 3.3 *Configuration Job Details*

Job Name	Purpose	Primary Source Table	Target Table
cfd_input_mart_load_dates_job	Populates the latest load date from the SETUP_PARAM table into the INPUT_MART_LOAD_DATES table. The SETUP_PARAM table contains the latest date for which the data is loaded in the base tables. This date is populated into the INPUT_MART_LOAD_DATES table by using this job to synchronize the date in the base tables with the date to be chosen for the batch run of the scoring templates. The INPUT_MART_LOAD_DATES table is updated every time the input mart is loaded.	SETUP_PARAM	INPUT_MART_LOAD_DATE

Writeback ETL Jobs

The writeback ETL jobs write back the analytical scores to the Foundation data mart tables. You have to schedule these jobs for execution after you complete the scoring-related tasks. For details about the scoring tasks, see *SAS Customer Analytics for Communications: User's Guide* and Chapter 6, "Scoring," on page 67 of this guide. For the recommended sequence in which you should run these jobs, see "Load Order Sequence for Writeback Jobs" on page 91.

Table 3.4 *Writeback Jobs*

Job Name	Purpose	Primary Source Table	Target Table
cac_analytical_rule_master_dtl_job	Loads records that contain rules of MBA (market basket analysis) or sequence analysis models. A single record contains information about one rule. The target table contains one record for a combination of ANALYTICAL_MODEL_SK and RULE_SK.	SCORING_MODEL_RULE_MASTER	ANALYTICAL_RULE_MASTER_DTL
cac_analytical_rule_dtl_job	Loads transformed records (that is, separate records for each entity in a specific rule) of rules of MBA or sequence analysis models. The target table contains one record for a combination of ANALYTICAL_MODEL_SK, RULE_SK, and RULE_UNIT_TYPE_CD per UNIT_ID.	SCORING_MODEL_RULE_DTLS	ANALYTICAL_RULE_DTL
cac_subscrp_rule_score_dtl_job	Loads scores of rules that are associated with models such as MBA and sequence analysis at subscription level. The SUBSCRP_RULE_SCORE_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK, SUBSCRP_MODEL_SCORE_DT, SUBSCRP_SK, and RULE_SK. The PREDICTIVE_MODEL_RUN_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK and MODEL_SCORE_DT.	RULES_SCORING_RESULT_SUB	• SUBSCRP_RULE_SCORE_DTL • PREDICTIVE_MODEL_RUN_DTL

Job Name	Purpose	Primary Source Table	Target Table
cac_cust_rule_score_dtl_job	Loads scores of rules that are associated with models such as MBA and sequence analysis at customer level. The CUST_RULE_SCORE_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK, CUST_MODEL_SCORE_DT, CUST_SK, and RULE_SK. The PREDICTIVE_MODEL_RUN_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK and MODEL_SCORE_DT.	RULES_SCORING_RESULT_CUST	• CUST_RULE_SCORE_DTL • PREDICTIVE_MODEL_RUN_DTL
cac_subscrp_model_score_dtl_job	Loads model scores at subscription level. SUBSCRP_MODEL_SCORE_DTL would contain one record per SUBSCRP_SK per ANALYTICAL_MODEL_SK per SUBSCRP_MODEL_SCORE_DT. PREDICTIVE_MODEL_RUN_DTL would contain one record per ANALYTICAL_MODEL_SK per MODEL_SCORE_DT.	PREDICTED_SCORING_RESULT_SUB	• SUBSCRP_MODEL_SCORE_DTL • PREDICTIVE_MODEL_RUN_DTL

Job Name	Purpose	Primary Source Table	Target Table
cac_cust_model_score_dtl_job	Loads model scores at customer level. Also loads the aggregated scores (aggregated from subscription level records to customer level) to customer level tables. The CUST_MODEL_SCORE_DTL table contains one record for a combination of CUST_SK, ANALYTICAL_MODEL_SK, and CUST_MODEL_SCORE_DT. The PREDICTIVE_MODEL_RUN_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK and MODEL_SCORE_DT.	PREDICTED_SCORING_RESULT_CUST	• CUST_MODEL_SCORE_DTL • PREDICTIVE_MODEL_RUN_DTL
cac_prspct_cust_model_score_dtl_job	Loads model scores at prospective customer level. The PRSPCT_CUST_MODEL_SCORE_DTL table contains one record for a combination of PRSPCT_CUST_SK, ANALYTICAL_MODEL_SK, and PRSPCT_CUST_MODEL_SCORE_DT. The PREDICTIVE_MODEL_RUN_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK and MODEL_SCORE_DT.	PREDICTED_SCORING_RESULT_PRS	• PRSPCT_CUST_MODEL_SCORE_DTL • PREDICTIVE_MODEL_RUN_DTL

Job Name	Purpose	Primary Source Table	Target Table
cac_cust_analytical_sgmt_dtl_job	Loads analytical segment scores at customer level. The CUST_ANALYTICAL_SGMT_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK, CUST_MODEL_SCORE_DT, and CUST_SK. The SEGMENTATION_MODEL_RUN_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK and SEGMENT_MODEL_SCORE_DT.	SEGMENT_SCORING_RESULT_CUST	• CUST_ANALYTICAL_SGMT_DTL • SEGMENTATION_MODEL_RUN_DTL
cac_model_writeback_job	Loads all model details from the writeback tables to the Foundation data mart (FDM) tables. These details include model details such as model type code and model ID. This information also includes other attributes such as services or offers for cross-sell and up-sell and segment details for segmentation models. The ANALYTICAL_MODEL_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK, ENTITY_LEVEL_CD, and LEVEL_ENTITY_SK. The ANALYTICAL_SGMT_DTL table contains one record for a combination of ANALYTICAL_MODEL_SK and ANALYTICAL_SEGMENT_CD.	• SCORING_MODEL • SCORING_MODEL_RULE_MASTER • SCORING_MODEL_SEGMENT_MASTER • SERVICE_D • OFFER_D	• ANALYTICAL_MODEL_DTL • ANALYTICAL_SGMT_DTL • ANALYTICAL_MODEL_D

Part 3

Application Management

Chapter 4
Configuring SAS Customer Analytics for Communications *39*

Chapter 5
Performing Middle-Tier Administrative Tasks *59*

Chapter 6
Scoring *67*

Chapter 4

Configuring SAS Customer Analytics for Communications

Configuring SAS Customer Analytics for Communications **40**
Overview 40
Configuring Libraries 40
Configuring Time Periods for Behavioral Variables 41
Configuring Source Tables 42
Configuring the Columns of a Source Table 43
Configure Dimensional Attribute Values for a Column 43

Creating a New Subset Map **43**
Overview 43
Define the Subset Map 44
Define Relationships among Subset Map Tables 44
Configure the Availability of the Subset Map 45
Example: Creating a Subset Map 45

Creating a Subject Group **49**
Overview 49
Create a Subset Criterion 50
Define a Subject Group 50
Configure the Availability of the Subject Group 51

Outcome-Based Filtering **51**

Build-Date Cap **52**

Defining Implicit Subset Criterion **52**
Overview 52
Create an Implicit Subset Criterion 53
Configure the Availability of the Implicit Subset Criterion 53

Creating an Implicit Variable **54**
Overview 54
Create an Implicit ABT Variable 55
Configure the Availability of the Implicit Variable 55

Creating a Post-Action Macro **56**
Overview 56
Create a Post-Action Macro 57
Configure the Availability of the Post-Action Macro 58

Restarting the Server-Side Web Application **58**

Configuring SAS Customer Analytics for Communications

Overview

SAS Customer Analytics for Communications is prepackaged with a number of libraries, source tables, time periods for behavioral variables, subset maps, and subject groups. These objects are preconfigured for use in the application. However, if required, you can modify these objects according to your requirements. You can also define and configure new libraries, source tables, time periods, implicit subset criteria, implicit variables, subset maps, and subject groups.

Note:

- It is recommended that you make sure that no user is logged on to the application when you modify any of the predefined objects or define new objects. Also, the changes that you make are reflected in the application interface only on the next logon.
- Also, when you configure these objects, make sure that you restart the server-side Web application. For details, see "Restarting the Server-Side Web Application" on page 58.

Configuring Libraries

SAS Customer Analytics for Communications provides a number of libraries that store the source tables that are prepackaged (and preconfigured for use). The LIBRARY_MASTER table (in the APDM library) stores the records of these libraries.

The following table shows the LIBRARY_MASTER table with sample records.

For descriptions of the columns in the LIBRARY_MASTER table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

If you want to add new source tables for use in the application, you can add the tables to any of these preconfigured libraries. However, if required, you can also define a new library for your new source tables. To define a new library, you need to add a record for that library in the LIBRARY_MASTER table.

Table 4.1 *LIBRARY_MASTER Table*

LIBRARY_SK	LIBRARY_ REFERENCE	LIBNAME_ STATEMENT	LIBRARY_ TYPE_CD	LIBRARY_ SHORT_NM	LIBRARY_ DESC
1	APDM	libname apdm "C:\SAS\Config\Lev1\AppData\SASCommunicationsCommonSvr\5.3\data\apdm";	SOU	DABT APDM Library	DABT APDM Library

LIBRARY_SK	LIBRARY_ REFERENCE	LIBNAME_ STATEMENT	LIBRARY_ TYPE_CD	LIBRARY_ SHORT_NM	LIBRARY_ DESC
3	inpbase	libname inpbase oracle user=BASE password=BASE path=OOC;	SOU	BASE	BASE
5	FACT	libname FACT oracle user=FACT password=FACT path=OOC;	SOU	FACT	FACT
6	DIM	libname dim "C:\SAS\Config\Lev1\AppData\SASCommAnalyticsArchSvr\5.3\data\dim";	SOU	DIM	DIM

Note: Replace the path in the LIBNAME_STATEMENT column according to the <SAS configuration directory> that is configured on your machine.

After you have added the record in the LIBRARY_MASTER table, the library name appears in the **Library** list in the Import Table window (in the Administrative workspace of the SAS Customer Analytics for Communications interface).

You can change the short name and description of a preconfigured library. You can also delete a library from the LIBRARY_MASTER table if the library is no longer used in the application—that is, there are no configured source tables that refer to that library.

Configuring Time Periods for Behavioral Variables

Before you can create behavioral variables in SAS Customer Analytics for Communications, you must define the time periods based on which the behavioral variables will be created. These time periods determine the data aggregations that are possible when defining behavioral variables. For more information about behavioral variables, see the *SAS Customer Analytics for Communications: User's Guide*.

SAS Customer Analytics for Communications provides a number of predefined time periods. The SOURCE_TIME_PERIOD table (in the APDM library) stores the definitions of these time periods.

The following table shows the SOURCE_TIME_PERIOD table with sample time period definitions:

Table 4.2 *SOURCE_TIME_PERIOD Table*

TIME_ PERIOD_SK	TIME_ FREQUENCY_ SK	TIME_ PERIOD_ FROM	TIME_ PERIOD_TO	TIME_ PERIOD_CD	TIME_ PERIOD_ SHORT_NM	TIME_ PERIOD_ DESC
1	1	1	1	B1M	Base 1 Month	Base 1 Month
2	1	2	2	B2M	Base 2 Month	Base 2 Month

TIME_ PERIOD_SK	TIME_ FREQUENCY_ SK	TIME_ PERIOD_ FROM	TIME_ PERIOD_TO	TIME_ PERIOD_CD	TIME_ PERIOD_ SHORT_NM	TIME_ PERIOD_ DESC
3	1	3	3	B3M	Base 3 Month	Base 3 Month
4	1	4	4	B4M	Base 4 Month	Base 4 Month
5	1	5	5	B5M	Base 5 Month	Base 5 Month
6	1	6	6	B6M	Base 6 Month	Base 6 Month

For descriptions of the columns in the SOURCE_TIME_PERIOD table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

You can use these predefined time periods to create behavioral variables. However, if required, you can also define new time periods. To define a new time period, you need to add a record for that time period in the SOURCE_TIME_PERIOD table.

You can change the short name and description of a predefined time period. You can also delete a time period from the SOURCE_TIME_PERIOD table if the time period is no longer used in the application—that is, there are no behavioral variables that are based on that time period.

Configuring Source Tables

In SAS Customer Analytics for Communications, source tables provide source data for the subset criterion and analytical base table (ABT) variables that you define for a project.

SAS Customer Analytics for Communications provides a number of preconfigured source tables. When you log on to SAS Customer Analytics for Communications, the Administrative workspace shows a list of these preconfigured tables. You can use these tables to create subset maps and ABT variables in the application. However, if required, you can modify the default configuration according to your requirement.

Note: The prepackaged source tables and their columns are preconfigured for use. You do not need to modify the default configuration unless you modify the source tables (for example, add new columns or delete existing columns). However, dimensional attribute values (values of the columns with column type Dimensional Attribute) are specific to an implementation. Therefore, you must modify the prepackaged dimensional attribute values and create new ones according to your requirements.

You can also configure new source tables for use in the application. However, before that, you must import those source tables in the application. After importing a source table, you must configure it such that it can be used for creating subset maps or certain types of ABT variables. For example, you can configure a table such that it can be used for creating only behavioral variables or only time-based variables.

A table that is configured for creating a particular type of variable appears in the **Data sources** list for that variable type in the New Variables window.

For information about how to configure source tables and perform other related tasks, see *SAS Customer Analytics for Communications: User's Guide*.

Configuring the Columns of a Source Table

If you add new columns in a source table that you have already imported into the application, you must import those columns before you can use them in the application. After you import the columns, you have to configure them according to your requirements. For details about how to import columns, configure them, and perform other related tasks, see *SAS Customer Analytics for Communications: User's Guide*.

Configure Dimensional Attribute Values for a Column

The values of columns, which are of Dimensional Attribute type can be implementation-specific and therefore cannot be preconfigured. Therefore, you have to configure all dimensional values according to your requirements. These values are required when you define a subset criterion or specify one or more dimensions when you define an ABT variable.

Note: For some columns of the dimensional attribute type, the values might be preconfigured. However, you must verify these values and change them according to your requirements. SAS Customer Analytics for Communications enables you to either import or define dimensional values. For details about how to configure the dimensional attributes of a source column, see *SAS Customer Analytics for Communications: User's Guide*.

Creating a New Subset Map

Overview

A subset map is a group of tables that contain interrelated data pertaining to a particular subject of analysis. For example, the probable subjects of analysis for a communications service provider can be customers and subscriptions. A subset map is designed in a specific way so that users can apply filters on any column of subset map tables. You apply filters on a subset map through a subset criterion. For more information about the subset criterion, see the *SAS Customer Analytics for Communications: User's Guide*.

SAS Customer Analytics for Communications provides a number of predefined subset maps. For details, see "Predefined Subset Maps" on page 83. You can also create new subset maps according to your business requirements. Internally, the filter conditions in a subset criterion might be on columns of one or more tables in a subset map. In such a case, the final SQL statement (filter condition) for the subset criterion needs a relationship among all the involved subset map tables. The relationship is in the form of join conditions among the tables.

To create a new subset map, complete the following tasks:

1. Configure all the tables that you plan to use in the subset map definition for subset criterion usage.

   In other words, for each table that you plan to use in the subset map definition, select **Subset Criterion** on the **Direct/Subset Criterion** tab (in the Configure Table Usage window). For details, see the *SAS Customer Analytics for Communications: User's Guide*.

2. Define the subset map. For details, see "Define the Subset Map" on page 44.

3. Define the relationships among the tables in the subset map. For details, see "Define Relationships among Subset Map Tables" on page 44.
4. Configure the availability of the subset map for various subjects of analysis. For details, see "Configure the Availability of the Subset Map" on page 45.

Define the Subset Map

To define the subset map, you need to add a record in the SUBSET_FROM_PATH_MASTER table (available in the APDM library). The following table shows the SUBSET_FROM_PATH_MASTER table with sample records:

Table 4.3 *SUBSET_FROM_PATH_MASTER Table*

SUBSET_FROM_PATH_SK	FROM_PATH_SHORT_NM	FROM_PATH_DESC
7	Payment Mode Subscriptions or Customers	Subscriptions or Customers with Postpaid or Prepaid payment mode
10	Customers with Active Status	Active Customers
11	Subscriptions with Active Status	Active Subscriptions

For descriptions of the columns in the SUBSET_FROM_PATH_MASTER table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Define Relationships among Subset Map Tables

Add one or more records in the SUBSET_TABLE_JOIN_CONDITION table to define relationships among the tables in the subset map. These relationships among the tables help in constructing the FROM clause of the final SQL statement.

The following table shows the SUBSET_TABLE_JOIN_CONDITION table with sample records:

Table 4.4 *SUBSET_TABLE_JOIN_CONDITION Table*

SUBSET_TABLE_JOIN_CONDITION_SK	SUBSET_FROM_PATH_SK	JOIN_CONDITION_SEQUENCE_NUMBER	JOIN_TYPE	LEFT_TABLE_SK	LEFT_COLUMN_SK	RIGHT_TABLE_SK	RIGHT_COLUMN_SK
42	7	1	INNER	9	103	15	666
43	7	2	INNER	15	665	60	673
10	10	1	NONE	8	99	.	.

For descriptions of the columns in the SUBSET_TABLE_JOIN_CONDITION table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Configure the Availability of the Subset Map

A subset map can be linked to one or more subjects of analysis. This linking specifies that the tables in a subset map can be used to define subset criteria in projects with the corresponding subjects of analysis. You need to define these links between the subset map and various subjects of analysis in the SUBSET_FROM_PATH_X_LEVEL table.

The following table shows the SUBSET_FROM_PATH_X_LEVEL table with sample records:

Table 4.5 *SUBSET_FROM_PATH_X_LEVEL Table*

SUBSET_FROM_PATH_SK	LEVEL_SK	SELECT_SOURCE_ COLUMN_SK
7	4	102
7	1	103
10	1	99

For descriptions of the columns in the SUBSET_FROM_PATH_X_LEVEL table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Example: Creating a Subset Map

Overview

Configure a new subset map that enables the user to select the customers based on the following attributes:

- customer type
- customer tenure
- payment mode

In order to create this subset map that you need to use the following tables of the Foundation data mart: CUST_D, CUST_AGRMNT_D, and OFFER_BUNDLE_D. As a result, users can define the subset criterion based on the columns of these tables.

The subsequent topics explain how to configure this subset map.

Configure the SUBSET_FROM_PATH_MASTER Table

Assuming that 31 subset maps with SK in sequence are defined, add the following joining condition of the query in the SUBSET_FROM_PATH_MASTER table.

Table 4.6 *Sample Record in SUBSET_FROM_PATH_MASTER Table*

SUBSET_FROM_PATH_SK	FROM_PATH_SHORT_NM	FROM_PATH_DES
32	Customers with specific payment mode	Customers with either prepaid or postpaid payment mode

Configure the SUBSET_TABLE_JOIN_CONDITION Table

After you configure the values in the SUBSET_FROM_PATH_MASTER table, make sure that you enter correct values for joining condition in the SUBSET_TABLE_JOIN_CONDITION table. When a user defines a subset criterion based on a subset map, it is converted into a query. This query is configured in the SUBSET_TABLE_JOIN_CONDITION table.

To configure the SUBSET_TABLE_JOIN_CONDITION table, complete these steps:

1. Open the SOURCE_TABLE_MASTER table (in the APDM library). Identify the SK of each of the CUST_D, CUST_AGRMNT_D, and OFFER_BUNDLE_D tables. For example, the SK for each table can be as mentioned in the following table.

Table 4.7 *Sample Records in SOURCE_TABLE_MASTER Table*

SOURCE_TABLE_NM	SOURCE_TABLE_SK
CUST_D	8
CUST_AGRMNT_D	15
OFFER_BUNDLE_D	60

2. Identify the joining keys in these tables.
   - CUST_D and CUST_AGRMNT_D tables are joined based on the CUST_ID column.
   - CUST_AGRMNT_D and OFFER_BUNDLE_D tables are joined based on the OFFER_ID and BASE_OFFER_ID columns in the respective tables.
3. Open the SOURCE_COLUMN_MASTER table (in the APDM library). Identify the SK of the joining keys from the respective tables. For example, these SK values can be as shown in the table below.

Table 4.8 *Sample Records in the SOURCE_COLUMN_MASTER Table*

SOURCE_TABLE_NM	SOURCE_TABLE_SK	SOURCE_COLUMN_NM	SOURCE_COLUMN_SK
CUST_D	8	CUST_ID	99
CUST_AGRMNT_D	15	CUST_ID	666
CUST_AGRMNT_D	15	OFFER_ID	665
OFFER_BUNDLE_D	60	BASE_OFFER_ID	673

4. Open the SUBSET_TABLE_JOIN_CONDITION table (in the APDM library). Assuming that the maximum SK value for the SUBSET_TABLE_JOIN_COONDITION_SK column is 43, add the following two records in the SUBSET_TABLE_JOIN_CONDITION table.

Table 4.9 *Sample Records in the SUBSET_TABLE_JOIN_CONDITION Table*

SUBSET_ TABLE_ JOIN_ COONDITION SK	SUBSET_ FROM_ PATH_SK	JOIN_ CONDITION_ SEQUENCE_ NUMBER	JOIN_ TYPE	LEFT_ TABLE_SK	LEFT_ COLUMN_ SK	RIGHT_ TABLE_SK	RIGHT_ COLUMN_ SK
44	32	1	INNER	8	99	15	666
45	32	2	INNER	15	665	60	673

Configure the SUBSET_FROM_PATH_X_LEVEL Table

After you configure the subset map, you have to make it available in the SAS Customer Analytics for Communications interface. A subset map can be linked to one or more subject of analysis. The subset map that is created in this example can be made available only at customer level. Therefore, you have to identify the SK of the CUST_ID column.

To identify the SK of the customer level, complete these steps:

1. Open the LEVEL_MASTER table (in the APDM library). Identify the LEVEL_SK, which corresponds to LEVEL_KEY_COLUMN_NM as CUST_ID. For example, this SK value can be 1.
2. Insert following record in the SUBSET_FROM_PATH_X_LEVEL table (in the APDM library).

Table 4.10 *Sample Record in SUBSET_FROM_PATH_X_LEVEL Table*

SUBSET_FROM_PATH_SK	LEVEL_SK	SELECT_SOURCE_COLUMN_SK
32	1	99

Subset Query

Let us assume that a user defines a subset criterion based on the subset map that you have created, to filter customer who have postpaid payment mode. In this case, the following query is created. The SELECT_SOURCE_COLUMN_SK column is selected as a part of the final query.

```
proc sql noprint;

create table tmp_lib.d_135 as select distinct CUST_D.CUST_ID

From DIM.CUST_D

(where =(&DABT_LOAD_USER_INPUT_DTTM between VALID_START_DTTM and VALID_END_DTTM))

INNER join

DIM.CUST_AGRMNT_D

(where =(&DABT_LOAD_USER_INPUT_DTTM between VALID_START_DTTM and VALID_END_DTTM))

on (CUST_D.CUST_ID = CUST_AGRMNT_D.CUST_ID)INNER join

DIM.OFFER_BUNDLE_D

(where =(&DABT_LOAD_USER_INPUT_DTTM between VALID_START_DTTM and VALID_END_DTTM))
```

```
on (CUST_AGRMNT_D.OFFER_ID = OFFER_BUNDLE_D.BASE_OFFER_ID)

where ((kupcase(ktrim(kleft( OFFER_BUNDLE_D.BASE_OFFER_PYMNT_MODE_CD )))
IN %kupcase(%quote((&PST_PYMNT_MODE_VALS)))));
quit;
```

This query can be split into three parts:

1. The first part (the selection part) is the configuration made in the SELECT_SOURCE_COLUMN_SK column of the SUBSET_FROM_PATH_X_LEVEL table.
2. The second part (the FROM condition) is the configuration made in the SUBSET_TABLE_JOIN_CONDITION table.
3. The last part indicates the filters that you define for the subset criterion that you defined for the subset map.

Configure the Subset Map at Subscription Level

If you want to define the same subset map at subscription level, one of the methods is to join the SUBSCRP_D table instead of CUST_D with the CUST_AGRMNT_D table. The joining column will be the same, that is, CUST_ID. The rest of the configuration will be the same for the SUBSET_TABLE_JOIN_CONDITION table. However, the LEFT_TABLE_SK and LEFT_COLUMN_SK columns will correspond to the SK (for example, 9) of the SUBSCRP_D table and the SK (for example, 103) of CUST_ID column of the SUBSCRP_D table. In this case, the records from the SUBSET_TABLE_JOIN_CONDITION table would look as shown below:

Table 4.11 *Sample Records in SUBSET_TABLE_JOIN_CONDITION Table*

SUBSET_TABLE_JOIN_CONDITION_SK	SUBSET_FROM_PATH_SK	JOIN_CONDITION_SEQUENCE_NUMBER	JOIN_TYPE	LEFT_TABLE_SK	LEFT_COLUMN_SK	RIGHT_TABLE_SK	RIGHT_COLUMN_SK
44	32	1	INNER	9	103	15	666
45	32	2	INNER	15	665	60	673

Edit the record in the SUBSET_FROM_PATH_X_LEVEL table. The value for the SELECT_SOURCE_COLUMN_SK column should represent the SK (for example, 103) of the CUST_ID column from the SUBSCRP_D table. Add one more rows in the SUBSET_FROM_PATH_X_LEVEL table corresponding to subscription as a level. If 4 is the value of LEVEL_SK for SUBSCRP_ID level from the LEVEL_MASTER table and 102 is the SK of SUBSCRP_ID column from the SUBSCRP_D table, then the records in the SUBSET_FROM_PATH_X_LEVEL table would look as shown below:

Table 4.12 *Sample Records in SUBSET_FROM_PATH_X_LEVEL Table*

SUBSET_FROM_PATH_SK	LEVEL_SK	SELECT_SOURCE_COLUMN_SK
32	1	103
32	4	102

If you use the above subset map for the projects with Customer as the subject of analysis, then the record corresponding to LEVEL_SK equal to 1 will be selected in the final query. However, if you use the same subset map for the projects with Subscription as the subject of analysis, then the record corresponding to LEVEL_SK equal to 4 will be selected in the final query.

Note: You can join the input tables of a subset map through different joining conditions. You can also use LEFT and RIGHT joins depending on the type of data you expect in the final ABT. If you want to use a single table in a subset map, the joining condition should be marked as NONE with entries only under LEFT_TABLE_SK and LEFT_COLUMN_SK columns. The RIGHT_TABLE_SK and RIGHT_COLUMN_SK columns should have Null values.

Creating a Subject Group

Overview

A subject group is a predefined group of members that share a common set of attributes. For example, all customers with postpaid payment mode can be grouped together. A subject group, together with the subset criterion that you define, determines the records in the final ABT data set. When an ABT is built, only those records that satisfy the following criteria are used to populate the ABT:

- records that belong to the specified subject group
- records that satisfy the filter conditions specified in the subset criterion

For more information, see the *SAS Customer Analytics for Communications: User's Guide*.

Subject groups are associated with a particular purpose and subject of analysis. The subject group that you can select for a project depends on the purpose and the subject of analysis that you select for the project. For projects with subject of analysis as customer and subscription respectively, SAS Customer Analytics for Communications provides following predefined subject groups:

- Prepaid Customers or Subscriptions
- Postpaid Customers or Subscriptions

You can also create your own subject groups. To create a new subject group, complete these steps:

1. Create a subset criterion that defines the subjects expected by the subject group that you plan to create. For details, see "Create a Subset Criterion" on page 50.
2. Define a subject group based on the subset criterion that you created in step 1. For details, see "Define a Subject Group" on page 50.
3. Configure the availability of the subject group for certain combinations of purposes and subjects of analyses. For details, see "Configure the Availability of the Subject Group" on page 51.

Create a Subset Criterion

A subject group defines a group of members that share a common set of attributes. These common attributes are internally specified in the form of a subset criterion. Therefore, to create a subject group, you first need to create a subset criterion. This involves certain configurations in the SUBSET_QUERY_MASTER, SUBSET_FILTER_NODE, SUBSET_FILTER_NODE_EXPRESSION, and FILTER_NODE_EXPRSSN_X_VALUE tables. These tables are available in the APDM library. For descriptions of these tables and their columns, see *SAS Customer Analytics for Communications: Data Reference Guide*.

You can either manually edit these tables and enter the configuration details for the new subset criterion, or you can use the SAS Customer Analytics for Communications interface to create a subset criterion. When you create a subset criterion through the interface, these tables are automatically updated with the configuration details of the new subset criterion.

To create a new subset criterion (through the interface), complete these steps:

1. Log on to SAS Customer Analytics for Communications with an appropriate profile.
2. Create a project with the subject of analysis for which you want to configure the subject group. For instructions on how to create a project, see the *SAS Customer Analytics for Communications: User's Guide*.
3. Open the project, and then create a subset criterion with the required subset map and filters such that the subset criterion defines the members (with a common set of attributes) as required by the subject group. For instructions on how to create a subset criterion, see the *SAS Customer Analytics for Communications: User's Guide*.
4. Note the project ID that is assigned to the project, and then log off from SAS Customer Analytics for Communications.
5. Open the PROJECT_MASTER table (available in the APDM library). This table contains a record for each project that is created in the application. Locate the record corresponding to your project. You can use the project ID to locate the record.
6. For this record, note the value in the SUBSET_QUERY_SK column. The value in this column uniquely identifies the new subset criterion definition in the SUBSET_QUERY_MASTER table.
7. Set the value of the SUBSET_QUERY_SK column to NULL. This breaks the link between the subset criterion and the corresponding project. That means, when you delete the project, the associated subset criterion is not deleted. Save the changes, and then close the PROJECT_MASTER table.
8. Log on to SAS Customer Analytics for Communications with an appropriate profile.
9. Delete the project that you created in step 2.

Define a Subject Group

To define a subject group based on the subset criterion that you created in "Create a Subset Criterion," you need to insert a record in the SUBJECT_GROUP_MASTER table (available in the APDM library). While doing that, in the SUBSET_QUERY_SK column, insert the value that you noted in step 6 in "Create a Subset Criterion."

The following table shows the SUBJECT_GROUP_MASTER table with sample records:

Table 4.13 SUBJECT_GROUP_MASTER Table

SUBJECT_ GROUP_SK	SUBJECT_ GROUP_CD	SUBJECT_ GROUP_ SHORT_NM	SUBJECT_ GROUP_DESC	SUBSET_ QUERY_SK
1	PREPD	Prepaid Customers or Subscriptions	Customers or Subscriptions with Prepaid Payment Mode	8
2	PSTPD	Postpaid Customers or Subscriptions	Customers or Subscriptions with Postpaid Payment Mode	7

For descriptions of the columns in the SUBJECT_GROUP_MASTER table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Configure the Availability of the Subject Group

After you create a subject group, you must configure it such that it is available for all the desired combinations of purposes and subjects of analyses in a project. To do this, you need to enter appropriate records in the SUBJECT_GROUP_SPCFCN_DTL table (available in the APDM library). Typically, you should configure a subject group such that it is available in projects with the subject of analysis same as that of the project that you created in step 2 in "Create a Subset Criterion."

The following table shows the SUBJECT_GROUP_SPCFCN_DTL table with sample records:

Table 4.14 SUBJECT_GROUP_SPCFCN_DTL Table

PURPOSE_SK	LEVEL_SK	SUBJECT_GROUP_SK
3	1	1
4	1	2
5	4	1
9	1	1
9	4	1

Outcome-Based Filtering

Outcome-based filtering enables the user to select correct observations in the modeling and scoring ABTs. If for a given subject, the event under consideration is already satisfied within the performance window itself, then such subjects are removed from the

corresponding modeling or scoring ABT. For more information about outcome-based filtering, see the *SAS Customer Analytics for Communications: User's Guide*.

The values in the following columns of the PURPOSE_LEVEL_CONFIG_DTL table (available in the APDM library) determine whether outcome-based filtering is enabled for modeling and scoring ABTs (for a given combination of purpose and subject of analysis):

APPLY_OC_PST_IMPL_SBST_MDL_FLG
indicates whether outcome-based filtering is enabled for modeling ABTs.

APPLY_OC_PST_IMPL_SBST_SCR_FLG
indicates whether outcome-based filtering is enabled for scoring ABTs.

By default, for all purposes except for customer retention, cross-sell, and up-sell, the values of these columns are set to N. This is because, outcome-based filtering is applicable only to customer retention, cross-sell, and up-sell projects. To enable outcome-based filtering, set the value of these columns to Y.

Build-Date Cap

Build-date cap is applied while calculating the value of an outcome variable in the modeling ABT. The variables that are involved in the calculation of the outcome variable might refer to a period that falls before the modeling ABT build date or the scoring date. In this case, only the data pertaining to the period after the ABT build date or the scoring date is considered for the calculation. For more information about the build-date cap, see the *SAS Customer Analytics for Communications 5.3: User's Guide*. The APPLY_BLD_DT_CAP_OUTCM_VAR_FLG column of the PURPOSE_MASTER table (available in the APDM library) determines whether the build-date cap is enabled. By default, the value of the APPLY_BLD_DT_CAP_OUTCM_VAR_FLG column is set to Y only for customer retention, cross-sell, and up-sell purposes. This is because, the build-date cap is not applicable to customer segmentation, customer lifetime value, market basket analysis, and customer acquisition projects. To disable the build-date cap, set the value of the APPLY_BLD_DT_CAP_OUTCM_VAR_FLG column to N.

Defining Implicit Subset Criterion

Overview

A subset criterion enables you to define the target population for your project. Certain business rules need to be applied to the base population to derive the correct target population. These business rules are mandatory and need to be applied irrespective of whether a subset criterion is defined for a project. These business rules can be applied in the form of an implicit subset criterion. For example, while creating a scoring ABT for a customer retention project, it is recommended to consider customers who have the active status. To enforce this business rule, an implicit subset criterion needs to be defined. An implicit subset criterion depends on the purpose and the subject of analysis. SAS Customer Analytics for Communications provides following predefined implicit subset criteria:

- Active Customers

- Active Subscriptions
- Fresh Prospects
- Contacted Prospects

You can also define your own implicit subset criterion. Creating an implicit criterion involves the following tasks:

1. Create an implicit subset criterion that defines the business rule that you want to apply to select the right population. For details, see "Create an Implicit Subset Criterion" on page 53.
2. Configure the availability of the implicit subset criterion for certain combinations of purposes and subjects of analyses. For details, see "Configure the Availability of the Implicit Subset Criterion" on page 53.

Create an Implicit Subset Criterion

An implicit subset criterion determines a group of members that satisfy certain business rules or attributes. These common attributes are internally specified in the form of a subset criterion. Therefore, to create an implicit subset criterion, you first need to create a subset criterion. This task involves certain configurations in the SUBSET_QUERY_MASTER, SUBSET_FILTER_NODE, SUBSET_FILTER_NODE_EXPRESSION, and FILTER_NODE_EXPRSSN_X_VALUE tables. These tables are available in the APDM library. For descriptions of these tables and their columns, see *SAS Customer Analytics for Communications: Data Reference Guide*.

You can either manually edit these tables and enter the configuration details for the new subset criterion, or you can use the SAS Customer Analytics for Communications interface to create a subset criterion. When you create a subset criterion through the interface, these tables are automatically updated with the configuration details of the new subset criterion. For details, see "Create a Subset Criterion" on page 50.

Configure the Availability of the Implicit Subset Criterion

After you have created an implicit subset criterion, you must configure it such that it is available for all the desired combinations of purposes and subjects of analyses in a project. To do this, you need to enter appropriate records in the IMPLICIT_SUBSET_SPCFCTN_DTL table (available in the APDM library). Typically, you should configure an implicit subset criterion such that it is available in projects with the subject of analysis same as that of the project for which you created the implicit subset criterion as explained above.

Note: The implicit business rules that are to be applied to a project can be different for modeling and scoring ABTs. Therefore, you must configure the implicit subset criterion for the modeling and scoring ABTs accordingly. To do so, in the IMPLICIT_SUBSET_SPCFCTN_DTL table, specify MDL or SCR as the value for the APPLICABLE_FOR_PROCESS_TYPE_CD column for modeling and scoring ABT respectively.

The following table shows the IMPLICIT_SUBSET_SPCFCTN_DTL table with sample records:

Table 4.15 IMPLICIT_SUBSET_SPCFCTN_DTL Table

PURPOSE_SK	LEVEL_SK	APPLICABLE_FOR_PROCESS_TYPE_CD	SUBSET_QUERY_SK
3	1	SCR	10
4	1	SCR	10
17	10	SCR	50
17	10	MDL	51
15	1	SCR	10
16	1	SCR	10

Creating an Implicit Variable

Overview

Similar to an implicit subset criterion, there can be some business rules, which require some mandatory variables to be a part of the modeling and or scoring ABT. These mandatory variables are called implicit variables. After you configure these variables, they are automatically added in the modeling and or scoring ABT. You will not able to edit or delete such variables from the SAS Customer Analytics for Communications interface.

For example, while creating an ABT for customer retention, the subscription's total revenue in past 6 weeks or last month's revenue needs to be present in a scoring ABT. Therefore, these variables need to be configured as implicit variables. The implicit variables that you configure, are added by default in the modeling ABT. You can decide whether you want to add them in the scoring ABT. An implicit variable can be of any standard type of variable, which can be created by using the SAS Customer Analytics for Communications interface.

SAS Customer Analytics for Communications provides the following predefined implicit variables:

- Customer's First Activation Date
- Customer's Termination Date
- Customer's tenure till ABT building/reference date in months
- Customer's tenure till ABT building/reference date in weeks
- Total revenue generated by a subscription over last 6 months
- Total revenue generated by a subscription over last 6 weeks

You can also create your own implicit variable. Creating a new implicit variable involves the following tasks:

1. Create an ABT variable that satisfies a given business rule. For details, see "Create an Implicit ABT Variable" on page 55.

2. Configure the availability of the implicit variable for certain combinations of purposes, subjects of analysis and the time grain. For details, see "Configure the Availability of the Implicit Variable" on page 55.

Create an Implicit ABT Variable

An implicit variable is the variable, which is mandatory for the modeling and or scoring ABT for given purpose, subject of analysis, and time grain. Therefore, to create an implicit variable, you first need to create an ABT variable. This involves certain configurations in the VARIABLE_MASTER, MODELING_ABT_MASTER, MODELING_ABT_X_VARIABLE tables and any of the BEHAVIORAL_VARIABLE, RECENT_VARIABLE, SUPPLEMENTARY_VARIABLE, DERIVED_VARIABLE and DERIVED_VAR_X_EXPRESSION_VAR tables based on the type of implicit variable that you want to create. These tables are available in the APDM library. For descriptions of these tables and their columns, see *SAS Customer Analytics for Communications: Data Reference Guide*.

You can either manually edit these tables, enter the configuration details for the new ABT variable or you can use the SAS Customer Analytics for Communications interface to create an ABT variable. When you create a variable through the interface, these tables are automatically updated with the configuration details of the new variable.

To create a new ABT variable (through the interface), complete these steps:

1. Log on to SAS Customer Analytics for Communications with an appropriate profile.
2. Create a project with the subject of analysis for which you want to configure the implicit variable. For instructions about how to create a project, see the *SAS Customer Analytics for Communications: User's Guide*.
3. Open the project, and then create an ABT. Save the ABT details. Create a variable that you want to configure as an implicit variable. For instructions about how to create an ABT and its variables, see the *SAS Customer Analytics for Communications: User's Guide*.
4. Note the project ID that is assigned to the project, and then log off from SAS Customer Analytics for Communications.
5. Open the PROJECT_MASTER table (available in the APDM library). This table contains a record for each project that is created in the application. Locate the record corresponding to your project. You can use the project ID to locate the record.
6. Open the MODELING_ABT_MASTER table. Set the value of the PROJECT_SK column to NULL. This breaks the link between the modeling ABT and the corresponding project. That means, when you delete the project, the associated modeling ABT is not deleted. Save the changes, and then close the PROJECT_MASTER and MODELING_ABT_MASTER table.
7. Log on to SAS Customer Analytics for Communications with an appropriate profile.
8. Delete the project that you created in step 2.

Configure the Availability of the Implicit Variable

After you have created an implicit variable, you must configure it such that it is available for all the desired combinations of purposes and subjects of analysis and time grains in a project. To do this, you need to enter appropriate records in the IMPLICIT_VAR_SPECIFICATION_DTL table (available in the APDM library).

Typically, you should configure an implicit variable such that it is available in projects with the subject of analysis same as that of the project that you created as explained above. Whenever you configure an implicit variable, it is automatically added in the modeling ABT. You can decide whether you want to include it in the scoring ABT. You should also configure whether the implicit variable is an outcome variable.

The following table shows the IMPLICIT_VAR_SPECIFICATION_DTL table with sample records:

Table 4.16 *IMPLICIT_VAR_SPCFCTN_DTL Table*

PURPOSE_SK	LEVEL_SK	AVAILBLE_FOR_ABT_TIME_GRAIN_SK	VARIABLE_SK	MDL_ABT_OUTCOME_VARIABLE_FLG	APPLICABLE_FOR_SCR_PROCESS_FLG	APPLICABLE_FOR_ACT_PROCESS_FLG
6	4	1	10000003	N	Y	N
10	4	1	10000003	N	Y	N
12	4	1	10000003	N	Y	N
15	1	1	100000007	N	N	N

Creating a Post-Action Macro

Overview

SAS Customer Analytics for Communications provides a framework to perform tasks starting from project creation to scoring template execution. There can be a situation in which certain additional processing is required for a project, ABT, model, published model, or a scoring template. To do so, SAS Customer Analytics for Communications provides the flexibility to execute certain additional processes along with the predefined process to arrive at the desired result. These processes can be run as post-action macros. You can run these macros at certain instances, termed as actions. After the specific action is complete, you can run the macros that you have configured.

For projects with customer as a subject of analysis and Customer Lifetime (CLTV) value as purpose, SAS Customer Analytics for Communications provides following predefined post-action macros:

cac_convert_lftm_val
: After the scoring template is executed successfully for CLTV, this macro converts the customer's expected tenure from months or weeks to days depending on the base offer payment mode of the customer.

cac_modify_timeids
: After the CLTV ABT is built, this macro converts the customer's activation and deactivation dates to date format from datetime format. This macro also assigns the date9. format and informat to them.

You can also create your own post-action macro. Creating a new post-action macro involves the following tasks:

1. Create the post-action macro, which accesses the keys of the action after which the macro will be executed. For details, see "Create a Post-Action Macro" on page 57.
2. Configure the availability of the post-action macro for certain combinations of action, purpose, and the macro name. For details, see "Configure the Availability of the Post-Action Macro" on page 58.

Create a Post-Action Macro

These macros are executed after a specific action from the SAS Customer Analytics for Communications interface is complete. SAS Customer Analytics for Communications provides a predefined set of actions after which a post-action macro can be executed. These are non-editable actions.

The following table lists the predefined actions that can be performed through the SAS Customer Analytics for Communications interface. You must not change the details in this table or add new actions to it.

Note: You can define a post-action macro for any of the actions that are listed below.

Table 4.17 *ACTION_MASTER Table*

ACTION_SK	ACTION_CD	ACTION_SHORT_NM
1	DEL_PRJ	Delete Project
2	NEW_PRJ	New Project
3	NEW_ABT	New ABT
4	DEL_ABT	Delete ABT
5	NEW_MDL	New Model Capture
6	EDT_MDL	Edit Model
7	DEL_MDL	Delete Model
8	PBL_MDL	Publish Model for Scoring
9	NEW_SCR_TMPLT	New Scoring Template
10	DEL_SCR_TMPLT	Delete Scoring Template
11	BLD_MDL_ABT	Build Modeling ABT
12	SCR_JOB	Scoring Job Execution
13	ACT_JOB	Actual Value Job Execution

Write the SAS code in the form of a SAS macro to execute the extra processes that you run. Save the SAS code at following location:

Windows

```
<SAS Home>/SASFoundation/9.3/cacsrv/ucmacros
```

UNIX

```
<SAS Home>/SASFoundation/9.3/ucmacros/cacsrv
```

Configure the Availability of the Post-Action Macro

After you create the SAS macro, you must configure it and make it available for all desired combinations of purpose and the action after which it is to be called. To do so, you need to enter appropriate records in the POST_ACTION_MACRO_CONFIG_DTL table (available in the APDM library).

The following table shows the POST_ACTION_MACRO_CONFIG_DTL table with sample records:

Table 4.18 *POST_ACTION_MACRO_CONFIG_DTL Table*

ACTION_SK	PURPOSE_SK	POST_ACTION_MACRO_NM
12	15	cac_convert_lftm_val
12	16	cac_convert_lftm_val
11	15	cac_modify_timeids
11	16	cac_modify_timeids

Restarting the Server-Side Web Application

When you configure objects such as subject groups or time periods, which cannot be configured from the SAS Customer Analytics for Communications interface, certain tables (in the APDM library) of the Application data mart are updated. In order to make these changes effective, you have to restart the server-side Web application. For example, you define new time periods in the SOURCE_TIME_PERIOD table. For details, see "Configuring Time Periods for Behavioral Variables" on page 41. In order that these time periods are available when you define a behavioral variable by using the SAS Customer Analytics for Communications interface, you have to restart the server-side Web application.

Chapter 5

Performing Middle-Tier Administrative Tasks

Middle-Tier Administration Overview **59**

Roles and Capabilities **59**

Change the Owner of a Project **62**

Working with Software Component Properties **63**
Overview of Software Component Properties 63
Cust Analytics Comm Svr 5.3 63
Cust Analytics Communications Mid 5.3 64

Configuring the Logging Folder **66**

Middle-Tier Administration Overview

The middle-tier component synchronizes all components of SAS Customer Analytics for Communications and enables them to function together as an integrated system. It interacts with the Web-based user interface of SAS Customer Analytics for Communications and the SAS library. Also, it connects to the SAS run-time environment in order to run analytical processes.

You have to perform certain administrative activities in order to set up and configure the middle-tier component on various servers such as the metadata server, the application server, and the SAS library. Also, you have to maintain the error logs that are generated by SAS Customer Analytics for Communications.

Roles and Capabilities

Different users of SAS Customer Analytics for Communications might have access to different functionality depending on their assigned roles. Each role is mapped to a set of predefined capabilities. A capability, also known as an application action, defines the operations that a user can perform. One or more roles can be assigned to a user who can access SAS Customer Analytics for Communications. If multiple roles are assigned to a user, then the least restrictive capability of each role is granted to the user.

The following table lists the capabilities that are defined in SAS Customer Analytics for Communications:

Note: The project-related capabilities enable a user to view or work on only those projects (and their components such as subset criteria, analytical base tables (ABTs), and models) that the user has created. A user cannot view or work on the projects that other users have created.

TIP The Create or the Delete capability on an object does not enable a user to create or delete the object unless the user also has the View capability on that object.

Table 5.1 *Capabilities in SAS Customer Analytics for Banking*

Workspace Name	Capability Name	Description
Projects	View Project	Enables a user to view projects.
	Create Project	Enables a user to create and edit projects.
	Delete Project	Enables a user to delete projects.
	View Subset Criterion	Enables a user to view subset criteria.
	Create Subset Criterion	Enables a user to create, save, edit, and delete subset criteria and filter nodes.
	View Modeling ABT	Enables a user to view modeling ABTs.
	Create Modeling ABT	Enables a user to create, edit, and delete modeling ABTs and variables. This capability also enables the user to import variables and share ABTs with other users.
	Build Modeling ABT	Enables a user to build modeling ABTs.
	Register Modeling ABT	Enables a user to register modeling ABTs with the SAS Metadata Server.
	View Model	Enables a user to view analytical models.
	Capture Model	Enables a user to capture and edit a model's information.
	Delete Model	Enables a user to delete a model's captured information.
	Publish Model	Enables a user to publish models for scoring.
Scoring	View Scoring Template	Enables a user to view scoring templates.
	Create Scoring Template	Enables a user to create and edit scoring templates.
	Delete Scoring Template	Enables a user to delete scoring templates.
Administrative	Manage Configuration of the Application Mart	Enables a user to import and configure source tables, columns, dimensional attribute values, and time periods for behavioral variables for use in SAS Customer Analytics for Banking.

SAS Customer Analytics for Communications ships with three predefined roles. Each role is pre-assigned a set of capabilities. The following table shows the default mapping of roles to capabilities:

Note: Using SAS Management Console, you can modify the roles and specify the capabilities according to your requirements. You can also define new roles. For more information about defining users and granting roles and capabilities, see *SAS 9.3 Management Console: Guide to Users and Permissions*. This document is available at **http://support.sas.com/documentation/cdl/en/mcsecug/63190/PDF/default/mcsecug.pdf**.

Table 5.2 *Mapping of Roles to Capabilities*

	Roles		
Capability	**Cust Analytics Communication: Business Analysis**	**Cust Analytics Communication: Data Analysis**	**Cust Analytics Communication: Administration**
View Project	Y	Y	Y
Create Project	Y	Y	N
Delete Project	Y	Y	N
View Subset Criterion	Y	Y	Y
Create Subset Criterion	Y	Y	N
View Modeling ABT	Y	Y	Y
Create Modeling ABT	Y	Y	N
Build Modeling ABT	Y	Y	N
Register Modeling ABT	Y	Y	N
View Model	Y	N	Y
Capture Model	Y	N	N
Delete Model	Y	N	Y
Publish Model	Y	N	N
View Scoring Template	Y	N	Y
Create Scoring Template	Y	N	N
Delete Scoring Template	Y	N	N
Manage Configuration of Application Data Mart Tables	N	N	Y

Change the Owner of a Project

When a user creates a project, the user is the default owner of the project. Only the owner of a project can view and work on the project. Other users cannot view or work on the project. However, according to the business requirements, you can assign the ownership of the project to another user.

Note: Only one user at a time can be the owner of a project.

To change the owner of a project:

1. Log on to SAS Data Integration Studio, and then connect to the desired metadata server.
2. On the menu, select **Tools** ⇨ **Code Editor**.
3. Type the following code in the Code Editor, and then click **Run**. If asked for a user name and password, use an appropriate user (for example, sassrv) to run the code.

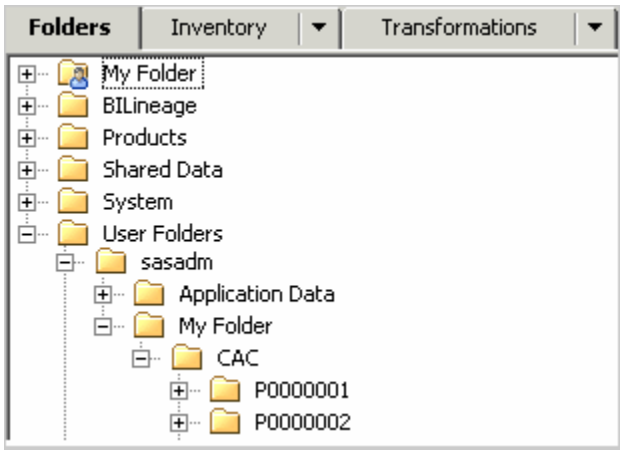
   In the code, replace <*Project ID*> with the ID of the project. Replace <*User ID*> with the user ID of the new owner.

   ```
   %cacinit;
   %dabt_change_project_ownership
   (m_project_id = <Project ID>, m_change_owner_to = <User ID>);
   ```

4. Move the project folder from the CAC folder of the previous owner to the CAC folder of the new owner. For instructions on how to move a folder to another folder in SAS Data Integration Studio, see the SAS Data Integration Studio Help.

   When a user creates the first project (in the Projects workspace), a folder structure for that user is automatically created in the SAS metadata. For each subsequent project that the user creates, a folder is automatically created under the CAC folder. The name of the folder is the same as the project ID. For example, for sasdemo, the folder structure looks similar to the following in SAS Data Integration Studio:

   Display 5.1 *Folder Structure for sasadm*

   Note: If the new owner of the project has not yet created a project, you must first manually create the required folder structure (up to the CAC folder) in SAS Data Integration Studio. Then, move the project folder from the CAC folder of the previous owner to the CAC folder of the new owner.

5. Close SAS Data Integration Studio.

Working with Software Component Properties

Overview of Software Component Properties

SAS Customer Analytics for Communications has properties defined for the following software components:

- Cust Analytics Comm Svr 5.3
- Cust Analytics Communications Mid 5.3

It is recommended that you do not modify the default values of these properties. However, if you modify the values, you must redeploy SAS Customer Analytics for Communications on your Web application server.

Cust Analytics Comm Svr 5.3

To view or modify the properties of the Cust Analytics Comm Svr 5.3 software component:

Note: You must not modify the default values of the following properties. Doing so might result in unexpected behavior or might even cause the application to fail. However, if you accidentally change these values, you can replace the changed values with the default values. For details about default values, see Table 5.3 on page 63.

- cacsrvc.dabt.app_init_stmt
- cacsrvc.dabt.buildabtlocation

1. In SAS Management Console, on the **Plug-ins** tab, expand the following sequence of nodes: **SAS Management Console** ⇨ **Application Management** ⇨ **Configuration Manager**.
2. Right-click **Cust Analytics Comm Svr 5.3**, and then select **Properties**.
3. In the Properties dialog box, click the **Advanced** tab. On this tab, you can see a list of the properties and their values. To change a value, in the Property Value column, click the value and enter the desired value. Click **OK**.

   The following table lists the properties of the Cust Analytics Comm Svr 5.3 software component:

Table 5.3 *Properties of Cust Analytics Comm Svr 5.3*

Property	Default Value	Description
cacsrvc.config.dir	C:\SAS\Config\Lev1	SAS configuration directory, including level.
cacsrvc.dabt.adminpath	<SDW configured path>/admin	Location where the temporary data sets for the Administrative workspace are created.

Property	Default Value	Description
cacsrvc.dabt.apdm.lib	APDM	The name of the application data model library that is used by the Web Application.
cacsrvc.dabt.app_init_stmt	%cacinit;	Statement to initialize the workspace server connection.
cacsrvc.dabt.buildabtlocation	build_modeling_abt	Name of the folder that is created under a user's project to store the built ABT data set.
cacsrvc.dabt.emn_repos_nm	Foundation	Repository name.
cacsrvc.dabt.projectpath	<SDW configured path>/project	Location where temporary data sets for the Projects workspace are created.
cacsrvc.dabt.scoringpath	<SDW configured path>/scoring	Location where temporary data sets for the Scoring workspace are created.
cacsrvc.dabt.servercontext	SASApp	Name of the application server context.
cacsrvc.dabt.usepooled	true	Indicates whether the pooled workspace server is used (recommended).
cacsrvc.version	5.3	Indicates the current version of the product.

Cust Analytics Communications Mid 5.3

To view or modify the properties of the Cust Analytics Communications Mid 5.3 software component:

Note: You must not modify the default values of the following properties. Doing so might result in unexpected behavior or might even cause the application to fail. However, if you accidentally change these values, you can replace the changed values with the default values. For default values of these properties, see Table 5.4 on page 65.

- cacmid.flex.APPLICATION_IDENTIFIER
- cacmid.flex.APPLICATION_NAME
- cacmid.flex.DOCUMENTATION_MACRO_VERSION
- cacmid.flex.DOCUMENTATION_PRODUCT_NAME
- cacmid.flex.QUICK_HELP_URL
- cacmid.flex.SESSION_TOUCH_ENABLED
- cacmid.flex.SESSION_TOUCH_URL
- cacmid.flex.SUPPORT_URL
- cacmid.flex.THEMES_DEFAULT
- cacmid.flex.THEMES_ENABLED
- cacmid.flex.TIMEOUT_ENABLED

1. In SAS Management Console, on the **Plug-ins** tab, expand the following sequence of nodes: **SAS Management Console** ⇨ **Application Management** ⇨ **Configuration Manager** ⇨ **SAS Application Infrastructure**.
2. Right-click **Cust Analytics Communications Mid 5.3**, and then select **Properties**.
3. In the Properties dialog box, click the **Advanced** tab. On this tab, you can see a list of the properties and their values. To change a value, in the Property Value column, click the value and enter the desired value. Click **OK**.

   The following table lists the properties of the Cust Analytics Communications Mid 5.3 software component:

Table 5.4 *Properties of Cust Analytics Communications Mid 5.3*

Property	Default Value	Description
cacmid.dabt.APP_ META_FOLDER	CAC	Name of the folder that is created under the **SAS Folders** ⇨ **User Folders** ⇨ ***<User ID>*** ⇨ **My Folder** folder on the **Folders** tab in SAS Management Console. This folder stores the metadata of the registered ABTs.
cacmid.flex.APPLICA IDENTIFIER	SASCAC	A unique identifier for the application.
cacmid.flex.APPLICA NAME	CAC	Name of the application.
cacmid.flex.DOCUME MACRO_VERSION	5.3	Version number of the product documentation. Used to locate the user's guide.
cacmid.flex.DOCUME PRODUCT_NAME	CACDOC	Product name corresponding to the product documentation. Used to locate the user's guide.
cacmid.flex.HELP_ ENABLED	true	This property is used internally and its value must be set to TRUE.
cacmid.flex.LOCAL E	en_US	The locale that is used if no locale is defined in the request or the locale that is defined in the request is not supported.
cacmid.flex.LOCALIZ ENABLED	true	Indicates whether localization is enabled.
cacmid.flex.SESSION_ TOUCH_ ENABLED	true	Enables you to configure the session touch action for synchronizing the client and server sessions.
cacmid.flex.SESSION_ TOUCH_URL	StreamContentServlet? NOOP=true	Configures the session touch URL.

Property	Default Value	Description
cacmid.flex.SUPPORT_URL	http://support.sas.com/documentation/cac.html	Link to the product support page.
cacmid.flex.THEMES_DEFAULT	Corporate	Specifies the default Flex theme.
cacmid.flex.THEMES_ENABLED	true	Indicates whether Flex themes are enabled.
cacmid.flex.TIMEOUT_ENABLED	true	Indicates whether session time-out is enabled.
cacmid.flex.TIMEOUT_PINGWINDOW_INTERVAL	10	Intervals (in milliseconds) at which the application checks whether the application server is active.
cacmid.flex.TIMEOUT_WARNING_DURATION	300	Duration (in seconds) for which the warning dialog box is displayed before the session times out.
cacmid.flex.USER_GUIDE_URL	cacug	This property is used internally.

Configuring the Logging Folder

The logs of the middle-tier component are maintained in the **<SAS configuration directory>/Lev1/Web/Logs** folder. For example, on a Windows computer, this location can be **C:/SAS/Config/Lev1/Web/Logs**.

The logs of the middle-tier component use log4j and therefore support the following features:

- Increase or decrease the level of detail that is generated in the logs.
- Define the location where the logs can be generated.
- Select the preferred format for the log files.

The log configuration file is maintained in the **<SAS configuration directory>/Lev1/Web/Common/LogConfig** folder. You can change the logging configuration for the middle-tier component. However, before you do so, see the log4j documentation. For more information, see **http://logging.apache.org/log4j/**.

Chapter 6

Scoring

The Scoring Process **67**
Overview 67
The Scoring Job 68
Scores Aggregation 70
Deleting a Scoring Model and the Associated Data 71

The Scoring Process

Overview

Scoring is the process of applying an analytical model to new data in order to compute outputs. In SAS Customer Analytics for Communications, scoring is performed with the help of a scoring template. A scoring template enables you to associate the scoring process with a model that has been published for scoring.

The scoring process involves the following tasks:

1. Create a scoring template for a model that has been published for scoring.

   You create a scoring template in the Scoring workspace of the SAS Customer Analytics for Communications interface. By default, business analysts have the required capabilities to view, create, and delete scoring templates. For information about how to create a scoring template, see the *SAS Customer Analytics for Communications: User's Guide*.

2. Schedule the scoring_run_job_exec_m_<scoring_template_id>.sas scoring job.

   For each scoring template, SAS Customer Analytics for Communications automatically creates a scoring job in the **<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsComm/5.3/scoring/<Scoring template ID>/scr_act_run_exported_code** folder. **<Scoring template ID>** represents the scoring template identification number—a unique number that is automatically assigned to each scoring template when it is created.

   You can use the scoring job to generate the predicted outcome values or scores for the current list of subject of analysis.

   SAS Customer Analytics for Communications 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 at the end of a month after the input mart is successfully loaded for that month.

Note: This job uses certain pre-assigned libraries defined in the SAS metadata server (such as, the APDM library). Therefore, ensure that these pre-assigned libraries are available to the session in which this job is run.

The Scoring Job

When the scoring_run_job_exec_<m_scoring_template_id>.sas is run, it performs the following tasks:

1. Identifies the scoring date—the date for which the scoring is to be performed.

   For the first scoring run

   The scoring job checks the INPUT_MART_LOAD_DATES table (available in the Control library) for the input mart load date—the date up to which the input mart contains data. This date is the scoring date for the first scoring run.

   Note: The INPUT_MART_LOAD_DATES table is updated every time the input mart is loaded. The only record in this table is a date. For example, 31 August 2011 indicates that the input mart contains data up to 31 August 2011. If the input mart has not been populated yet (that is, if there is no date in the INPUT_MART_LOAD_DATES table), the scoring job stops execution.

   For the subsequent scoring runs

   a. For a given scoring template, the scoring job checks the INPUT_MART_LOAD_DATES table for the input mart load date. Let us call it Date 1.

   b. For a given scoring template, the scoring job checks the SCORING_CONTROL_DETAIL table (available in the APDM library) for its corresponding last scoring date—the date as of which this scoring job was last run successfully. Let us call it Date 2. Then, the job compares Date 2 with Date 1. If Date 2 is earlier than Date 1 (Date 2 < Date 1), Date 1 is identified as the scoring date.

2. Builds the scoring analytical base table (ABT) with reference to the scoring date (identified in step 1 or 1.b). The scoring ABT is created with the significant variables —the variables that the model considers significant for scoring.

   The scoring ABT data set is created in the **<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsCommServer/5.3/scoring/<scoring_template_ID>/scoring_run** folder. The ABT data set is created with the name that was specified in the **Table name** field in the New Scoring Template window (in the Scoring workspace of the SAS Customer Analytics for Communications interface).

3. Applies the SAS Enterprise Miner scoring code on the scoring ABT.

   The scored ABT data set is created in the **<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsCommServer/5.3/scoring/<Scoring_template_ID>/scoring_run/scratch** folder. This ABT data set contains the scoring output (for example, predicted values or customer segments).

4. Writes the scoring output to an appropriate Application data mart tables (analytical results area of the Application data mart).

Table 6.1 *Application Data Mart Tables*

Table Name	Description
PREDICTED_SCORING_RESULT_CUST	Stores the values that are predicted by the analytical models for which the subject of analysis is Customer.
PREDICTED_SCORING_RESULT_SUB	Stores the values that are predicted by the analytical models for which the subject of analysis is Subscription.
PREDICTED_SCORING_RESULT_PRS	Stores the values that are predicted by the customer acquisition models for which the subject of analysis is Prospect Customer.
SEGMENT_SCORING_RESULT_CUST	Stores the segments that are created by segmentation models for which the subject of analysis is Customer.
RULES_SCORING_RESULT_CUST	Stores the association rules that are created by association models for which the subject of analysis is Customer.
RULES_SCORING_RESULT_SUB	Stores the association rules that are created by association models for which the subject of analysis is Subscription.

Note: The records in these tables are later used to populate the Foundation data mart.

5. Inserts a record in the SCORING_CONTROL_DETAIL table. This table contains a record for each scoring run. The record contains information such as the scoring date, the status of the scoring run, and so on. The following table shows the SCORING_CONTROL_DETAIL table with sample records:

Table 6.2 *SCORING_CONTROL_DETAIL Table*

SCORING_ CONTROL_ DETAIL_SK	SCORING_ TEMPLATE_ SK	SCORING_ AS_OF_ DTTM	SCORING_ STATUS_SK	SCORING_ PROCESSED_ DTTM	OUTCOME_ PERIOD_ END_DTTM	ACTUAL_ RESULT_ RUN_ ACTIVE_FLG
1	1	31-01-2009 23:59	1	14-11-2011 1:21	.	
2	2	31-01-2009 23:59	1	14-11-2011 5:31	.	
4	4	31-01-2009 23:59	1	14-11-2011 6:26	.	
5	6	31-01-2009 23:59	1	15-11-2011 2:37	.	

For descriptions of the columns in the SCORING_CONTROL_DETAIL table, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Note: If the execution of the scoring job fails, check the log files that are generated during the scoring process.

Scores Aggregation

Overview

SAS Customer Analytics for Communications supports analytical models at various levels such as customer, subscription, prospect customer, and so on. The scores are generated at the corresponding level.

Depending on the requirement, scores can be aggregated from one level to another. For example, customer churn is calculated at subscription level. Therefore, the churn scores are always generated at subscription level. However, business also needs to generate the churn scores at customer level. In order to do so, the scores at subscription level are aggregated at customer level. The scores are aggregated by using the weighted average method. The usage revenue in the past 6 months for postpaid subscriptions and in the past 6 weeks for prepaid subscriptions is used for assigning scores at customer level. For example, let us assume that the subscriptions generate revenue worth 15 USD, 85 USD, and 25 USD per month respectively. In this case, the churn scores can be derived by using the weighted average method as explained below:

CUST_MODEL_SCORE_NUM = SUM(WT_CHURN_SCORE) / SUM (REVENUE_AMOUNT)

Weighted churn score = Subscription probability * REVENUE_AMOUNT

Therefore, for the above example the churn score is calculated as follows:

CUST_MODEL_SCORE_NUM = (15*0.35+0.8*85+0.45*25) /125 = 0.676 ~ 0.68

This flags the customer as a churner.

The scores are aggregated mainly for customer churn, cross-sell, and up-sell models, which are built at subscription level.

The following high-level tasks that are performed when the scores are aggregated for a given model.

1. When a modeling ABT is defined for customer churn, cross-sell, or up–sell purposes, the S_SUM_PSU_URVAT_L6M or S_SUM_PRU_URVAT_L6W variables are added in the corresponding ABT depending on whether the ABT is to be built for postpaid or prepaid payment mode. These variables are configured as implicit variables for the above-mentioned purposes and also applicable for the scoring ABT. Therefore, when a scoring ABT is generated for any one of these purposes, these variables are also added in the scoring ABT. You have to configure these variables as the aggregation variables in the DABT_CONFIG_PARAM_DTLS table of the APDM library. Also, the value of these variables is used as the weighing value while aggregating the scores. If you want to add any other variable as a weighing variable, then you have to configure that variable as an implicit variable. For details, see "Creating an Implicit Variable" on page 54.

2. After the modeling ABT is built, a customer churn, cross-sell, or up-sell model is built on the modeling ABT. When the customer churn, cross-sell, or up-sell model's information is captured through the SAS Customer Analytics for Communications interface, the respective variable (S_SUM_PSU_URVAT_L6M or

S_SUM_PRU_URVAT_L6W) is marked as PREDICTED_OUTCOME_VALUE_1 in the MODEL_SCORING_OUTPUT_COLUMN table of the APDM library. This mapping is performed through the CAC_UPDATE_SCR_RSLT_COLUMN SAS macro, which resides at the following location:

Windows

<SAS Home>/SASFoundation/9.3/cacsrv/ucmacros

UNIX

<SAS Home>/SASFoundation/9.3/ucmacros/cacsrv

This macro contains the logic of mapping the corresponding weighing column with the scoring output column. If you want to change this mapping and point it to any other column, then modify this macro accordingly. If you want to change the name of the macro, then configure the new name as a value of the MCR_DABT_UPDATE_SCR_RSLT_COLUMN column of the DABT_CONFIG_PARAM_DTLS table of the APDM library.

3. After the model's information is captured and the model is published for scoring, the scoring template is created on the scoring model. The scoring template is executed and the scores at subscription level are written into the PREDICTED_SCORING_RESULT_SUB table of the APDM library.
4. The scores are also written to the writeback tables of the Foundation data mart by using the SAS Data Integration Studio jobs. In SAS Data Integration Studio, these jobs are available in the following path: **Products ⇨ SAS Customer Analytics for Communications 5.3 ⇨ Analytical Writeback ⇨ Jobs**. The cac_subscrp_model_score_dtl_job job aggregates the scores at customer level as mentioned above if the scores aggregation flags, namely CFDN_AGGR_CHURN_SCRS_FLG or CFDN_AGGR_XS_SCRS_FLG are set to Y. The customer-level aggregated scores are first written to PREDICTED_SCORING_RESULT_CUST table of the APDM library, and then to the CUST_MODEL_SCORE_DTL table of the Foundation data mart by using the cac_cust_model_score_dtl_job job. The subscription-level scores are also written back to the SUBSCRP_MODEL_SCORE_DTL table of the Foundation data mart.

Deleting a Scoring Model and the Associated Data

Overview

If you no longer need a scoring model and the associated data (scoring results), you can delete them manually. Also, if you want to retain the scoring model and delete only the associated data, you must manually delete the data from the analytical results area.

Delete a Scoring Model and the Associated Data

1. Start SAS Data Integration Studio, and then connect to the desired metadata server by opening an appropriate connection profile.

   Note: Make sure that you are connected to the correct server.

2. Select **Tools ⇨ Code Editor**.
3. In the Code Editor, type the following code, and then click **Run**. If asked for a user name and password, use an appropriate user (for example, sassrv) to run the code.

```
%cacinit;
%dabt_get_scoring_model_dtl;
```

This code generates an html file named scoring_model_details.html in the **<SAS configuration directory>/Lev1/AppData/SASCustAnalyticsCommServer/5.3/admin/<*User ID*>/log** folder.

<*User ID*> represents the user name that you used to run the code.

The scoring_model_details.html file contains a list of all the scoring models that are present in SAS Customer Analytics for Communications. The file also shows whether the scoring model is linked to a scoring template.

4. Open the html file, and note the SCORING_MODEL_SK of the scoring model that you want to delete.

   If the scoring model is linked to a scoring template (that is, if the value in the corresponding LINKED_TO_SCORING_TEMPLATE_FLG column is Y), note the ID of the scoring template (SCORING_TEMPLATE_ID). Then, complete step 5. However, if the value in the LINKED_TO_SCORING_TEMPLATE_FLG column is N, skip step 5.

5. Log on to SAS Customer Analytics for Communications as an administrator, and delete the scoring template that is linked to the scoring model. For instructions on how to delete a scoring template, see the *SAS Customer Analytics for Communications: User's Guide*.

6. In the SCORING_MODEL table, search for the SCORING_MODEL_SK of the scoring model that you want to delete. Also, note the corresponding value of the MODEL_KEY column.

   To retrieve the value of the MODEL_KEY column, you could use the following sample code in SAS Data Integration Studio:

   ```
   proc sql;

   select model_key from  apdm.Scoring_model where SCORING_MODEL_SK=

   <SCORING_MODEL_SK that you noted in step 4>;

   quit;
   ```

7. From the following tables, delete all the records that have the SCORING_MODEL_SK that you noted in the above step.

   Note: Make sure that you delete the records from the tables in the sequence that is mentioned below.

   a. MODEL_SCORING_OUTPUT_COLUMN
   b. SCORING_MODEL_RULE_DTLS
   c. SCORING_MODEL_RULE_MASTER
   d. SCR_MODEL_X_ACT_OUTCOME_VAR
   e. SCR_MODEL_X_SCR_INPUT_VARIABLE
   f. SCORING_MODEL_SEGMENT_MASTER
   g. SCORING_MODEL

8. In the MODEL_MASTER table, change the model status so that you can publish the model again.

   To change the model status, complete the following steps:

   a. In the MODEL_STATUS_MASTER, note the value of the MODEL_STATUS_SK column that corresponds to the MODEL_STATUS_CD column with value M_NTSBMT_SCR.

b. In the MODEL_MASTER table, search for the record for which the MODEL_SK value is same as the MODEL_KEY value that you noted in step 6. For this record, change the value of the MODEL_STATUS_SK to the value that you noted above.

To complete the tasks that are detailed in 8a and 8b, you could use the following sample code in SAS Data Integration Studio:

```
proc sql;

update apdm.Model_master set model_status_sk=

(select MODEL_STATUS_SK from apdm.Model_status_master

where MODEL_STATUS_CD='M_NTSBMT_SCR')

where MODEL_SK=<model_key retrived in step 6> ;

quit;
```

9. Close SAS Data Integration Studio.

Deleting Scoring Results

If you want to delete only the data (scoring results) associated with a scoring model and retain the model for later use, you must manually delete the data from the analytical results area (certain tables in the Application data mart). According to your requirements, you can delete all the data associated with a scoring model or delete only the data pertaining to specific time periods.

Based on the scoring as of date (SCORING_AS_OF_DTTM) for a scoring model (SCORING_MODEL_SK), you need to delete records from the appropriate Application data mart tables. The following table lists the tables that store scoring values for scoring models for different projects. For more information about these tables, see *SAS Customer Analytics for Communications: Data Reference Guide*.

Table 6.3 *Application Data Mart Tables*

Purpose	Subject of Analysis	Scoring Results Table Name
Segmentation Prepaid	Customer	SEGMENT_SCORING_RESULT_CUST
Segmentation Postpaid	Customer	SEGMENT_SCORING_RESULT_CUST
Churn Prepaid	Subscription	PREDICTED_SCORING_RESULT_SUB
Churn Postpaid	Subscription	PREDICTED_SCORING_RESULT_SUB
Cross-sell Prepaid	Customer	PREDICTED_SCORING_RESULT_CUST
Cross-sell Prepaid	Subscription	PREDICTED_SCORING_RESULT_SUB
Cross-sell Postpaid	Customer	PREDICTED_SCORING_RESULT_CUST

Purpose	Subject of Analysis	Scoring Results Table Name
Cross-sell Postpaid	Subscription	PREDICTED_SCORING_RESULT_SUB
Up-Sell Prepaid	Subscription	PREDICTED_SCORING_RESULT_SUB
Up-Sell Prepaid	Customer	PREDICTED_SCORING_RESULT_CUST
Up-Sell Postpaid	Subscription	PREDICTED_SCORING_RESULT_SUB
Up-Sell Postpaid	Customer	PREDICTED_SCORING_RESULT_CUST
MBA XSUS Postpaid	Customer Offer	RULES_SCORING_RESULT_CUST
MBA XSUS Postpaid	Customer Service First Activation Date	RULES_SCORING_RESULT_CUST
MBA XSUS Postpaid	Subscription Service First Activation Date	RULES_SCORING_RESULT_SUB
MBA XSUS Postpaid	Customer Offer Agreement Start Date	RULES_SCORING_RESULT_CUST
MBA XSUS Postpaid	Customer Service	RULES_SCORING_RESULT_CUST
MBA XSUS Postpaid	Subscription Service	RULES_SCORING_RESULT_SUB
MBA XSUS Prepaid	Customer Offer	RULES_SCORING_RESULT_CUST
MBA XSUS Prepaid	Customer Service First Activation Date	RULES_SCORING_RESULT_CUST
MBA XSUS Prepaid	Subscription Service First Activation Date	RULES_SCORING_RESULT_SUB
MBA XSUS Prepaid	Customer Offer Agreement Start Date	RULES_SCORING_RESULT_CUST
MBA XSUS Prepaid	Customer Service	RULES_SCORING_RESULT_CUST
MBA XSUS Prepaid	Subscription Service	RULES_SCORING_RESULT_SUB
Customer Lifetime Postpaid	Customer	PREDICTED_SCORING_RESULT_CUST
Customer Lifetime Prepaid	Customer	PREDICTED_SCORING_RESULT_CUST
Customer Acquisition	Prospect Customer	PREDICTED_SCORING_RESULT_PRS

Part 4

Appendixes

Appendix 1
Parameter Configuration *77*

Appendix 2
Predefined Subset Maps *83*

Appendix 3
Load Order Sequence *89*

Appendix 4
Troubleshooting *93*

Appendix 1

Parameter Configuration

As a post-installation task, you have to configure certain parameters that are defined in the DABT_CONFIG_PARAM_TBL table (available in the APDM library). The following table lists these parameters.

Table A1.1 *Parameters*

Parameter Name	Parameter Value	Is Editable?	Description
WEEK_END_DAY_NM	SATURDAY	N	For weekly grain ABT, only the last day of a week can be selected as the ABT build date in the Build ABT window. This parameter defines the last day of week. This configured day is the only weekday that user can select while choosing an ABT build date.
LIST_OF_MODEL_TBL_TO_DEL	MODEL_RULE_MASTER	N	List of tables from which the model-related records should be deleted while deleting the information of the captured model.
MCR_DABT_CREATE_EVENT_DATA	cac_create_event_data	N	The name of the solution-specific macro that creates event data. If you want to replace the cac_create_event_data macro with your own, the value of this parameter must be modified here.
MCR_DABT_UPDATE_ACT_RSLT_COLUMN	cac_update_act_rslt_column	N	The name of the solution-specific macro that updates the actual results. If you want to replace the cac_update_act_rslt_column macro with your own, you must modify the value of this parameter.

Parameter Name	Parameter Value	Is Editable?	Description
MCR_DABT_UPDATE_SCR_RSLT_COLUMN	cac_update_scr_rslt_column	N	The name of the solution-specific macro that updates the scoring results. If you want to replace the cac_update_scr_rslt_columnn macro with your own, you must modify the value of this parameter.
MCR_DABT_RULES_FILTER	cac_rules_filter	N	The name of the solution-specific macro that filters association rules. If you want to replace the cac_rules_filter macro with your own, you must modify the value of this parameter.
PR_UPSELL_PURPOSE_AGG_VAR_NM	S_SUM_PRU_URVAT_L6W	Y	Name of the variable that is to be used for aggregating subscription-level scores to customer-level for prepaid up-sell.
PR_RETENTION_PURPOSE_AGG_VAR_NM	S_SUM_PRU_URVAT_L6W	Y	Name of the variable that is to be used for aggregating subscription-level scores to customer-level for prepaid customer retention.
PR_CROSSSELL_PURPOSE_AGG_VAR_NM	S_SUM_PRU_URVAT_L6W	Y	Name of the variable that is to be used for aggregating subscription-level scores to customer-level for prepaid cross-sell.
PST_UPSELL_PURPOSE_AGG_VAR_NM	S_SUM_PSU_URVAT_L6M	Y	Name of the variable that is to be used for aggregating subscription-level scores to customer-level for postpaid up-sell.
PST_RETENTION_PURPOSE_AGG_VAR_NM	S_SUM_PSU_URVAT_L6M	Y	Name of the variable that is to be used for aggregating subscription-level scores to customer-level for postpaid customer retention.
PST_CROSSSELL_PURPOSE_AGG_VAR_NM	S_SUM_PSU_URVAT_L6M	Y	Name of the variable that is to be used for aggregating subscription-level scores to customer-level for postpaid up-sell.

Parameter Name	Parameter Value	Is Editable?	Description
PR_SEGMENT_ PURPOSE_CD	CSR	N	Purpose code that indicates prepaid customer segmentation. You must modify the value of this parameter if the purpose code for prepaid customer segmentation is changed in PURPOSE_MASTER.
PST_SEGMENT_ PURPOSE_CD	CSP	N	Purpose code that indicates postpaid customer segmentation. You must modify the value of this parameter if the purpose code for postpaid customer segmentation is changed in PURPOSE_MASTER.
PST_RETENTION_ PURPOSE_CD	CRP	N	Purpose code that indicates postpaid customer retention. You must modify the value of this parameter if the purpose code for postpaid customer retention is changed in PURPOSE_MASTER.
PR_RETENTION_ PURPOSE_CD	CRR	N	Purpose code that indicates prepaid customer retention. You must modify the value of this parameter if the purpose code for prepaid customer retention is changed in PURPOSE_MASTER.
PST_CROSS_SELL_ PURPOSE_CD	XSP	N	Purpose code that indicates postpaid cross-sell. You must modify the value of this parameter if the purpose code for postpaid cross-sell has changed in PURPOSE_MASTER.
PR_CROSS_SELL_ PURPOSE_CD	XSR	N	Purpose code that indicates prepaid cross-sell. You must change the value if the purpose code for prepaid cross-sell is changed in PURPOSE_MASTER.

Parameter Name	Parameter Value	Is Editable?	Description
PR_UP_SELL_PURPOSE_CD	USR	N	Purpose code that indicates prepaid up-sell. You must modify the value of this parameter if the purpose code for prepaid up-sell has changed in PURPOSE_MASTER.
PST_UP_SELL_PURPOSE_CD	USP	N	Purpose code that indicates prepaid up-sell. You must modify the value of this parameter if the purpose code for prepaid up-sell has changed in PURPOSE_MASTER.
PST_MBA_PURPOSE_CD	MBP	N	Purpose code that indicates postpaid association rule analysis. You must modify the value of this parameter if the purpose code for postpaid association rule analysis has changed in PURPOSE_MASTER.
PR_MBA_PURPOSE_CD	MBR	N	Purpose code that indicates prepaid association rule analysis. You must modify the value of this parameter if the purpose code for prepaid association rule analysis has changed in PURPOSE_MASTER.
PR_CLT_PURPOSE_CD	CLR	N	Purpose code that indicates prepaid customer lifetime value. You must modify the value of this parameter if the purpose code forprepaid customer lifetime has changed in PURPOSE_MASTER.
PST_CLT_PURPOSE_CD	CLP	N	Purpose code that indicates postpaid customer lifetime value. You must modify the value of this parameter if the purpose code for postpaid customer lifetime value has changed in PURPOSE_MASTER.

Parameter Name	Parameter Value	Is Editable?	Description
PST_CAQ_PURPOSE_CD	CAQ	N	Purpose code that indicates customer acquisition. You must modify the value of this parameter if the purpose code for customer acquisition has changed in PURPOSE_MASTER.
PST_PYMNT_MODE_ VALS	"POSTPAID"	Y	Valid values that indicate a postpaid offer payment mode.
PRE_PYMNT_MODE_ VALS	"PREPAID"	Y	Valid values that indicate a prepaid offer payment mode.
CUST_ACT_STATUS_ VALS	"CS_1"	Y	Valid values that indicate ACTIVE customer status.
SUBSCRP_ACT_STATUS_ VALS	"SS_1"	Y	Valid values that indicate ACTIVE subscription status.
NO_CAPMPAIGN_VALS	""	Y	Valid values that indicate that a customer has not been contacted for a campaign.
DATETIME_COLMN_ TYP_CD	DTT	N	Column data type code that indicates the datetime format for a date variable. The value of this parameter can be changed if the configuration of a timestamp date variable is changed in the COLUMN_DATA_TYPE_ MASTER table of the APDM library.
MBA_TRANSPOSE_ FLAG_VALUE	Y	N	The value of transpose flag in the output data set generated by the association node in SAS Enterprise Miner that is used for association analysis purpose.
SEQ_TRANSPOSE_ FLAG_VALUE	1	N	The value of transpose flag in the output data set generated by the association node in SAS Enterpise Miner that is used for sequence analysis purpose.

Appendix 2

Predefined Subset Maps

The following table lists the predefined subset maps that SAS Customer Analytics for Communications provides.

Table A2.1 *Predefined Subset Maps*

Subset Map	Purpose	Source Tables
Payment Mode Subscriptions or Customers	Enables you to select customers or subscriptions that have either prepaid or postpaid as the base offer payment mode. For example, you might want the target population to contain customers who have Postpaid as their Base Offer Payment Mode. To do so, when you define a subset criterion, select this subset map. When you define the filter condition, select the **Base Offer Payment Mode** variable, = (equal to) as the operator, and **Postpaid** as the value of the variable. *Note:* You have to configure the payment mode values (Postpaid and Prepaid) as the dimensional attribute values of the Base Offer Payment Mode column of the Offer Bundle Dimension table. For details, see the *SAS Customer Analytics for Communications: User's Guide*.	• SUBSCRP_D • CUST_AGRMNT_D • OFFER_D

Subset Map	Purpose	Source Tables
Customers with Active Status	Enables you to select customers whose status is active. For example, you want the target population to contain customers whose status is Active. To do so, when you define the subset criterion, select this subset map. When you define the filter condition, select the **Customer Status Code** variable, **=** (equal to) as the operator, and **Active** as the value of the variable. *Note:* You have to configure the customer status codes (for example, Active, Dormant, and Suspended) as the dimensional attribute values of the Customer Status Code column of the Customer Dimension table. For details, see the *SAS Customer Analytics for Communications: User's Guide*.	CUST_D
Subscriptions with Active Status	Enables you to select subscriptions whose status is active. For example, you want the target population to contain subscriptions whose status is Active. To do so, when you define the subset criterion, select this subset map. When you define the filter condition, select the **Subscription Status Code** variable, **=** (equal to) as the operator, and **Active** as the value of the variable. *Note:* You have to configure the subscription status codes (for example, Active, Dormant, and Suspended) as the dimensional attribute values of the Subscription Status Code column of the Subscription Dimension table. For details, see the *SAS Customer Analytics for Communications: User's Guide*.	SUBSCRP_D

Subset Map	Purpose	Source Tables
Customers or Subscription for BG	Enables you to select customers or subscriptions that belong to a specific business group (BG). For example, you want the target population to contain customers who belong to a business group whose ID is 7. To do so, when you define the subset criterion, select this subset map. When you define the filter condition, select **Business Group ID** as the variable, = (equal to) as the operator, and **7** as the value of the variable. *Note:* In order to define the subset criterion, you must configure appropriate values for the Business Group ID column of the Customer X Business Group Tagging table.	• SUBSCRP_D • CUST_X_BUSINESS_GROUP_BRIDGE
All Prospects	Enables you to select only prospective customers in your target population. You must use this subset map when you want to build the analytical model for customer acquisition.	PROSPECT_CUST_D
Customer Service Selection Postpaid	Enables you to create the modeling ABT for association rules analysis. You should select this subset map to analyze behavior of postpaid customers with reference to activation of services. In addition, after you choose this subset map, you can define filter conditions based on various service-level and customer-level attributes. The target population contains customers who satisfy the subset criteria. Therefore, the ABT would contain records of postpaid customers for the services that they activated at different points of time.	• CUST_D • SERVICE_D • PST_PD_SERVICE_ACTIVITY_B

Subset Map	Purpose	Source Tables
Customer Offer Selection Postpaid	Enables you to create the modeling ABT for association rules analysis. You should select this subset map to analyze behavior of postpaid customers with reference to agreement dates of offers. In addition, after you choose this subset map, you can define filter conditions based on various offer-level and customer-level attributes. The target population contains customers who satisfy the subset criteria. Therefore, the ABT would contain records of postpaid customers for the offers that they signed up for at different points of time.	• CUST_D • OFFER_D • PST_PD_CUST_OFFER_ SNPSHT_B
Subscription Service Selection Postpaid	Enables you to create the modeling ABT for association rules analysis. You should select this subset map to analyze behavior of postpaid subscriptions with reference to activation of services. In addition, after you choose this subset map, you can define filter conditions based on various service-level and subscription-level attributes. The target population contains the subscriptions that satisfy the subset criteria. Therefore, the ABT would contain records of postpaid subscriptions for the services that are activated for these subscriptions at different points of time.	• SUBSCRP_D • SERVICE_D • PST_PD_SERVICE_ACTIVITY_ B
Customer Service Selection Prepaid	Enables you to create the modeling ABT for association rules analysis. You should select this subset map to analyze behavior of prepaid customers with reference to activation of services. In addition, after you choose this subset map, you can define filter conditions based on various service-level and customer-level attributes. The target population contains the customers who satisfy the subset criteria. Therefore, the ABT would contain records of prepaid customers for the services that they activated at different points of time.	• CUST_D • SERVICE_D • PRE_PD_SERVICE_ACTIVITY_ B

Subset Map	Purpose	Source Tables
Customer Offer Selection Prepaid	Enables you to create the modeling ABT for association rules analysis. You should select this subset map to analyze behavior of prepaid customers with reference to agreement dates of offers. In addition, after you choose this subset map, you can define filter conditions based on various offer-level and customer-level attributes. The target population contains the customers who satisfy the subset criteria. Therefore, the ABT would contain records of prepaid customers for the offers that they signed up for at different points of time.	• CUST_D • OFFER_D • PRE_PD_CUST_OFFER_ SNPSHT_B
Subscription Service Selection Prepaid	Enables you to create the modeling ABT for association rules analysis. You should select this subset map to analyze behavior of prepaid subscriptions with reference to activation of services. In addition, after you choose this subset map, you can define filter conditions based on various service-level and subscription-level attributes. The target population contains the subscriptions that satisfy the subset criteria. Therefore, the ABT would contain records of prepaid subscriptions for the services that are activated for these subscriptions at different points of time.	• SUBSCRP_D • SERVICE_D • PRE_PD_SERVICE_ACTIVITY_ B
Rule-Based Customer Selection	Enables you to select customer population based on output of association rules analysis, which is rules. You should select this subset map to create a cross-sell or an up-sell model that is based on the association rules analysis models that you have created. In addition, after you choose this subset map, you can create filters based on various attributes of rules such as rule text, rule unit type (left hand or right hand of rule), and the ID values of services or offers, which are part of the rule.	• CUST_RULE_SCORE_DTL • ANALYTICAL_RULE_DTL • ANALYTICAL_MODEL_DTL

Subset Map	Purpose	Source Tables
Rule-Based Subscription Selection	Enables you to select subscription population based on output of association rules analysis, which is rules. You should select this subset map to create a cross-sell or an up-sell model that is based on the association rules analysis models that you have created. In addition, after you choose this subset map, you can create filters based on various attributes of rules such as rule text, rule unit type (left hand or right hand of rule), and the ID values of services or offers, which are part of the rule.	• SUBSCRP_RULE_SCORE_DTL • ANALYTICAL_RULE_DTL • ANALYTICAL_MODEL_DTL

Appendix 3

Load Order Sequence

Scheduling ETL Jobs

The tables that are listed below provide information about the ETL flows and the dependencies that exist between various jobs of a group. This information will help you schedule the ETL jobs. However, consider the following instructions before you schedule ETL jobs.

- Consider the order in which the jobs of a particular job group are to be run. The Group Order column represents the parent group, and the Level Within Group column indicates the sublevels within each group.
- Execute the jobs in one group only after all jobs of previous groups are executed. For example, the jobs in group 2 should be started only when all the jobs in group 1 are executed.
- Within a group, jobs should be run based on the value in the Level within Group column. For example, there can be two jobs within a group that have level number 1 and level number 2. In this case, Job 2 should be run only after Job 1 is complete.
- Within a group, jobs that have the same level number can be run in parallel.
- The bill-monthly and weekly jobs can be run in parallel.

Load Order Sequence for Bill-Monthly Jobs

Table A3.1 *Load Order for Bill-Monthly Jobs*

Group Order	Level within Group	Job Name
1	1	cfd_pst_pd_bill_usage_b_job
1	1	cfd_pst_pd_cust_acct_snpsht_b_job
1	1	cfd_pst_pd_cust_bill_b_job
1	1	cfd_pst_pd_cust_bill_nonusage_b_job
1	1	cfd_pst_pd_cust_interaction_b_job
1	1	cfd_pst_pd_cust_loyalty_b_job
1	1	cfd_pst_pd_equip_activity_b_job

Group Order	Level within Group	Job Name
1	1	cfd_pst_pd_payment_base_b_job
1	1	cfd_pst_pd_payment_drvd_b_job
1	1	cfd_pst_pd_service_activity_b_job
1	1	cfd_pst_pd_subscrp_bill_nonusage_b_job
1	1	cfd_pst_pd_subscrp_interaction_b_job
1	1	cfd_pst_pd_subscrp_loyalty_b_job
1	1	cfd_pst_pd_usage_b_job
1	1	cfd_pst_pd_cust_offer_snpsht_b_job
1	1	cfd_pst_pd_cust_snpsht_b_job
1	1	cfd_pst_pd_usage_summary_b_job

Load Order Sequence for Weekly Jobs

Table A3.2 *Load Order for Weekly Jobs*

Group Order	Level within Group	Job Name
1	1	cfd_pre_pd_cust_interaction_b_job
1	1	cfd_pre_pd_cust_loyalty_b_job
1	1	cfd_pre_pd_equip_activity_b_job
1	1	cfd_pre_pd_service_activity_b_job
1	1	cfd_pre_pd_subscrp_bucket_drvd_b_job
1	1	cfd_pre_pd_subscrp_bucket_usage_b_job
1	1	cfd_pre_pd_subscrp_interaction_b_job
1	1	cfd_pre_pd_subscrp_loyalty_b_job
1	1	cfd_pre_pd_subscrp_usage_drvd_b_job

Group Order	Level within Group	Job Name
1	1	cfd_pre_pd_usage_b_job
1	1	cfd_pre_pd_usage_recharge_b_job
1	1	cfd_pre_pd_cust_offer_snpsht_b_job
1	1	cfd_pre_pd_cust_snpsht_b_job
1	1	cfd_pre_pd_usage_summary_b_job

Load Order Sequence for Writeback Jobs

Table A3.3 *Load Order for Writeback Jobs for SAS Database*

Group Order	Level within Group	Job Name
1	1	cac_model_writeback_job
1	2	cac_cust_analytical_sgmt_dtl_job
1	3	cac_analytical_rule_master_dtl_job
1	4	cac_analytical_rule_dtl_job
1	5	cac_subscrp_rule_score_dtl_job
1	6	cac_cust_rule_score_dtl_job
1	7	cac_prspct_cust_model_score_dtl_job
1	8	cac_subscrp_model_score_dtl_job
1	9	cac_cust_model_score_dtl_job

Table A3.4 *Load Order for Writeback Jobs for Databases Other than SAS*

Group Order	Level within Group	Job Name
1	1	cac_model_writeback_job
1	2	cac_cust_analytical_sgmt_dtl_job
2	1	cac_analytical_rule_master_dtl_job
2	2	cac_analytical_rule_dtl_job
3	1	cac_subscrp_rule_score_dtl_job

Group Order	Level within Group	Job Name
3	1	cac_cust_rule_score_dtl_job
3	1	cac_prspct_cust_model_score_dtl_job
3	1	cac_subscrp_model_score_dtl_job
4	1	cac_cust_model_score_dtl_job

Appendix 4

Troubleshooting

Troubleshooting a Modeling ABT Building Process **93**

Troubleshooting the Scoring Process **95**

Troubleshooting the Nonexistence of the APDM Library **96**
Overview 96
Add Remote Engine LIBNAME Statement in the SASApp Autoexec File 96
Add Base LIBNAME Statement in the SAS Share Server Autoexec File 97
Restart Servers 97

Troubleshooting a Modeling ABT Building Process

If you encounter any execution errors while working in the SAS Customer Analytics for Communications interface, you need to get more details about these errors in order to debug them. These details are stored in the relevant log files and SAS data sets. To generate these log files and SAS data sets, you have to update the DABT_POST_INIT macro.

To generate the log files and SAS data sets, complete these steps:

1. Go to **<SAS Home>** and open the DABT_POST_INIT macro.
2. Change the value of the DABT_DEBUG_FLG and CAB_APPL_DEBUG_FLG variables to Y. The statements in the macro would look like as mentioned below:

   ```
   %let DABT_DEBUG_FLG = Y;

   %let CAB_APPL_DEBUG_FLG = Y;
   ```

3. Save the code.
4. Restart the services. For details, see "Restart Servers" on page 97.

This section describes the locations of the log files and the temporary data sets that are generated during a modeling ABT building process. It also gives details about the various log files. If a modeling ABT building process fails, you can use the information in these log files to troubleshoot the issue. Search these files for text containing ERROR or WARNING to get to the preliminary analysis of the problem.

During a modeling ABT building process, the temporary data sets are created in the following folder:

<Project path>/<Project ID>/build_modeling_abt/<ABT table name>/scratch

The log files are created in the following folder:

<Project path>/<Project ID>/build_modeling_abt/<ABT table name>/log

<Project path> represents the repository for the Projects workspace. For example, on a Windows computer, this location can be **C:/SAS/Config/Lev1/AppData/SASCustAnalyticsCommServer/5.3/project**. In this location, there is a folder for each project, with the corresponding project ID as the folder name. This folder contains the temporary data sets and the logs for all the activities that are performed in that project.

The project path is defined in the cacsrvc.dabt.projectpath property of the Cust Analytics Comm Server 5.3 software component. For details, see "Cust Analytics Comm Svr 5.3" on page 63.

<Project ID> is a unique number (PROJECT_SK) that is automatically assigned to a project when it is created.

<ABT table name> is the **Table Name** that you specify for an ABT when you create the ABT.

For example, for a project with project ID 1000008 and ABT table name ACC_RTN_ABT, the location of the log files (on a Windows computer) can be:

C:/SAS/Config/Lev1/AppData/SASCustAnalyticsCommServer/5.3/project/1000008/build_modeling_abt/ACC_RTN_ABT/log

The following table lists the various log files and the details that the log files contain:

Table A4.1 *Log Files: Modeling ABT Building Process*

Log File	Description
build_mdl_abt_wrapper_<ABT_SK>.log	ABT_SK is a unique number that is automatically assigned to an ABT when it is created. This log file contains the logs for the initial and the final steps in the modeling ABT building process. These steps include reading the ABT metadata from the application mart, transforming it into the required format, and then storing it in the staging area.
masterloop_<ABT_SK>.log	Contains the logs that are generated during the process when the master loop job invokes the other associated jobs for execution.

For a standard ABT, the **<Project path>/<Project ID>/build_modeling_abt/<ABT table name>/log** folder also contain a folder corresponding to each ABT build date. For example, if you built the ABT as of **31 August 2010**, there is a folder named 31082010. A folder corresponding to an ABT build date contains a log file for each job that is invoked and run during the ABT building process as of that date.

For a stacked ABT, the **<Project path>/<Project ID>/build_modeling_abt/<ABT table name>/log** folder contains a folder for each stacked date specified during an ABT building process. A folder corresponding to a stacked date contains a log file for each job that is invoked and run during the ABT building process as of that date.

Troubleshooting the Scoring Process

This section describes the locations of the log files and the temporary data sets that are generated during the scoring processes (that is, during the execution of scoring jobs). It also gives details about the various log files. If the execution of a scoring job fails, you can use the information in these log files to troubleshoot the problem. Search these files for text containing ERROR or WARNING to get to the preliminary analysis of the problem.

The temporary data sets that are created while creating a scoring ABT are stored in the following folder:

<Scoring path>/<Scoring template ID>/scoring_run/scratch

The log files that are generated while creating a scoring ABT are stored in the following folder:

<Scoring path>/<Scoring template ID>/scoring_run/log

<Scoring path> represents the repository for the Scoring workspace. For example, on a Windows computer, this location can be **C:/SAS/Config/Lev1/AppData/SASCustAnalyticsCommServer/5.3/scoring**. In this location, there is a folder for each scoring template, with the corresponding scoring template ID as the folder name. This folder contains the temporary data sets and the logs for all the scoring activities corresponding to that scoring template.

The scoring path is defined in the cacsrvc.dabt.scoringpath property of the Cust Analytics Comm Server 5.3 software component. For details, see "Cust Analytics Comm Svr 5.3" on page 63.

<Scoring template ID> is a unique number (SCORING_TEMPLATE_SK) that is automatically assigned to a scoring template when it is created.

For example, for a scoring template with scoring template ID 10000005, the location of the log files (on a Windows computer) can be:

C:/SAS/Config/Lev1/AppData

/SASCustAnalyticsCommServer/5.3/scoring/

10000005/scoring_run/log.

The following table lists the various log files and the details that the log files contain:

Table A4.2 *Log Files: Scoring Process*

Log File	Description
scoring_run_job_wrapper_<SCORING_TEMPLATE_SK>.log	Contains the logs for the initial and the final steps in the scoring ABT building process. These steps include the following: • Read the input mart load date and the last scoring date (the date as of which scoring was last performed). • Import the required metadata from the scoring and actual calculation jobs. • Initiate the ABT building process. • Apply scoring code on the scoring ABT. • Write the scoring results to the analytical results area.
masterloop_<SCORING_TEMPLATE_SK>.log	Contains the logs that are generated during the process when the master loop job invokes the other associated jobs for execution.

The **`<Scoring path>/<Scoring template ID>/scoring_run/log`** folder also contains a folder corresponding to each scoring date. For example, for a scoring date of **30 September 2010**, there is a folder named 30092010. This folder contains a log file for each job that is invoked and run during the scoring ABT building process.

Troubleshooting the Nonexistence of the APDM Library

Overview

If you are unable to see the APDM library in the SAS environment, you have to add the LIBNAME statements in the autoexec files.

Add Remote Engine LIBNAME Statement in the SASApp Autoexec File

To add remote engine LIBNAME statement in the SASApp Autoexec file, complete these steps:

1. Open SAS Management Console and connect to a suitable profile.
2. On the **Plug-ins** tab, expand **Environment Management** ⇨ **Data Library Manager** ⇨ **Libraries**.
3. Right-click the **cs_apdm_remote** library and select **Properties**.
4. On the **Options** tab, click **Advanced Options**.
5. Deselect the **Library is Pre-assigned** check box.
6. Click **OK** to close the Advanced Options window.
7. Click **OK** to close the Properties window.

8. Right-click the **cs_apdm_remote** library and select **Display LIBNAME statement**.
9. Copy the LIBNAME statement and add this statement in the appserver_autoexec_usermods.sas file, which located in the **<SAS configuration directory/Lev1/SASApp** folder.

   The LIBNAME statement should appear as mentioned below:

   ```
   LIBNAME apdm REMOTE  HOSTNAME="<Host name of the SAS Share server>"

   Server=__8551 slibref=apdm  USER="<User ID>"  PASSWORD="<Password>";
   ```

10. Save the file.

Add Base LIBNAME Statement in the SAS Share Server Autoexec File

To add Base LIBNAME statement in the SAS Share Server file, complete these steps:

1. Open SAS Management Console and connect to a suitable profile.
2. On the **Plug-ins** tab, expand **Environment Management** ⇨ **Data Library Manager** ⇨ **Libraries**.
3. Right-click the **cs_apdm_base** library and select **Properties**.
4. On the **Options** tab, click **Advanced Options**.
5. Deselect the **Library is Pre-assigned** check box.
6. Click **OK** to close the Advanced Options window.
7. Click **OK** to close the Properties window.
8. Right-click the **cs_apdm_base** library and select **Display LIBNAME statement**.
9. Copy the LIBNAME statement and add this statement in the autoexec_usermods.sas file, which located in the **<SAS configuration directory/Lev1/ ShareServer** folder.

   The LIBNAME statement should appear as mentioned below:

   ```
   LIBNAME APDM '<SAS configuration directory>/Lev1/AppData

   /SASCommunicationsCommonSvr/5.3/data/apdm';
   ```

10. Save the file.

Restart Servers

After you add the LIBNAME statements, perform the following steps:

1. Restart the object spawner.
2. Go to **<SAS configuration directory>/Leve1/SASApp** folder.
3. Depending on whether the operating system is UNIX or Windows, execute the sas.sh or sas.bat file.
4. In the SAS Editor, verify that the APDM library is assigned automatically and also verify that it appears in the list of libraries that is displayed in the SAS Explorer.
5. Restart the SAS server that you have installed for SAS Customer Analytics for Communications.

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. Short form: ABT.

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

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

churn
the process of losing active customers and their related revenue. Churn can be classified as either voluntary or involuntary, depending on the reason for discontinuing the subscription or service.

churn score
a process that uses analytical data and process models to predict the likelihood of customer churn. The churn models are developed based on data from account, client, household, subscription, and equipment information. The

churner
a subscriber that involuntarily or voluntarily disconnects a subscription.

cube
See OLAP cube

data mart
a subset of the data in a data warehouse. A data mart is optimized for a specific set of users who need a particular set of queries and reports.

dimension
a data element that categorizes values in a data set into non-overlapping categories that can be used to group, filter, and label the data in meaningful ways. Hierarchies within a dimension typically represent different groupings of information that pertains to a single concept. For example, a Time dimension might consist of two hierarchies: (1) Year, Month, and Date, and (2) Year, Week, and Day.

ETL
See extract, transform, load

ETL job
a set of instructions that is used to specify ETL processes that are needed to create output.

extract, transform, load
a data warehousing process in which data is extracted from outside sources, transformed according to operational and quality needs, and loaded into a target database.

fact table
the central table in a star schema or snowflake schema. The fact table contains the individual facts that are being stored in the database as well as the keys that connect each fact to the appropriate value in each dimension.

OLAP
See online analytical processing

OLAP cube
a logical set of data that is organized and structured in a hierarchical, multidimensional arrangement to enable quick analysis of data. A cube includes measures, and it can have numerous dimensions and levels of data.

online analytical processing
a software technology that enables users to dynamically analyze data that is stored in multidimensional database tables (cubes).

Index

A

architecture components 4
  client tier 6
  data tier 5
  middle tier 6
  server tier 5
architecture overview
  client tier 4
  data tier 3
  middle tier 4
  server tier 3

B

build-cap date 52

C

client tier
  components of 6
  overview of 4
configuration 40
  columns 43
  dimensional attribute values 43
  implicit subset criteria 52
  implicit variables 54
  libraries 40
  project owners 62
  source tables 42
  subject groups 49
  subset maps 43
  time periods 41
configuration job 32

D

data flow 7
data tier
  components of 5
  overview of 3

E

ETL jobs
  analytics jobs 27
  configuration job 32
  loading order 89
  writeback jobs 32

I

implicit subset criteria 52
  configuring 53
  creating 53
implicit variables 54
  configuring 55
  creating 55
installation 13

L

loading order 89
  bill-monthly jobs 89
  weekly jobs 90
  writeback jobs 91

M

master loop job 20
metadata layout 15
middle tier
  components of 6
  overview of 4
middle-tier 59
  logging 66
  software compoments 63

O

outcome-based filtering 51

P

parameters 77
post-action macros 56
  configuring 58
  creating 57
post-installation 17
post-installation tasks
  configuring parameters 20
  defining users and assigning roles 20
  deploying master loop jobs 20
  overview of 17
  verifying users and granting permissions 19
pre-installation 11
predefined libraries 17
project owners, configuring 62

R

roles and capabilities 59

S

SAS Customer Analytics for Communications
  data flow 7
  solution flow 8
scoring
  deleting results 71
  ETL job 68
  process 67
  score aggregation 70
server tier
  components of 5
  overview of 3
solution flow 8
subject groups
  configuring 51
  creating 50
subset maps 43
  configuring 45
  creating 44
  predefined 83
system requirements 11

T

troubleshooting
  APDM library 96
  modeling ABT building process 93
  scoring process 95

U

unconfiguration tasks 22

V

verification 15

W

writeback jobs 32