THE POWER TO KNOW.

SAS® Service Parts Optimization 4.2

Administrator's Guide

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2009. *SAS® Service Parts Optimization 4.2: Administrator's Guide*. Cary, NC: SAS Institute Inc.

SAS® Service Parts Optimization 4.2: Administrator's Guide

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

1st electronic book, December 2009

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

What's New in SAS Service Parts Optimization 4.2

Overview

SAS Service Parts Optimization 4.2 has the following enhanced features:

- new workspace-based user interface with better interactivity
- improved process flow
- enhanced analytical capabilities
- documentation enhancements

New Workspace-Based User Interface

Overview

The user interface of SAS Service Parts Optimization contains four workspaces that are dedicated to analyze inventory status, plan orders, develop scenarios, and manage forecast results.

The improved user interface enables you to perform the following common functions on the data table in each workspace:

- filter the contents
- search for a specific argument and filter the contents
- customize the columns
- export data from a table to a Microsoft Excel file or a comma-separated values (CSV) file

Inventory Analysis Workspace

The new and enhanced features of the Inventory Analysis workspace enable you to perform the following activities:

- generate reports and alerts
- identify issues with inventory metrics
- identify the root cause of inventory issues by drilling down to the facility and item pair

- view period reports for the metrics

Order Suggestions Workspace

The new and enhanced features of the Order Suggestions workspace enable you to perform the following activities:

- view order information
  - view the recommended order quantity from the primary source
  - view lateral transshipment suggestions
  - view the replenishment plan details
  - prepare the orders to be exported to the Enterprise Resource Planning (ERP) system
  - view attributes for facility and item pair
- manage order plans
  - lock orders
  - view projected service levels
  - view excess, normal, or shortage inventory of items
  - view the total transshipment cost and quantity

Scenario Development Workspace

The new and enhanced features of the Scenario Development workspace enable you to perform the following activities:

- create data selection
  - create data selections for items, facilities, and networks
  - view predefined attributes for items and facilities
- manage scenario projects
  - create four types of scenario projects:
    - service level sensitivity analysis
    - customer-facing facility analysis
    - internal facility service level analysis
    - ad hoc analysis
  - view scenario results
  - view scenario settings
  - edit scenario settings
  - copy a scenario
  - re-optimize a scenario
  - update results of a scenario in a master list (promote scenario settings) to make the scenario permanent

Forecast Management Workspace

The new and enhanced features of the Forecast Management workspace enable you to perform the following activities:

- manage data driven business rules and customization capabilities
  - define forecast hierarchy in control tables
  - input batch parameters through control tables
  - change parameters that control the batch process
  - restore parameters that control the batch process to their pre-configured values
- review results of the forecasting process
  - view the status of forecasting and inventory optimization batch process
  - view results of the forecasting process
  - identify time series that need manual intervention
  - view forecasted and actual demand details for all the facility and item pairs within a forecast group for the available period
  - view forecasted and actual demand details for all time series of a selected facility and item pair
  - modify the parameters that control the forecasting batch process
  - input external forecast for items with no or insufficient history
  - view lifetime demand for a facility and item pair

Improved Process Flow

Batch Mode

The batch mode of SAS Service Parts Optimization contains two batch processes: Forecasting and Inventory Optimization.

The new or enhanced options in the forecasting batch process perform the following tasks in an automated way:

- associate a facility and item pair with a forecast group
- set up regression variables and attach independent variables to the forecast input tables
- create and manage input data that is needed by the forecasting process
- categorize all time series based on the forecast errors
- control the automatic forecast correction process
- specify automatic model selection and scoring
- forecast successor items
- add models that are created in SAS Forecast Studio in the forecasting batch process

- store time series with lower accuracy forecasts as an input data set for SAS Forecast Studio
- use the same nomenclature for the input data in SAS Forecast Studio as is used for the forecasting batch process

The new or enhanced options in the inventory optimization batch process perform the following tasks in an automated way:

- select facility and item pairs for optimization for the specified base period
- suggest optimized order quantities from primary and alternative sources
- calculate inventory metrics

User-Interactive Mode

The user-interactive mode of SAS Service Parts Optimization contains four workspaces: Inventory Analysis, Order Suggestions, Scenario Development, and Forecast Management. For details, see "New Workspace-Based User Interface" on page v.

Enhanced Analytical Capabilities

SAS Service Parts Optimization provides the following features:

- integration with SAS Forecast Server to correct forecast results with lower accuracy
- ascertain the optimum service level at customer-facing locations to meet budget constraints
- assess the impact of input parameters such as service level, lead time, demand, and unit cost on inventory cost
- rebalance the inventory across the distribution network

Documentation Enhancements

SAS Service Parts Optimization 4.2 documentation is restructured. The following deliverables are available with the solution:

Online Help
: contains descriptions for the graphical user interface (GUI) elements in the view or dialog box that is currently active. The Help is available with the application and can be accessed by pressing the **F1** key on your keyboard.

User's Guide
: describes the core functionality of SAS Service Parts Optimization and how to work in the user interface to manage forecast results, analyze inventory status, develop scenarios, and plan orders.

Administrator's Guide
: explains how to install the solution and perform the post-installation tasks, and describes the data and system administration tasks.

Data Dictionary
: describes the dimension tables, user interface tables, analytical base tables (ABT), solution data layer (SDL) tables, and also their columns.

Data Models
: contains the logical and physical data models of SAS Service Parts Optimization.

Recommended Reading

- *SAS Forecast Server: Administrator's Guide*
- *SAS Data Integration Studio: User's Guide*
- *SAS Management Console: User's 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 at:

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513
Telephone: 1-800-727-3228
Fax: 1-919-531-9439
E-mail: sasbook@sas.com
Web address: support.sas.com/bookstore

Chapter 1

Introduction to SAS Service Parts Optimization

Welcome to SAS Service Parts Optimization

What is SAS Service Parts Optimization?

SAS Service Parts Optimization enables you to forecast demand for service parts based on historical consumption data. You can plan inventory and orders to fulfill the forecasted demand, and manage the introduction of new parts and new product launches into the service chain.

SAS Service Parts Optimization is a Java client application. The dynamic user interface of the solution caters to the needs of different users within an organization. You can view, review, interpret, and analyze results of the forecasting and inventory optimization processes. You can also perform what-if scenario analysis, review suggested order quantities, and then submit them to the enterprise resource planning (ERP) systems.

Functional Components of SAS Service Parts Optimization

SAS Service Parts Optimization includes the following main functional components:

data warehousing
: consists of the ongoing ETL jobs that update the data warehouse with changes from the source system.

analytics
: consists of the ETL jobs for the forecasting and inventory optimization processes.

  SAS Forecast Studio that is integrated with SAS Service Parts Optimization enables you to perform advanced analytical functions.

Java applications
: consists of the SAS Service Parts Optimization middle tier and client tier.

These components are seamlessly integrated to offer a true end-to-end solution. The SAS Service Parts Optimization solution takes data from various source systems and applies appropriate analytical methods to provide information for advanced business decisions.

How Does SAS Service Parts Optimization Help You?

SAS Service Parts Optimization provides advanced analytical capabilities for service organizations. Major uses of SAS Service Parts Optimization include the following:

- provide demand and inventory forecasts
- provide input to set inventory levels
- provide input to plan replenishment activities
- perform what-if analysis

SAS Service Parts Optimization provides automation and analytical sophistication to the forecasting and inventory optimization processes. You can generate millions of forecasts in a reasonable turnaround time. Also, you can calculate inventory replenishment policies for parts distribution systems and thereby maintain adequate stock levels. These capabilities enable you to reduce costs, increase revenues, and improve customer satisfaction and brand image.

Accessibility Features of SAS Service Parts Optimization

About the Accessibility Features

SAS Service Parts Optimization includes the following accessibility and compatibility features that improve usability of the product for users with disabilities. These features are related to accessibility standards for electronic information technology adopted by the U.S. Government under Section 508 of the U.S. Rehabilitation Act of 1973, as amended.

If you have questions or concerns about the accessibility of SAS products, send e-mail to **accessibility@sas.com**.

SAS Service Parts Optimization conforms to accessibility standards for the Windows platform, with the following exceptions:

- The dialog boxes fail to adjust in size to accommodate any increase in text size.
- There is insufficient support for high contrast and custom color styles in SAS Service Parts Optimization.
- There is insufficient support for column navigation using keyboard in the tables.
- When a screen reader is used, the following is true:
  - The screen reader cannot read the tables.
  - The screen reader cannot read the toolbar in the Suggested Orders dialog box.
  - The screen reader cannot read a child dialog box that is opened from the New/Edit/Copy Scenario dialog boxes for service level sensitivity analysis scenario type.
  - The screen reader cannot read some labels in the Edit Parameters dialog box.
  - The screen reader reads incorrect mnemonics for the suboptions of the **Open Demand Projection Details** option from **Menu** of the Forecast Results view.

Keyboard Navigation

SAS Service Parts Optimization can be navigated by using the keyboard. The following table includes some guidelines:

Table 1.1 *Standard Keyboard Navigation Controls*

Task	Keyboard Control
Displays the Inventory Analysis workspace	CTRL + 1
Displays the Order Suggestions workspace	CTRL + 2
Displays the Scenario Development workspace	CTRL + 3
Displays the Forecast Management workspace	CTRL + 4
Opens Help	CTRL + F1/F1

Task	Keyboard Control
Displays a list of the open views and opens the previous view	CTRL + F7
Opens **Menu** for the active view	CTRL + F10
Detaches the view in a new window	ALT + SHIFT + W
Moves forward through controls	TAB
Moves backward through controls	SHIFT+TAB
Performs the action that corresponds to the active control	ENTER

You can access a main menu option by pressing the ALT key and the underlined letter that is shown on the menu. Some menu options in the main menu and in the **Menu** of a view have keyboard shortcut keys assigned to them. If an option has a shortcut key, then the key is listed on the menu next to the option.

Using the Documentation

Purpose

The Administrator's Guide describes the deployment, and the system and data administration tasks for the SAS Service Parts Optimization solution. This guide provides information about the following topics:

- system architecture
- installation and post-installation tasks
- data architecture and data flows
- initial and incremental data loading tasks
- system administration tasks
- troubleshooting instructions

Required Skill Sets

To install, configure, administer, and use the SAS Service Parts Optimization solution, the following individuals with the necessary skill sets are required for each particular piece of administration and use:

System Administrator
: must have the skills to perform installation and configuration of the SAS Intelligence Platform and the SAS Service Parts Optimization solution on the required operating system. For example, to install the SAS Intelligence Platform on the Windows operating system, the administrator must have the following skills:

  - be an administrator of the computer

- be familiar with concepts of the operating system
- know how to create folders
- know how to run DOS .bat files
- be familiar with application server concepts

The system administrator must use the SAS Management Console software to maintain the metadata for servers, users, and other global resources that are required by the solution.

Data Administrator
: must maintain the solution's data and perform other solution administration to enable users to analyze data. The administrator must be able to create ETL processes and programs that are used by the solution.

SAS Service Parts Optimization Solution User
: must understand the data to be analyzed, the requirements for analysis, and the results of data analyses.

Organization of this Guide

The Administrator's Guide provides an overview of the solution and its architecture, and explains the steps to set up and manage the solution. These steps can be categorized under the following high-level tasks:

- installation and post-installation
- data administration and management
- system administration

This guide is organized as follows:

Introduction
: introduces you to your SAS solution and explains how the guide is organized and presented. Also, provides additional resources for you to explore if you need more information about your SAS solution.

Installation and Post-Installation
: provides the steps to install the SAS Service Parts Optimization solution and also provides the post-installation tasks.

Data Administration and Management
: provides the data administering tasks that you must perform to load data for the first time and also covers the ongoing data tasks.

System Administration and Maintenance
: discusses the necessary system administration and maintenance tasks such as administering users.

Appendixes
: contains additional information that is referenced from within sections of the documentation.

SAS Service Parts Optimization Architecture

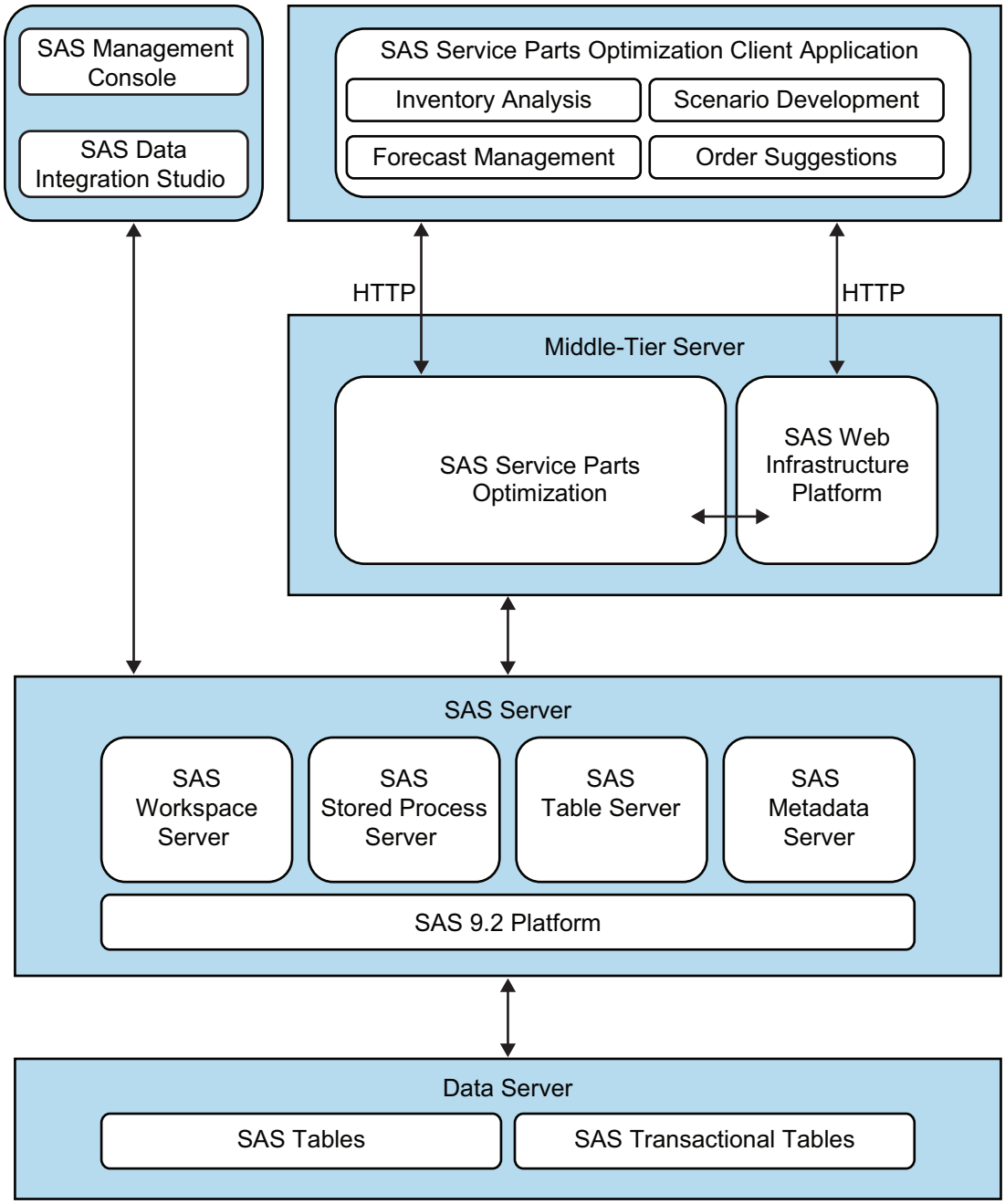
Overview of the Solution Architecture

SAS Service Parts Optimization has an N-tier architecture that consists of the following tiers:

- client
- middle
- server
- data

The following figure shows the generic division of responsibilities across the different tiers.

Figure 1.1 *SAS Service Parts Optimization Architecture*



The subsequent sections explain the purpose and components of each tier.

Client Tier

Overview

The client tier is responsible for user interaction, navigation flows, preparation of a call for user choice, and displaying of computational results.

Client-Tier Components

SAS Service Parts Optimization requires the following applications:

SAS Management Console
enables you to perform metadata administration activities.

SAS Data Integration Studio
provides the ability to manage ETL jobs for the extraction, transformation, and loading of data.

The SAS Service Parts Optimization client displays computational results and enables user interaction. The client-side user interface is developed using the rich client platform (RCP), which is an eclipse plug-in-based development platform. The client contains five plug-ins: a separate one for each of the four workspaces and one plug-in that is defined for the main application. The following table displays details about the different workspaces and their corresponding plug-ins.

Table 1.2 *Plug-ins in SAS Service Parts Optimization*

Plug-in	Workspace	Use
sas.spo.demandplanning	Forecast Management	Plug-in for Forecast Management workspace and its associated activities.
sas.spo.inventory	Inventory Analysis	Plug-in for Inventory Analysis workspace and its associated activities.
sas.spo.scenarios	Scenario Development	Plug-in for Scenario Development workspace and its associated activities.
sas.spo.orders	Order Suggestions	Plug-in for Order Suggestions workspace and its associated activities.
sas.spo.client	Main Application	The main application plug-in that is common for all the plug-ins. It serves the common functionality for all the workspaces.

SAS Forecast Studio is bundled with SAS Service Parts Optimization to enable you to perform advanced analytical functions on the forecast results with low accuracy. For more information about SAS Forecast Studio, see *SAS Forecast Studio: User's Guide* and *SAS Forecast Studio: Administrator's Guide*.

Middle Tier

Overview

The middle-tier component is responsible for intercepting the client request and providing an execution environment to the client. The middle tier is developed over SAS business intelligence (BI) middle-tier technologies such as Web Infrastructure Platform (WIP) and Platform Services. These technologies provide the middle-tier application with the ability to run requests over SAS servers.

Middle-Tier Components

SAS Service Parts Optimization requires the following middle-tier components:

SAS Service Parts Optimization Services
: take adequate server-side actions based on client-side user actions.

Web Infrastructure Platform (WIP) Services
: are used for standard operations such as configuration, authentication, e-mail, and so on.

  Middle-tier components use the HTTP-based protocol to communicate with the following WIP-based services:

  - Authentication Service
  - User Info Service
  - Mail Service

The client tier communicates with the middle tier by using HTTP-based communication model. The client messages are transferred over the wire using Spring HTTP invoker.

Server Tier

Overview

The server tier hosts SAS application servers that run the requests made by clients. Topological support might include support for non-functional requirements such as fail-over, pooling, and load balancing.

Server-Tier Components

SAS Service Parts Optimization requires the following server-tier components:

SAS Workspace Server
: uses SAS code execution for a number of clients at a time.

SAS Stored Process Server
: runs a registered stored process on behalf of the client.

SAS Table Server
: accesses SAS transactional database and other Base SAS data sets for better performance.

SAS Metadata Server
: sets up various configurations such as stored processes, workspace server, stored process server, table server DSNs, and so on.

Data Tier

Overview

Data tier consists of the following data storage types:

SAS tables
: are stored in Base SAS formats under different libraries. For more information about the libraries, see Library Descriptions on page 20.

SAS transactional tables
: are stored in the SAS Transactional Database. The tables contain transactional and application data.

SAS Service Parts Optimization Modes of Operations

You can operate SAS Service Parts Optimization in two modes, online mode and batch mode.

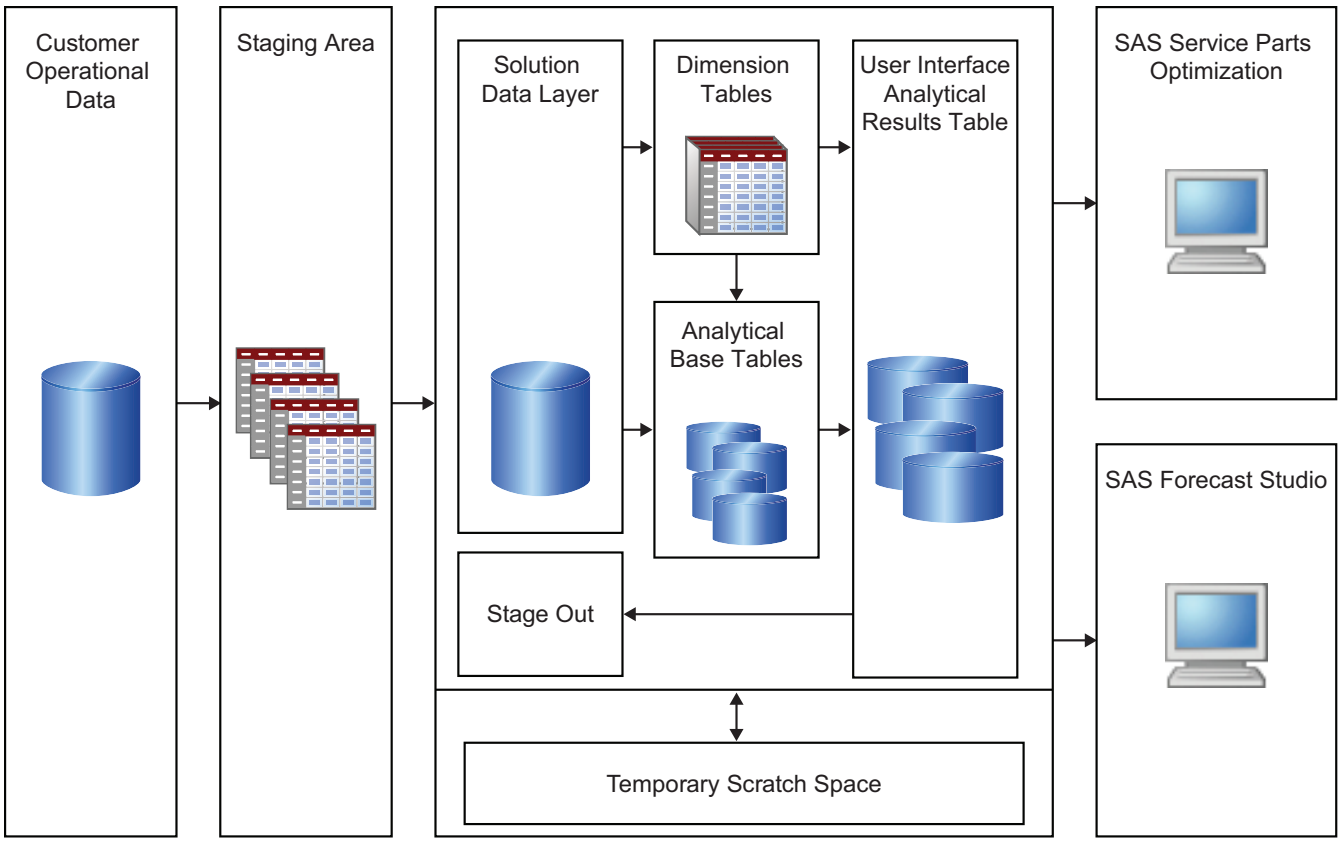
- In online mode, the middle tier, server tier, and data tier are running and you can use and work with the SAS Service Parts Optimization application.
- In batch mode, the ETL jobs to load, forecast, optimize, and report on data are running. The SAS Service Parts Optimization middle tier is shutdown and you cannot work with the client application. However, the SAS system uses ETL jobs to update and maintain the data.

  Note: A few ETL jobs update the UIART library. Ensure that the period of the job run is scheduled for a maintenance period that does not affect request processing.

Back-End Data Flow for Loading and Accessing Data

The following figure shows the general flow for the back-end data.

Figure 1.2 *Back-End Data Flow*



The general data flow works as follows:

1. Source data is stored at the data source layer in the customer operational data system.
2. Tables in the staging area are populated from the source data.

3. The solution data layer (SDL) is loaded with the data from the staging area. The recommended method to load data into the SDL is as follows:

   a. Identify data structures in the SDL that need to be loaded to meet your business needs.

   b. Identify and locate source systems for the data by mapping the SDL data and the source system data.

   c. Extract, cleanse, and transform the source system data and consolidate the data into the staging area that is designed to load data into the SDL. The data in the staging area is validated to ensure that all interdependencies or relationships in the SDL are defined.

   d. Load lookup (reference) tables that store code values that are required in the SDL.

   e. Load data into the SDL in accordance with a specific sequence. For more information about the sequence, see "Load All Solution Data Layer Tables " on page 21.

   Note: All the earlier steps are site-specific. After the SDL is loaded, SAS Service Parts Optimization provides ETL jobs that use the SDL data for further processing. Initial jobs are run to load the configuration tables and to set up the environment for running the next set of ETL jobs, which are the ongoing jobs. One of the configuration table stores the base period — the time between two replenishment orders that can be placed. The base period is customizable and you can specify whether the base period must be week, month, or quarter.

4. The ongoing jobs are run to load or update the SAS Service Parts Optimization tables in accordance with the base period. The jobs perform the following main tasks:

   a. Load dimension tables with the latest attributes from the SDL.

   b. Run back-end processes for forecasting and inventory optimization to load the analytical base tables.

   c. Load the application tables (also called as user interface analytical result tables - UIARTs).

   d. Load the tables in the Stageout library to store the results of the inventory optimization process. These results can be updated in the source system.

   Note: The temporary library Scratch is used as an intermittent storage area.

5. After the SAS Service Parts Optimization tables are loaded, the latest data is displayed on the interface of the SAS Service Parts Optimization application.

   You can refine a set of forecast results from SAS Service Parts Optimization by using the SAS Forecast Studio client. The results can be later viewed through the SAS Service Parts Optimization user interface.

   For more information about the data management tasks, see Chapter 3, "Data Administration and Management of SAS Service Parts Optimization," on page 19.

SAS Service Parts Optimization Clients

Overview

SAS Service Parts Optimization manages a separate session for each user who logs on. The options that are available on the user interface depend on the profile with which you log on.

SAS Forecast Studio is integrated with SAS Service Parts Optimization and enables you to perform advanced analytical processes on a set of forecast results. The outcome of the processes can be further viewed through the SAS Service Parts Optimization user interface. The following sections describe these client applications and their purpose.

Access to the clients is controlled through security settings that are set up for the users. For more information, see "Administer Users" on page 42.

SAS Service Parts Optimization Rich Client

The SAS Service Parts Optimization rich client enables you to manage forecasts, analyze inventory, develop scenarios, and suggest orders.

The client is installed on a computer by running a client installation program. The client starts a session of its own. All tasks are performed under this session. The session ends when you exit the application or when the middle tier shuts down.

SAS Forecast Studio Client

SAS Forecast Studio provides a graphical interface to the forecasting and time series analysis procedures. The SAS Forecast Studio client provides a user-friendly graphical interface to view forecasts. You can view forecasts that reflect the real scenarios of the business. You can also improve the forecasting performance across all products and locations, at any level of aggregation.

Related SAS Software

Overview

Many features that are not found in SAS Service Parts Optimization are available in other SAS solutions or in SAS products that are used with this SAS solution. If you do not find a feature that you need in this software, you might find it in one of the following SAS solutions or products.

SAS Forecast Studio

SAS Forecast Studio is a client application that provides market-driven planning through accurate demand forecasting. Given a time-stamped data set, SAS Forecast Studio accumulates time-stamped data. SAS Forecast Studio then performs the following tasks:

1. forms a fixed-interval time series

2. diagnoses the time series
3. creates a list of candidate model specifications
4. fits each candidate model specification to your time series
5. generates forecasts
6. selects the most appropriate model specification
7. creates a forecast score
8. generates forecasts from the forecast score
9. evaluates the forecast

In short, SAS Forecast Studio automatically generates large volumes of reliable forecasts to support your planning and business decisions.

SAS Inventory Optimization

SAS Inventory Optimization enables you to manage your inventory levels while fulfilling the customer service levels by providing optimal inventory policies and recommending when and how much to order.

SAS Inventory Optimization enables you to create what-if scenarios by using different parameters or forecasts, and to determine the impact on policies and projected customer service levels and costs.

SAS Enterprise Data Integration Server

SAS Enterprise Data Integration Server is an application that enables you to manage extract, transform, and load (ETL) process flows. These tasks are sequences of steps for the extraction, transformation, and loading of data. SAS Enterprise Data Integration Server enables you to do the following:

- specify metadata for sources, such as tables in an operational system
- specify metadata for targets, such as tables and other data stores in a data warehouse
- create jobs that specify how data is extracted, transformed, and loaded from a source to a target

Where to Go for More Information

Online Help

For information about how to operate your software, select **Help** ⇨ **Help Contents** from within the application.

For information about the version of the software that you are running, select **Help** ⇨ **About SAS Service Parts Optimization** from within the application.

SAS Technical Support Services

As with all SAS products, the SAS Technical Support staff is available to respond to problems and answer technical questions about SAS Service Parts Optimization.

Chapter 2

Installation and Post-Installation Tasks for SAS Service Parts Optimization

Default File Locations

The default physical locations that are described in this guide are created during the standard configuration process. The SAS Consulting services representatives working at your site can change these locations when they customize your setup to meet your requirements. These representatives can then provide documentation of the locations for your site.

A standard file system location that is referred to in this guide is the SAS configuration directory (written as <SASCONFIG>). <SASCONFIG> refers to the directory path where the SAS configuration process writes the finalized software image.

Different files are written to this location. In a Windows operating environment, the default location for <SASCONFIG> is as follows: **`C:\SAS\<ImageName>`**, where <ImageName> represents the site-specific name for your software bundle as defined in the SAS plan file that is part of your order. For UNIX, this location can vary and your SAS Consulting services representative can help you to determine the default file location for your site.

SAS Intelligence Platform Documentation

Because SAS Service Parts Optimization is designed to work with the SAS Intelligence Platform, it is strongly recommended that you read the SAS Intelligence Platform documentation, which is located at **`http://support.sas.com/documentation/onlinedoc`**, before you install SAS Service Parts Optimization. That documentation provides pre-installation tasks and instructions to guide you through a typical installation of the SAS Intelligence Platform.

Deployment Outline

To deploy SAS Service Parts Optimization:

1. Work with your SAS representative to develop a plan and designate the host computers for your deployment of SAS Service Parts Optimization.

   Note: While you create the software order, ensure that you include the following components in the order:

   - Service Parts Optimization Server
   - Data Integration Server
   - Enterprise Business Intelligence Server

2. Install the SAS Service Parts Optimization on each host computer as outlined in your plan.xml file. During installation, ensure the following:

   - Specify the e-mail address that can be used by the SAS Service Parts Optimization application to send e-mails.
   - SAS Service Parts Optimization environment can connect to any SAS environment where the SAS Service Parts Optimization middle tier is deployed. SAS environments are defined in the sas-environment.xml file.

     Specify the path to the sas-environment.xml file, that is, provide the following URL: **`http://host:port/SASLogon/sas-environment.xml`**. This URL is also available in the Notices section of the instructions.html file for your middle-tier installation. Contact your administrator for the host name and port number details, or alternate URL, to be used.

3. Complete the post-installation tasks. For more information, see "Post-Installation Tasks" on page 16.

Post-Installation Tasks

Install Current Hot Fix for SAS Inventory Optimization

You must install the current hot fix A39001 for SAS Inventory Optimization 2.1. The hot fix includes some defect fixes for the MIRP procedure that is used in SAS Service Parts Optimization.

The required hot fix can be downloaded from the following Web address:

`http://ftp.sas.com/techsup/download/hotfix/HF2/A39.html#A39001`

Update the sasbatch.bat File

To run the ETL jobs in a batch by using the SAS Batch Server, update the **`sasbatch.bat`** file by removing the **`noxcmd`** option.

To do so:

1. Browse to the SAS Batch Server folder. By default, the path is **<SASCONFIG>\Lev1\SASApp\BatchServer**.
2. Open the **sasbatch.bat** file in edit mode.
3. In the file, search for the text **–noxcmd** and comment out this text.

   For example, consider the following text:

   ```
   CMD_OPTIONS="-noxcmd -lrecl 32767"
   ```

   In Windows, you can comment it out as follows:

   ```
   rem CMD_OPTIONS="-noxcmd -lrecl 32767"
   ```

   In UNIX, the commented text is as follows:

   ```
   # CMD_OPTIONS="-noxcmd -lrecl 32767"
   ```

Update the SAS Scripts

On a UNIX operating environment, you need to update certain SAS script files to ensure that the tables that are created by the SAS Workspace Server and the SAS Stored Process Server have the necessary write permissions.

Specify the umask setting of 002 in the following SAS scripts:

- <SASCONFIG>/Lev1/SASApp/BatchServer/sasbatch_usermods.sh
- <SASCONFIG>/Lev1/SASApp/PooledWorkspaceServer/PooledWorkspaceServer_usermods.sh
- <SASCONFIG>/Lev1/SASApp/StoredProcessServer/StoredProcessServer_usermods.sh
- <SASCONFIG>/Lev1/SASApp/WorkspaceServer/WorkspaceServer_usermods.sh

Chapter 3

Data Administration and Management of SAS Service Parts Optimization

Overview

After you complete the post-installation tasks, the SAS Service Parts Optimization solution is ready for use. You can now perform the following data-related tasks:

1. Specify the base period.
2. Perform the pre-initial data load tasks.
3. Perform the ongoing data tasks.

For more information about the general data flow, see "Back-End Data Flow for Loading and Accessing Data" on page 10.

Setting Up the Base Period

The base period for the SAS Service Parts Optimization solution must match the planning frequency of your organization. The frequency could be weekly, monthly, or quarterly. SAS Service Parts Optimization supports one planning frequency or base period.

You can specify the forecasting base period as week, month, or quarter in the SDL table Forecast_Group.Time_Hierarchy_Level_No. For inventory optimization, you must specify the base period in the GL_BASE_PERIOD parameter of the global parameter table.

The default base period for the inventory optimization process is set to week. The tables that are used by the inventory optimization process use WK in their name and the physical name fields to signify that the base period is week. For example, the table IO_WK_Arc_Data_ABT contains WK. If you modify the base period to either month or quarter, then you must modify WK in the name and the physical name of the tables to MTH or QTR respectively. For example, if you specify month as the base period, then the table name IO_WK_Arc_Data_ABT must be changed to IO_MTH_Arc_Data_ABT. All the tables that need to be changed are available in the following location: **`/Products/SAS Service Parts Optimization/Service Parts Optimization 4.2/SPO_TABLES/ABT_TABLES/IO`**.

Note: For better accuracy, specify the same base period for the forecasting and the inventory optimization processes.

Pre-Initial Data Load Tasks

Overview

Before you load initial data in the SAS Service Parts Optimization 4.2 solution, perform the following tasks:

1. Confirm that libraries are created.
2. Load all tables in the solution data layer (SDL) library.
3. Run the initial jobs.
4. Assign permissions to tables.
5. Customize the global parameter values.
6. Set or modify the job parameter values.

Confirm That Libraries are Created

After you complete the installation of the SAS Service Parts Optimization solution, ensure that all the libraries are created. The default path for the libraries is as follows:**`<SASCONFIG>/Lev1/AppData/SASServicePartsOptimization/data`**. To view the libraries through the SAS Management Console, select **Data Library Manager** ⇨ **Libraries**. Here is a list of the libraries that are created:

SDL
: stores tables that contain the source data. You must load these tables with customer-specific data.

Control
: stores tables that contain control parameters. These parameters are used to run the solution jobs. These tables are created and loaded by the initial one-time jobs. The control parameters can be customized as per customer requirements.

ABT
: stores tables that are specific to analytical solutions.

DIM
: stores dimension tables that are loaded with the latest attributes from the SDL tables.

UIART
: stores application tables that are loaded through ETL processes. These tables contain information that are required to be displayed on the user interface of the application.

TSDB
: stores tables that are loaded or updated by the user. The library is used for concurrent usage of tables from the user interface and back end. The tables in this library are not physically present, but are appended to the spodb database that is created during installation.

Forecast
: stores tables that contain time series details for facility and item pairs with low accuracy forecast results. A table is created per forecast group.

Scenario
: stores tables that contain information about the scenarios that are created.

Scratch
: stores intermediate tables that are used in various processes.

SPO_ER
: stores tables with information about user-defined events that are specific to each forecast group.

SPO_MR
: stores user-defined custom models of a catalog.

Stageout
: stores solution data that must be updated in the source system. For example, optimization policies, approved orders, and promoted settings.

Load All Solution Data Layer Tables

Set Up the Solution Data Layer

The solution data layer (SDL) tables contain source data from the source system in a format that is uniform and complete. The SDL tables ensure that the source data format facilitates their accurate import into the tables of SAS Service Parts Optimization.

Create empty SDL tables by running the **`sdl_ddl.sas script`** file that is provided as a part of the installable. For the Windows operating environment, the script is available at **`!SASROOT\spartoptmva\sasmisc`**. If you install SAS in the default folder, then the script is available at **`C:\Program Files\SAS\SASFoundation \9.2\spartoptmva\sasmisc`**.

For the UNIX operating environment, the script is available at **`!SASROOT/misc/spartoptmva`**.

Loading Sequence of Solution Data Layer

The loading of the solution data layer (SDL) must follow a sequence to ensure that all the table dependencies are taken care of. The following table provides the loading sequence of SDL tables. The sequence consists of groups. You can load the tables that belong to a group simultaneously or in any order. However, you must load all the tables of a group before loading tables from the next group, that is, the load must follow a groupwise ascending sequence.

Table 3.1 *Loading Sequence of Solution Data Layer Tables*

Group Number	Table Name
1	• Forecast_Group • Item_Category • Location • Lookup_Master • Route_Type_Ref • Network_Model
2	• Employee • Item • Lookup_Detail • Organization
3	• BOM • Customer • Facility • Item_Price • Item_Substitute • Item_Succession • Vendor
4	• Customer_Order • Facility_X_Item • Purchase_Order • Route • Vendor_Facility_Item

Group Number	Table Name
5	• Back_Order_Summary • Dispatch • Facility_Item_Demand • Facility_Item_Ind_Variable • Facility_Item_Inventory • Facility_Item_X_Network • Forecast_Group_Item_Detail • Network_X_Route • Pipeline_Inventory • Receipts • Route_X_Item

Loading of Lookup Tables

The solution data layer (SDL) library has two reference or lookup tables that store code values and their descriptions. These value and description pairs are required by the attribute tables or are used to display information on the user interface. The ETL code derives the description of the code columns from the lookup detail table. Hence, you must ensure that the required code values are present in the lookup detail tables.

The following two reference tables exist for lookup:

- Lookup master table (Lookup_Master):

  This table contains the SDL table name and the _CD column name that is present in the SDL table.

  Table 3.2 *A Sample Record in the Lookup Master Table for Facility Type Code*

TABLE_NAME	COLUMN_NAME
Facility	Facility_Type_CD

- Lookup detail table (Lookup_Detail):

  This table stores details of the code columns, that is, the code values and their descriptions.

  Table 3.3 *Sample Records in the Lookup Detail Table for Facility Type Code*

TABLE_NAME	COLUMN_NAME	CODE_VALUE	CODE_DESC
Facility	Facility_Type_CD	100	IT COMPANY
Facility	Facility_Type_CD	110	SERVICE BASED
Facility	Facility_Type_CD	120	PRODUCT BASED
Facility	Facility_Type_CD	130	CALL CENTER

The following table provides the table name and its mandatory code columns that must be loaded.

Table 3.4 *Table and Its Mandatory Code Column Names*

Table Name	Code Column Name
Facility	Facility_Type_CD
Facility_X_Item	• Policy_Type_CD • Service_Type_CD
Forecast_Group	Forecast_Group_CD
Item	• Item_Group_CD • Item_Status_CD • Item_Type_CD • Make_Or_Buy_CD
Item_Price	Price_Type_CD

Run the Initial Jobs

The initial jobs are to be run only once after you install the solution.

The initial jobs load the initial tables that are required before loading data for the solution. You can access these jobs from the following location in SAS Data Integration Studio:

`/Products/SAS Service Parts Optimization/Service Parts Optimization 4.2/CONTROL/INITIAL_ONE_TIME_JOBS`

Run the jobs in the following sequence:

1. 01_CREATE_GLOBAL_PARAMETER_LIST
2. 02_CREATE_CONTROL_TABLES
3. 03_CREATE_FCST_ABT
4. 04_CREATE_TIME_PERIOD_DATA
5. 05_CALENDAR_HIERARCHY1
6. 06_CALENDAR_HIERARCHY2
7. 07_TIME_DIM
8. 08_CREATE_TIME_VIEWS
9. 09_CREATE_USER_TABLES
10. 10_POPULATE_HPF_PREFERENCES

For more information about each job, see Appendix A2, “ETL Job Details,” on page 59.

After you load the initial jobs, the following main tasks are accomplished:

- All the global parameters are loaded.

- The control tables and user tables are created. Control tables store the controlling or key information about SAS Service Parts Optimization. The Control.Job_Status table contains the status for some of the ETL jobs. You can review the statuses of the jobs in this table to verify that the jobs ran successfully.

  User tables are the intermediate tables that are created either in the Table Server or in the UIART library. These tables are used to store information that is displayed on the user interface and received through the interface.

- The time dimension table is created that occupies an important place in every data warehouse. In SAS Service Parts Optimization, the time dimension table is populated only once before running the ongoing ETL jobs. The table supports the hierarchy calendar Day → Week → Month → Quarter → Year. You can specify the start date and the number of years for which the table is to be generated. To view the default values of the parameters that are used to populate the time dimension table, see "Job Parameter Table" on page 56. With the time dimension table, views for week, month, quarter, and year are also created, which are required for running the ETL jobs.

All these tasks ensure that the configuration of the overall environment for SAS Service Parts Optimization is complete.

Assign Permissions to Tables

You must assign the necessary access permissions to the tables TSDB.Scenario_Params and TSDB.Scenarios to enable the SAS code to use the tables. These access permissions are not automatically set.

To set access permissions:

1. In SAS Management Console, expand **Environment Management** ⇨ **Table Server Manager** ⇨ **SASTS** ⇨ **SAS Data** ⇨ **Transactional Databases** ⇨ **spodb**.
2. Double-click **SQL Data**. A list of tables is displayed in the right pane.
3. In the right pane, right-click the **SCENARIO_PARAMS** table, and from the pop-up menu, select **Properties**. The SCENARIO_PARAMS Properties dialog box opens.
4. Click the **Authorization** tab, and for the **PUBLIC** users, in the **Effective Permissions** section, under the Grant column, select the check boxes for the following permissions:
   - ReadMetadata
   - Select
   - Update
5. Repeat the steps 3 and 4 for the **SCENARIOS** table.

Customize the Global Parameter Values

Some of the ETL processes populate the analytical base tables (ABTs) and user interface analytical result tables (UIARTs). These processes require certain values for correctly extracting data from the solution data layer (SDL). All the global parameters are listed in the Control.Global_Parameter_List table.

For example, to extract the latest data from the SDL, the following condition is used in the ETL process:

```
WHERE VALID_TO_DTTM = &GLOBAL_HIGH_DTTM_VALUE
```

Here, the macro variable Global_High_DTTM_Value holds the parameter value:

```
"01JAN5999:00:00:00"DT
```

All global parameters are created after you run the initial jobs. The parameters are set to their default values. You can customize these parameters as per your requirement.

To view a list of the global parameter settings, see "Global Parameter Table" on page 47.

Set or Modify the Job Parameter Values

Job parameters are the control parameters for ETL jobs. You must set up these parameters for running the jobs. All parameters have a specific value or a default value that you can change as per your requirement.

To set or modify the value of a job parameter:

1. In SAS Data Integration Studio, right-click the job, and from the pop-up menu, select **Properties**. The Properties dialog box opens.
2. Click the **Parameters** tab. If parameters are defined for the job, then a list of all the parameters is displayed.
3. Select the parameter to be modified and click **Edit**. The Edit Prompt dialog box opens.
4. Click the **Prompt Type and Values** tab and in the **Default value** box, type the new value.
5. Click **OK** twice to close both the dialog boxes. The parameter value is changed.

In the **Parameters** tab, you can edit the existing parameters, if required. To view the jobs for which parameters need to be specified, see "Job Parameter Table" on page 56.

Ongoing Data Tasks

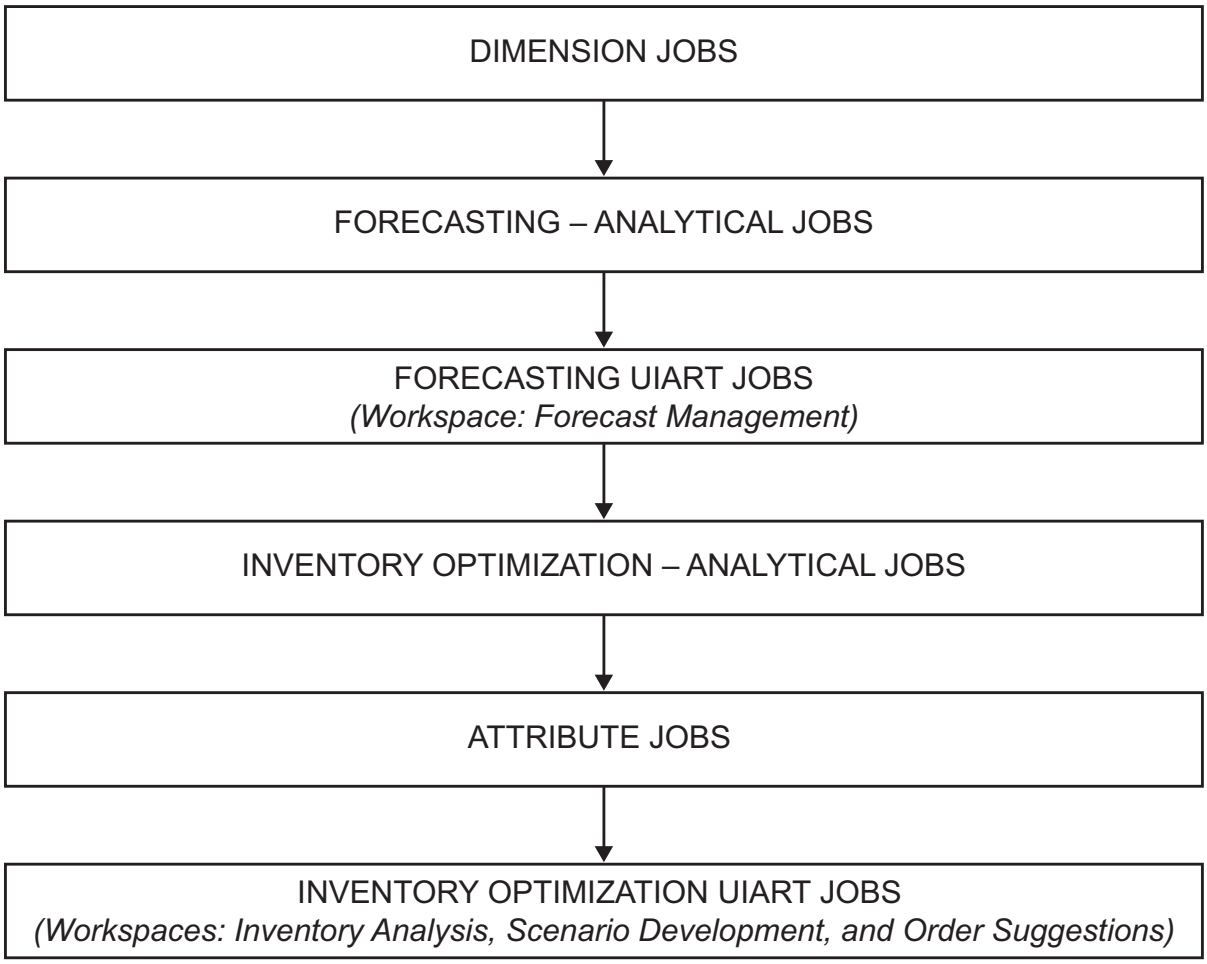
Overview

After you perform the pre-initial data load tasks, you can run the solution-specific jobs. These jobs need to be run periodically based on the base period that you specify in one of the control tables. The jobs can run on a weekly, monthly, or quarterly basis. However, you must run some jobs on a daily basis. The daily run of jobs ensure that any settings that are updated through the user interface are reflected on the interface on the next day.

During the nightly, weekly, monthly, or quarterly offline periods, you must run or schedule the ETL jobs to load initial or incremental data. Also, load the back-end jobs to run the solution-specific tasks.

The following figure shows the generic flow of the SAS Service Parts Optimization jobs.

Figure 3.1 *Basic Job Sequence*



The subsequent sections provide an overview of each of the jobs. For more information about each job, see Appendix A2, "ETL Job Details," on page 59.

Overview of Dimension Jobs

Attributes of the facility and item dimension tables might undergo change if there is an update to an existing facility or item, or if a new facility or item is added. When the dimension jobs run for the first time, the jobs load all the required attributes from the SDL. For ongoing loads, the jobs handle loading of incremental data by using extract transformation.

Note: Extract transform is a user-written transform that is provided with the ETL package of the solution. The transform extracts only incremental records from the source table.

The facility dimension table describes details of the distribution center warehouse with its five-level organization hierarchy. The facility attributes consist of facility ID, name, type code, and open and closure date time. The organization attributes consist of organization ID, name, and internal or external indicator.

The item dimension table describes the item with its ten-level item category hierarchy. The item attributes consist of item ID, name, description, group code, type code, status code, pack type code, sales introduction or discontinued data, brand name, and make or buy code. The table also stores item indicators, such as whether the item is a finished good, an assembly item, an item bundle, and so on.

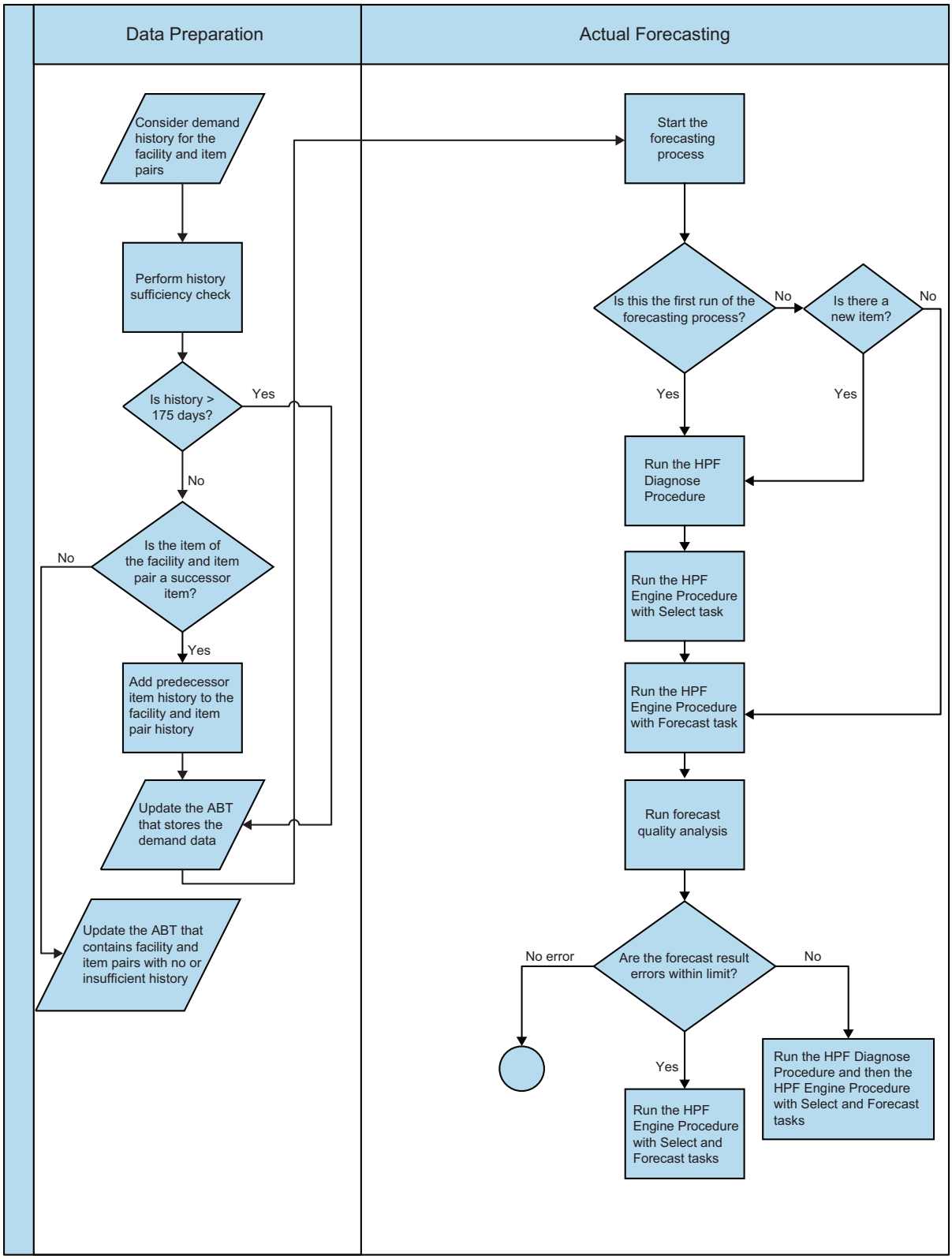
Forecasting: Analytical Jobs

Overview of Forecasting

The forecasting process in SAS Service Parts Optimization enables you to decide the future demand levels at user-defined hierarchies. The forecasting module provides forecast of the demand quantity over the planning horizon with historical demand history as input. The forecasting process can be run either in a batch mode or through ETL jobs.

The following figure explains the forecasting process flow for SAS Service Parts Optimization.

Figure 3.2 *The Forecasting Process*



The forecasting results are divided into different categories: normal forecast results, revisited forecast results, and low accuracy forecast results, depending on their accuracy.

The forecasting process identifies demand quantity for successor items and displays the forecast results on the user interface of the solution. Results for time series with insufficient or no history are also displayed on the user interface.

Forecasting Batch Process Jobs

The forecasting batch process jobs are categorized as follows:

- new-part forecasting and short-term forecasting jobs: If the SDL stores historical data for less than 175 days, then the new-part forecasting jobs are run. If the historical data is for 175 days or more, then the short-term forecasting jobs run.
- long-term forecasting jobs: If historical data for at least 40% of the planning horizon is available, the long-term forecasting jobs can run.
- jobs to populate user interface analytical result tables: These tables are used for displaying information on the user interface of the SAS Service Parts Optimization application.

To run the forecasting batch process:

1. Run the incremental demand data accumulation jobs to populate ABTs for short-term forecasting and new-part forecasting. The combined demand data ABT is used as an input by the high-performance forecasting (HPF) engine to generate forecasting results. All forecast results of different hierarchy levels are stored in separate folders.
2. Run the initial forecasting jobs. You must run these jobs whenever the forecast system is started.
    a. These jobs delete any existing forecast results. All data about time series classification that is based on forecast accuracy is deleted from the control tables.
    b. The main forecasting job diagnoses possible forecast models based on the available demand history, fits, and selects the best model as per the specified criteria.
    c. The jobs classify forecast results into normal, revisited, low accuracy, successor items, and items with no history, and perform subsequent corrective measures on the results.
    d. Monthly, quarterly, and weekly results are combined into a single result table and forecast results history is maintained.
3. Run subsequent forecasting jobs to incorporate any newly introduced time series and to rerun the forecasting process.
    a. These jobs handle any newly introduced time series.
    b. Prediagnosed and preselected models are used for forecasting. The jobs also classify all the time series into normal, revisited, and low accuracy forecast results and take subsequent corrective measures.
    c. Monthly, quarterly, and weekly results are combined into a single result table and forecast results history is maintained.
4. Run long-term forecasting jobs. Demand quantities of facility and item pairs to be used for forecasting are aggregated at a yearly level. The long-term forecasting results are obtained based on Regression and Bass model methods.

For more information about each job, see Appendix A2, “ETL Job Details,” on page 59.

Forecasting: UIART Jobs

The forecasting UIART jobs update tables that are used to display forecasting data on the user interface of the solution.

The tables in the UIART library contain the overall time series that can be divided into normal, revisited, and low accuracy forecast results. The tables also store time series with insufficient history and time series with successor items.

All the actual and forecasted values for items provide important statistical parameters. For example, root mean square error (RMSE), mean absolute percent error (MAPE), standard deviation (STD), forecasted lower limit, forecasted upper limit, and demand error.

While all the tables are stored in the UIART library, the No_History_Timeseries_Detail, Forecast_Result, and HPF_Preferences_UI tables are stored in the TSDB library for concurrent access of data.

The external demand data that is specified through the user interface is stored in a table. Also, default values of various options or parameters that are required for running the forecasting process are stored in a table. These parameters can be updated from the user interface of the solution. The modified parameters and the external demand data are used as input and forecasting process is run again.

Inventory Optimization: Analytical Jobs

Overview of Inventory Optimization

The inventory optimization back-end process in SAS Service Parts Optimization provides answers to the following questions:

- When should you place orders to restock inventory?
- What should be the appropriate inventory level?
- What is the projected customer service level?
- How should you replenish inventory to reduce costs and increase turns?

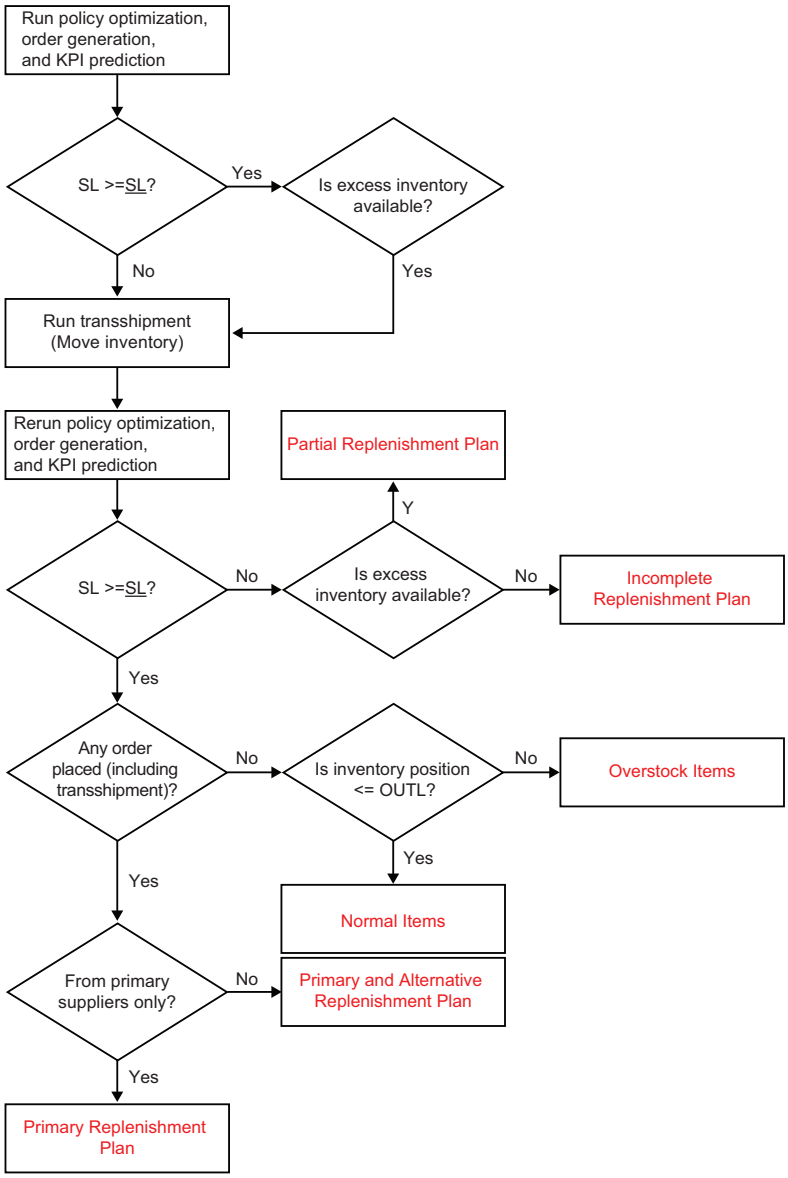
The inventory optimization process handles the following conditions:

- item succession, wherein the inventory data is populated such that the inventory of the predecessor can be merged with that of the successor items
- repaired parts, wherein the inventory data of the repaired parts can be added to the current inventory data
- back orders, wherein demand data can be adjusted for the back orders
- kit or bundled items, wherein demand can be specified for both, a pack or a bundle of items, and for individual items that constitute the bundle

The process provides optimized inventory replenishment policies and order quantities. Policy optimization is performed on the basis of specific constraints such as target service levels and user-defined inputs such as lead time and inventory cost; thereby, enabling users to maintain optimized stock levels over a multi-echelon service network, improve customer-satisfaction, and reduce total costs.

The following figure explains order generation process for SAS Service Parts Optimization.

Figure 3.3 *The Order Generation Process*



Legend

SL	Average Projected Service Level
SL	Lower Bound of Targeted Service Level
OUTL	Order-up-to Level

Inventory Optimization Batch Process Jobs

The inventory optimization batch process jobs are categorized as follows:

- jobs to create input tables for storing node, arc, demand, and inventory details that are required for the inventory optimization process
- jobs to run the inventory optimization process
- jobs to populate tables that are required for displaying information on the user interface of the solution

The inventory optimization batch process requires forecasted values of the customer demand and uses the following information from the SDL:

- node (facility and item pair) information such as lead time, service level requirement, and unit holding cost in a network
- network structure information or arcs between predecessors and successors
- the amount of inventory of an item to arrive at a facility for a period in the planning horizon

To run the inventory optimization batch process:

1. Specify the base period parameter in the Control.Global_Parameter_List table.
2. Run the jobs to create input tables for the inventory optimization process.
3. Run the inventory optimization process jobs. Any warning or error messages that occur when the process runs are stored in a table. The workflow of the inventory optimization process jobs is as follows:
   a. For all facility and item pairs, the process runs the MIRP procedure for policy optimization, order generation, and key performance indicator (KPI) prediction. All the facility and item pairs are grouped into two categories:
      - Facility and item pairs with the average projected service level value higher than or equal to the lower bound of the target service level. These facility and item pairs are further split into two subcategories:
        - facility and item pairs with excess inventories
        - facility and item pairs without excess inventories
      - Facility and item pairs with average projected service level value less than the lower bound of the target service level.

   b. The inventory optimization process then runs the transshipment module for the facility and item pairs with excess inventories. The transshipment module is also run for facility and item pairs with average projected service level value less than the lower bound of the target service level value.

      The transshipment module recommends the inventory to be transported from the excess facility and item pairs to the facility and item pairs in shortage, within each inventory pool. After the move, the inventory pipeline with excess facility and item pairs is updated with results that are suggested by transshipment.

   c. The inventory optimization process runs the MIRP procedure again for policy optimization, order generation, and KPI prediction, based on the updated inventory pipeline after transshipment.
4. Run the jobs to populate the tables that are required to display information on the user interface of the solution.

For more information about each job, see Appendix A2, “ETL Job Details,” on page 59.

Attribute Jobs

The attribute tables for facility and item are loaded periodically with the latest incremental data from the dimension tables. The attribute jobs load the required attribute description from the dimension and lookup tables and display the information on the user interface of the solution.

Here is a list of attributes that are loaded in the attribute tables:

- facility attributes
  - facility ID
  - facility name
  - facility type code and its description
  - organization ID
  - organization name
  - facility location hierarchy
- item attributes
  - item ID
  - item name
  - finished goods and assembly indicator
  - item type code and its description
  - make or buy code and its description
  - item status code and its description
  - item group code and its description
  - item category hierarchy

For more information about all the tables and their columns, see *SAS Service Parts Optimization: Data Dictionary*.

For more information about each attribute job, see Appendix A2, "ETL Job Details," on page 59.

Inventory Optimization: UIART Jobs

Overview

The UIART jobs are categorized into jobs for the following workspaces:

- Inventory Analysis
- Scenario Development
- Order Suggestions

Inventory Analysis Workspace

You can view details of inventory metrics in the Inventory Analysis workspace. The ETL jobs calculate inventory metrics and then display the values on the user interface of the solution.

Inventory metrics are calculated in three steps:

1. Metrics or key performance indicators (KPIs) and their attributes are calculated. For inventory optimization, the SAS Service Parts Optimization process calculates five metrics:
   - lead time
   - service level downstream
   - service level upstream

- demand
- inventory cost

All the metrics are calculated for the base period and are derived for the period under consideration. For example, if the base period is week, then start date and end date of the previous week are retrieved, and the metrics are calculated over this period.

For more information about calculation of each metric, see "Calculations of Inventory Metrics and Thresholds" on page 93.

2. The inventory metric periodic table is loaded. The periodic table is required for detailed view of the inventory on the user interface. The periodic table stores aggregated and definite metric values for different facility and pairs in a data table. A history table is also loaded that stores the history of all the actual metric values before the run date of the base period.
3. The inventory metric daily tables are loaded. To review the performance of the service supply chain, information in the Inventory Analysis workspace must be monitored and analyzed per day. This requires that the information be loaded and updated on a daily basis. The daily tables with inventory metric details include the following information:

   Alerts
   : include the alert limits for lead time, service level, and demand metrics. You can specify and control these limits for different item categories and facilities.

   Metric analytical result table (ART)
   : includes the average metric values with their actual, target, and threshold values for every period.

   Item category hierarchical data
   : includes the aggregated metric values for all items and facilities, grouped by item categories.

   Facility-level data
   : contains the aggregated metric values for all items, grouped by the corresponding facilities. Using this information, the data table displays metric values for multiple items, single facility combination on the user interface of the solution.

   Network-level data
   : includes the number of networks that contain a facility.

Scenario Development Workspace

You can develop and work with scenarios in the Scenario Development workspace. Scenario analysis enables you to conduct what-if analysis and solve operational dilemmas. For example, you can analyze the impact of change in service level, lead time, demand, and unit cost on the inventory cost and other cost metrics. You can perform four types of analysis. Each analysis is used for different purposes.

- Service level sensitivity analysis:

  In this type of scenario, you can select a facility, group of items, and specify the range of service level. This analysis uses the MIRP procedure to provide optimized inventory cost for the selected range of service level, for the specified facility and group of items.

- Customer-facing facility analysis:

  By performing this analysis, you can maximize the target service level for a selected group of items and customer-facing facilities, subject to the inventory cost. You can create a scenario by selecting items and customer-facing facilities. You can also specify the budget constraint and the minimum service level for the selected group of facility and item pairs. The ESLOPT procedure is used for optimizing the service level for

customer-facing facilities. You can view three types of metric values on the user interface:

- current values. Current metric values are obtained by running the procedure with the current service level value as specified in the SDL and without any budget constraint.
- optimized values. Optimized metric values at all the selected customer-facing facilities are obtained by running the procedure with specified budget and minimum service level constraints. If you do not provide a budget, then the total cost across the selected items and facilities is based on the current service level target is taken as the budget. If a minimum service level value is not specified, then the procedure runs without a minimum service level constraint.
- new values. By default, new metric values contain the optimized values. You can override the optimized service level values by specifying new values through the user interface. With these new values, service level values for the remaining facility and item pairs are optimized.

- Internal facility service level analysis:

  In this type of scenario, you can select items, facilities, and networks for analysis. The entire network that is associated with the selected items and facilities is analyzed. The analysis uses the MIRP procedure to optimize service levels at all internal facilities. The procedure uses the service level values for the customer-facing facilities. You can view three types of metric values:

  - current values. Current metric values are obtained by running the MIRP procedure with objective EVALISL for the current service levels that are specified in the SDL.
  - optimized values. Optimized metric values are obtained by running the procedure with objective OPTISL.
  - new values. By default, new metric values contain the optimized values. You can override the optimized service level values by specifying new values through the user interface. With the new values, the other metric values are evaluated. The evaluated values are obtained by running the MIRP procedure with objective EVALISL.

- Ad hoc analysis:

  Ad hoc analysis enables you to evaluate the impact of input parameters such as lead time, service level, projected demand, and unit cost of an item on policy parameters and costs. You can create input sets with different combinations of these input parameter values and then compare the current and optimized values in each set. The MIRP procedure with objective EVALISL evaluates the cost metrics for the changed values of metrics such as service level and lead time.

For more information about calculations of the scenario-specific metrics, see "Calculations of the Scenario-Specific Metrics" on page 100.

Order Suggestions Workspace

SAS Service Parts Optimization provides information about how much to order and when to receive the order. Recommendations about the quantities to be ordered from the primary and alternate channels for all facility and item pairs are also provided.

The order quantities from the primary channel are available as an output of the MIRP procedure. Order quantities from the alternate channels are available as an output of the transshipment process. Based on the order quantities that are recommended by the underlying process, the facility and item pairs are grouped into the following categories:

- Normal. Items in this category have the following features:

  - The average projected service level is higher than or equal to the lower bound of the target service level.
  - Orders are not placed and no transshipments are received.
  - Inventory positions are within the range of reorder level and order-up-to-level.
- Overstock. Items in this category have the following features:
  - The average projected service level is higher than or equal to the lower bound of the target service level.
  - Orders are not placed and no transshipments are received.
  - Inventory positions are higher than the order-up-to-level.
- Primary. Items in this category have the following features:
  - The average projected service level is higher than or equal to the lower bound of the target service level.
  - Orders are placed from the primary suppliers only and no transshipments are received.
- Primary and Alternative. Items in this category have the following features:
  - The average projected service level is higher than or equal to the lower bound of the target service level.
  - Transshipments are received from the inventory pool.
- Partial. Items in this category have the following features:
  - The average projected service level is less than the lower bound of the target service level.
  - Excess inventories are available in the inventory pool and shipping is possible. However, the transshipment module does not suggest the orders because they are not cost effective. You can ignore the cost impact and receive inventory from the inventory pool to fix the shortage problem.
- Incomplete. Items in this category have the following features:
  - The average projected service level is less than the lower bound of the target service level.
  - Excess inventories are not available in the inventory pool or the delivery time is not feasible. Even if you are willing to pay the high cost of orders, the inventory is not sufficient to fix the shortage problem. You need to order the items from locations that are outside the inventory pool or from external vendors to fix the shortage problem.

The facility and item pairs in the primary, primary and alternative, partial, and incomplete categories are called as low stock facility and item pairs. For each such facility and item pair, the system generates a unique identifier (ID). These IDs consist of seven characters followed by a sequential eight-digit number. For example, Primary00000001, PrimAlt00000001, Partial00000001, and Incompl00000001.

Chapter 4

Administration and Maintenance of SAS Service Parts Optimization

System Administration and Maintenance

Administrative Tools

The system or solution administrator might use the following administrative tools:

- SAS Management Console. You can use the following plug-ins of SAS Management Console to manage metadata for the SAS Metadata Server:
  - Server Manager
  - Data Library Manager
  - User Manager
  - BI Manager
  - Table Server Manager
  - Authorization Manager
- SAS Data Integration Studio. You can use the SAS Data Integration Studio to manage ETL and other jobs, which are sequences of steps for the extraction, transformation, and loading of data.

For more information about the administrative tools, refer to each product's online Help and related documentation.

Servers for Solution Use

The following table shows the servers that are used by the SAS Service Parts Optimization components.

Table 4.1 *Servers for Solution Components*

Server	Dependent Components
SAS Metadata Server	All SAS components
SAS Object Spawner (SAS Workspace Server, Pooled SAS Workspace Server, and SAS Stored Process Servers)	Batch jobs and middle tier
SAS Service Parts Optimization Middle Tier	Client tier
Table Server	Middle tier
WIP Middle Tier	Middle tier and client tier
Remote Services	Middle tier
SAS Analytics Platform Server	Forecasting-specific batch jobs and middle-tier components
Application Server (JBoss, WLS, or WAS)	Client tier

Maintenance Activities

To maintain the architecture, back-end data, and solution client, you must perform the maintenance activities listed in the following table.

Table 4.2 *Maintenance Activities for SAS Service Parts Optimization*

Maintenance Activity	Comments
Stop and start the middle-tier server as required for the incremental data flow updates and nightly jobs.	The incremental data update does not cause any problem to the middle tier and client tier. However, if some client task is being performed when data is updated in the UIART tables, then the client task might fail. To avoid failure of the client task, you can stop the middle tier until the data load completes.
Maintain a log file for the SAS Service Parts Optimization middle-tier server.	Log files must be archived time to time. The middle-tier application server creates log files that are sorted by date. Log files for workspace server and stored process server are located in the `log` folder.
Install updates to software.	None.

Maintenance Activity	Comments
Back up and restore data, as required by your organization.	Site-specific.
Back up the SAS Metadata Server.	For details, see documentation for SAS Intelligence Platform.

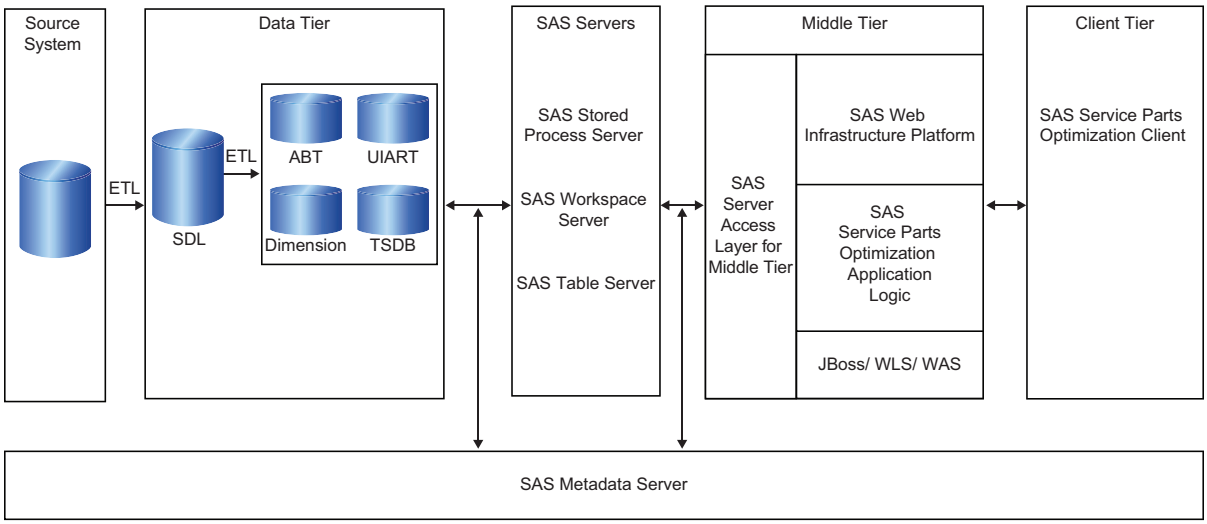
Note: If you want to change the name of the SAS Workspace Server to a language other than English, ensure that the spoDatabase.properties file is encoded in UTF-8 format.

Tier Dependencies

SAS Service Parts Optimization has three functional tiers:

- server tier
- middle tier
- client tier

Figure 4.1 *Tier Architecture of SAS Service Parts Optimization*



Legend

ETL	Extract, Transform, Load
SDL	Solution Data Layer
ABT	Analytical Base Table
UIART	User Interface Analytical Result Table
TSDB	Table Server Database
WLS	WebLogic Server
WAS	Websphere Application Server

Servers and services in a tier might depend on servers on other tiers. The dependency among the tiers is provided in the following tables.

Table 4.3 *Tier Dependency Summary*

Tier	Dependent On
Middle tier	Server tier
Client tier	Middle tier

Table 4.4 *Tier Dependency Details*

Server or Service	Dependent On
Middle tier	
SAS Service Parts Optimization middle tier	1. Table server 2. Object spawner 3. Remote services 4. SAS Web Infrastructure Platform (WIP) services
Client Tier	
SAS Service Parts Optimization Client	1. SAS Service Parts Optimization middle tier 2. SAS WIP services

Note: When you start any of the dependent server or service, ensure that you start the dependent server or service in the order that is specified in the earlier table.

Administer Users

Each SAS Service Parts Optimization user is assigned a role that determines the functions that the user can perform in the SAS Service Parts Optimization application.

The following table provides descriptions for the different user roles in SAS Service Parts Optimization.

Table 4.5 *SAS Service Parts Optimization Roles and Descriptions*

Role	Display Name	Description
Buyer	SPO: Buyer	Performs all tasks within the Order Suggestions workspace.

Role	Display Name	Description
Forecaster	SPO: Forecaster	Performs all tasks within the Forecast Management workspace, except editing the forecasting process parameters.
Advance Forecaster	SPO: Advance Forecaster	Performs all tasks within the Forecast Management workspace.
Inventory Analyst	SPO: Inventory Analyst	Performs all tasks within the Inventory Analysis and Scenario Development workspaces, except promoting scenarios.
Advance Inventory Analyst	SPO:Advance Inventory Analyst	Performs all tasks within the Inventory Analysis and Scenario Development workspaces.

One of these roles must be assigned to every SAS Service Parts Optimization user.

After you create users for SAS Service Parts Optimization, you must map their IDs (that you create in the SAS Management Console) with their corresponding employee IDs.

To map a user ID with its corresponding employee ID:

1. Browse to the folder that you defined for the UIART library. For Windows operating environment, the default path to this folder is as follows: **`C:\SAS\SASConfig\Lev1\AppData\SASServicePartsOptimization\data\uiart`**
2. Open the **`user_mapping.sas7bdat`** file with SAS 9.2. The SAS window opens and displays a table with the following columns:
   - SMC Identifier
   - Business Key for the Employee
   - User Identifier
   - Buyer Indicator
   - Email Identifier
3. Ensure that the **Business Key for the Employee** and **User Identifier** columns display details for all the employees. This list is populated as a part of the installation of the solution and is updated periodically depending on the defined base period.
4. For the user that you want to map, identify its business key and in the SMC Identifier column, type the user ID that you specified while creating the user.
5. In the Buyer Indicator column, if the user is a buyer, type *1*, else type *0* for a non-buyer user.
6. In the Email Identifier column, type the user's e-mail ID.
7. Save all changes.

Troubleshooting Instructions

The Client Fails to Open

Problem:

You provided correct logon credentials, but the client application fails to launch.

Possible Cause:

The client application could be incorrectly installed.

Possible Solution:

Re-install the client application.

The Client Cannot Connect to the Server

The client cannot connect to the server.

Table 4.6 Possible Causes and Solutions for Connection Problems

Possible Cause	Possible Solution
The sas-environment.xml file contains an incorrect URL in the <service-registry> tag.	Provide the correct URL in the sas-environment.xml file.
The SAS Application Server is not started at the specified computer and port.	Start the SAS Application Server at the correct computer and port.
The SAS Application Server started but with errors.	Open the log files to determine the root cause of the problem and take appropriate action. For more information about some of the possible causes for the errors and their corresponding solutions see, "The SAS Application Server Starts with Errors" on page 44.
Network connection between the client and server is broken.	Check the network connection between the client and the server and take corrective actions, if needed.
The SAS environment is incorrectly configured.	In SAS Management Console, manually specify the correct URL and port for the server in the SAS environment.

The SAS Application Server Starts with Errors

The SAS Application Server starts but with errors.

Table 4.7 *Possible Causes and Solutions for Server-Related Errors*

Possible Cause	Possible Solution
The server is incorrectly installed.	Re-install the server.
The server could not connect to the database.	Start the SAS database servers by using the correct database computer and port details. Then, verify that the network connection between the server and the database servers is working.

Appendix 1

Global and Job Parameter Table Details

Global Parameter Table

The following table lists all the global parameters. You can change these parameters as per your requirement.

Table A1.1 *Global Parameters and Their Descriptions*

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
History management process	GLOBAL_HIGH_DTTM_VALUE	Specifies a globally conventional high date value (datetime format) to identify valid records. You can change this date value as per your requirement.	"01JAN5999:00:00:00"DT

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
Extract, transform, and load (ETL) process	GL_SHORT_YES	Specifies a short value to indicate yes.	Y
	GL_SHORT_NO	Specifies a short value to indicate no.	N
	GL_WK_BASE_PERIOD	Specifies a short value to indicate week as the base period.	W
	GL_MTH_BASE_PERIOD	Specifies a short value to indicate month as the base period.	M
	GL_QTR_BASE_PERIOD	Specifies a short value to indicate quarter as the base period.	Q
	GL_BASE_PERIOD	Specifies a short value to indicate the base period for inventory optimization. The possible values are as specified in the GL_WK_BASE_PERIOD, GL_MTH_BASE_PERIOD, and GL_QTR_BASE_PERIOD parameters. You must set this parameter to your selected base period value.	W
	GL_SHORT_BOTH	Specifies a short value to indicate that the facility is both internal and external	B
	GL_SHORT_EXTERNAL	Specifies a short value to indicate that the facility is external.	E
	GL_TIME_CAL_ASSOC_CD1	Specifies the association code for standard time calendar hierarchy: Day → Week	CL1

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
Extract, transform, and load (ETL) process	GL_TIME_CAL_ASSOC_CD2	Specifies the association code for standard time calendar hierarchy: Day → Month → Quarter → Year	CL2
	GL_TIME_LTF_ASSOC_CD	Specifies the association code for long-term forecasting time calendar hierarchy: Day → Month → Quarter → Year	CL3
	GL_UNSPECIFIED_CATEGORY	Specifies a value for the item category and the facility or organization hierarchy that are not specified.	Unspecified
	GL_MIXED_HIERARCHY	Specifies mixed hierarchy for forecasting.	MIXED
	GL_NOT_MIXED_HIERARCHY	Specifies non-mixed hierarchy for forecasting.	NOT MIXED
	GL_ITEM_PROFILE_A	Specifies class A item profile.	A
	GL_ITEM_PROFILE_B	Specifies class B item profile.	B
	GL_ITEM_PROFILE_C	Specifies class C item profile.	C

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
Forecasting process	GL_DATA_STORAGE_PATH	Specifies the path in the environment where all the hierarchical forecasting data sets are created. The data sets include actual demand and prediction results along with model repositories and events data. You must set this parameter value to the path where you want to store the forecasting data sets.	C:\
	GL_FORECAST_START_DT	Specifies the start date of the actual demand to be considered for fitting the forecasting model for all forecast groups. You must set this parameter to the date from when you want to consider the data for forecasting.	FIRST
	GL_FORECAST_END_DT	Specifies the end date of the actual demand to be considered for fitting the forecasting model for all forecast groups. You must set this parameter to the date until when you want to consider the data for forecasting.	LAST

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
Inventory optimization process	GL_PLANNING_HORIZON	Specifies the number of base periods (any value > 0) for which policy parameters, key performance indicators, or both are computed for inventory optimization. You must set this parameter to a value >= max (lead time as per base period) + 1.	6
	GL_IO_BATCH_CURR_DATE	Specifies the current date that is used to populate the inventory data and demand data for inventory optimization. The default date is the current date. However, a specific date can be provided. For example, "01JAN2011"D. You must change this parameter value to the date from when your inventory optimization process starts.	date()
	GL_SERVICE_TYPE	Specifies the service type that is used to calculate the service level. The possible values are FR for fill rate, RR for ready rate, and BR for backorder ratio. You must modify this parameter as per your selected service type.	FR

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
Inventory optimization process	GL_NODE_PRICE_TYPE_CD	Specifies the price type code to be used in node data for inventory optimization. You must modify this parameter to the price type code as per your data.	STD
	GL_IO_MIN_CV	Specifies the minimum value of coefficient of variation that is related to demand forecast. If the coefficient of variation for a demand forecast value is less than the specified minimum value of coefficient of variation, then the MIRP procedure outputs a warning message. The variance of the demand value is increased to meet the value in this parameter. You must modify this parameter as per your requirement.	0.01

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
Inventory optimization process	GL_IO_MAX_CV	Specifies the maximum value for coefficient of variation that is related to demand forecast. If the coefficient of variation for a demand forecast value is greater than the specified maximum value of coefficient of variation, then the MIRP procedure outputs a warning message. The variance of the demand value is decreased to meet the value in this parameter. You must modify this parameter as per your requirement.	1
	GL_POOLING_NETWORK_FLG	Specifies whether the network is a pooling network by comparing the value of the network flag with the value of this variable.	P

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
User interface	GL_CURRENCY_SYMBOL	Specifies the symbol or abbreviation for currency. You must modify this parameter as per your currency.	USD
	GL_FCST_UI_DISPLAY_VALUES	Specifies the number of periods for which actual demand quantity is to be displayed in the Forecast Management workspace. You must modify this parameter to the number of past periods for which you want to view data on the user interface.	5
	GL_IO_METRICS_HISTORY_PERIOD	Specifies the number of base periods of historical data to be displayed in the Inventory Analysis workspace. You must modify this parameter as per your requirement.	2
	GL_DEMAND_THRESHOLD_COEFF	Specifies the coefficient value (a positive number < 3) for the given Confidence Interval (CI). The coefficient value is used to calculate the lower and upper bounds for the forecasted demand. You can modify the coefficient value as per your requirement.	1.96

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
User interface	GL_IO_ITEM_CAT_LVLNO	Specifies the number of levels for item categories to be displayed in the Inventory Analysis workspace. You must change this parameter as per the number of hierarchical levels in the item category.	3
	GL_LOOSE_CNTRL_TYPE	Specifies the loose receipt control type that is used to determine the period boundaries for calculating the service level.	LOOSE
	GL_TIGHT_CNTRL_TYPE	Specifies the tight receipt control type that is used to determine period boundaries for calculating the service level.	TIGHT
	GL_RECEIPT_CNTRL_TYPE	Specifies receipt control type (possible values are LOOSE or TIGHT) that is used to determine period boundaries for calculating the service level. You can modify this parameter as per your requirement.	LOOSE
	GL_SERVICE_THRESHOLD	Specifies the threshold to calculate the lower and upper bounds of service level. You can modify this threshold value as per your requirement.	0.02

Category (Parameter Used In)	Parameter Name	Parameter Description	Default Value
User interface	GL_MAX_BUDGET	Specifies a high value for the inventory cost budget to be used for customer-facing facility analysis in the Scenario Development workspace.	1.00E+15
	GL_MIN_DECREASE_COST_PCT	Specifies the maximum percentage up to which the unit cost of an item can be decreased in ad hoc analysis in the Scenario Development workspace. You can modify this parameter value as per your requirement.	10

Job Parameter Table

The following table lists the jobs and their parameters that need to be specified.

Table A1.2 *Job Parameters and Their Descriptions*

Job Name	Parameter Name	Parameter Description	Default Value
05_CREATE_TIME_PERIOD_DATA	CAL_YR_START_DT	Specifies the start date of the year from which data would be generated.	'01JAN2000'D
	CAL_NO_OF_YRS	Specifies the number of years for calendar time dimension.	15
	SUMM_NO_OF_YRS	Specifies the number of years for which the summarized time dimension is to be populated. This is used in long-term forecasting.	15
	LOADING_DTTM	Specifies the date time value to be loaded.	'01JAN2000:00:00:00'DT
4_SPLIT_DIAGNOSE_SELECT_FORECAST	METAREPOSITORY	Specifies the metadata repository.	Foundation
02_LTF_PROCESS	OUTLIER_START_PERIOD	Specifies the period from which the outlier detection process is to be started.	1
	MAX_NO_MISSING	Specifies the maximum number of missing values that are acceptable in a time series.	0

Appendix 2

ETL Job Details

Overview

You can access all ETL jobs from SAS Data Integration Studio. The basic location of all ETL jobs for SAS Service Parts Optimization is as follows: **`/Products/SAS Service Parts Optimization/Service Parts Optimization 4.2`**.

The ETL jobs are placed under specific folders in this location. The subsequent topics explain the ETL jobs and provide only the specific location for the jobs. For example, the location for the initial one-time jobs is given as **`CONTROL/INITIAL_ONE_TIME_JOBS`**. This folder structure is available under the basic location of the ETL jobs.

Note: The input requirements that are mentioned for each job are directly used by that job. You do not need to provide any inputs or perform any prerequisite tasks.

Initial ETL Job Sequence

The following table lists the sequence of the ETL jobs that are used in SAS Service Parts Optimization. You must follow this sequence when you are planning to run the jobs for the first time. Later, each job is to be run either daily or on the basis of the base period.

The value that is displayed in **`monospace`** format is the directory or container for the subsequent jobs. The Predecessors column lists the job numbers of the jobs that must be run before a particular job is run.

Table A2.1 *ETL Job Sequence*

Job Number	Job Name	Predecessors
Dimension Jobs		
`01_DIMENSION_JOBS`		
1	FACILITY_DIM	None
2	ITEM_DIM	None
Forecasting Jobs		
`A_CREATE_DEMAND_ABT`		
3	1_NPF_DEMAND_ABT	1,2
4	2_SPLIT_NPF_ABT_DEPENDING_ON_HISTORY	3
5	3_STF_DEMAND_ABT	1,2
6	4_COMBINE_STF_NPF_ABT	4,5
`INITIAL_FORECASTING_JOBS`		
7	1_DELETE_RESULT_FOLDERS	None
8	2_DELETE_WORKFLOW_CONTROL_DATA	None
9	3_CREATE_HPF_PREFERENCES_UI_TABLE	None
10	4_SPLIT_DIAGNOSE_SELECT_FORECAST	3,4,5,6,7,8,9

Job Number	Job Name	Predecessors
11	5_CREATE_HPF_ART	10
12	6_HPF_VERSIONING_ART	11
13	7_CREATE_HPF_OUTFOR_ART	10
SUBSEQUENT_FORECASTING		
14	1_CREATE_HPF_PREFERENCES_UI_TABLE	None
15	2_SPLIT_DIAGNOSE_SELECT_FORECAST	7,8,9,10,11,12,13
16	3_FORECAST	15
17	4_CREATE_HPF_ART	16
18	5_HPF_VERSIONING_ART	17
19	6_CREATE_HPF_OUTFOR_ART	16
A_CLUSTERING		
20	01_SELECT_FACILITY_ITEM_FOR_LTF_CLUSTER	None
21	02_CALCULATE_YEAR_DEMAND_HISTORY_LTF_CLUSTER	20
22	03_LTF_GET_CLUSTER_PARAM	None
23	04_LTF_CLUSTERING_PROCESS	21,22
B_BATCH_PROCESS		
24	01_LTF DEMAND_DATA_AGGREGATION	20,22
25	02_LTF_PROCESS	24
UIART Jobs for the Forecast Management Workspace		
D_UI_TABLES		
26	01_CREATE_UI_TABLES	4

Job Number	Job Name	Predecessors
27	02_CREATE_DEMAND_PROJECTION_DETAIL	13
28	03_NO_HISTORY_DP_DETAIL	4
29	04_ONE_TIME_BUY	25
30	05_CREATE_TIMESERIES_DETAIL	13
31	06_NO_HISTORY_TIMESERIES_DETAIL	4
32	08_FILTER_ATTRIBUTE_RANGE	27
33	09_NO_HIST_FILTER_ATTRIBUTE_RANGE	28
E_FORECAST_IO_INTEGRATION		
34	01_AGGREGATE_FCST_ART	7,8,9,10,11,12,13,14,15,16,17,18,19
35	02_AGGREGATE_FCST_ART_FROM_NPF	34, 31
F_INTERMITTENT_DEMAND		
36	GET_INTERMITTENT_DEMAND	34,35
Inventory Optimization Jobs		
A_CREATE_IO_ABT		
37	01_POPULATE_NODE_DATA	36
38	02_POPULATE_ARC_DATA	None
39	03_POPULATE_DEMAND_DATA	34,35,36
40	04_POPULATE_INVENTORY_DATA	None
B_IO_PROCESS		
41	01_POLICY_ORDER_KPI_1	37,38,39,40

Job Number	Job Name	Predecessors
42	02_TRANSSHIPMENT	41
43	03_POLICY_ORDER_KPI_2	42
C_UI_ART_JOBS		
01_ATTRIBUTE_JOBS		
44	FACILITY_ATTR	1
45	ITEM_ATTR	2
Inventory Optimization Metric Calculation Jobs		
02_METRICS_CALC		
46	01_GET_LEAD_TIME	41,42,43
47	02_GET_SL_DOWNSTREAM	37,38,39,40
48	03_GET_SL_UPSTREAM	37,38,39,40
49	04_GET_DEMAND_ACT	None
50	05_GET_DEMAND_TGT	41,42,43
51	06_GET_COST_ACT	None
52	07_GET_COST_TGT	41,42,43
53	08_GET_KPI_PREDICT_min_max	41,42,43
UIART Jobs for the Inventory Metrics View		
03_METRICS_BASE_PERIOD		
54	01_IO_METRICS_PERIODIC_ART_HIST	41,42,43,46,47,48,49,50,51,52,53
55	02_IO_METRICS_PERIODIC_ART_LOAD	54
56	03_IO_METRICS_PERIODIC_ART_REPLACE	55
04_METRICS_DAILY		
57	01_IO_METRICS_ALERT_SETTINGS_LOAD	37,38,39,40,46,53

Job Number	Job Name	Predecessors
58	02_IO_METRICS_ALERT_SETTINGS_REPLACE	57
59	03_IO_METRICS_ART_LOAD	1,2,54,55,56,58
60	04_IO_METRICS_ART_VENDOR_DETAILS	59
61	05_IO_METRICS_ART_REPLACE	60
62	06_IO_METRICS_ITEM_FACITY_NETWORK	61
UIART Jobs for the Scenario Development Workspace		
05_SCENARIO_BASE_PERIOD		
63	SCENARIO_UI_TABLES	37,38,39,40
UIART Jobs for the Order Suggestions Workspace		
06_ORDERS_BASE_PERIOD		
64	01_CREATE_LOCK_PLAN_SETTING	To be run only once after populating the SDL tables.
65	02_CREATE_SKU_BUCKETS	41,42,43,44,45,58
66	03_POPULATE_DRP_TABLE	65
67	04_POPULATE_ORDER_RESULT	66
68	05_POPULATE_REPL_BY_PERIOD_LOOKUP	63,67
69	06_POPULATE_ORDER_DETAIL	68
70	07_POPULATE_REPL_PLAN_DETAIL	69
71	08_POPULATE_REPL_PLAN_METRICS	70
72	09_CREATE_REPL_PLAN_SUMMARY	71
07_ORDERS_DAILY		

Job Number	Job Name	Predecessors
73	LOAD_ORDER_DETAILS	72

Initial One-Time Jobs

The following table provides an overview of the initial one-time jobs. These jobs need to be run only once after you install the solution. The specific folder structure for these jobs is as follows:

CONTROL/INITIAL_ONE_TIME_JOBS

Table A2.2 *Initial One-Time Jobs*

Job Name	Job Description	Input Requirements
01_CREATE_GLOBAL_PARAMETER_LIST	This job populates the Control.Global_Parameter_List table and globally initializes all the parameters.	None
02_CREATE_CONTROL_TABLES	This job creates and populates the control tables by using the script files **load_spo_io_control_data.sas** and **create_spo_fcst_control_table.sas**. These tables are used by the forecasting and inventory optimization process. This job also creates the Control.Job_Status table.	None
03_CREATE_FCST_ABT	This job creates a structure for all the analytical base tables (ABTs) that are required by the forecasting process.	None

Job Name	Job Description	Input Requirements
04_CREATE_TIME_PERIOD_DATA	This job generates data for Time_Period, Time_Period_Assoc, Time_Period_Assoc_Type, and Period_Type tables of the solution data layer (SDL). The job also creates hierarchies by using associations between the various time period entities. • General Calendar: Day → Week • Summarized Calendar: Day → Month → Quarter → Year	You can specify parameters such as start date and number of years for which data is to be generated, as values of the parameters for the job.
05_CALENDAR_HIERARCHY1	This job reduces the levels in the calender hierarchy1 (Day → Week) and stores the output in the Scratch.Calendar_Flatten_Hier1 table.	The tables SDL.Time_Period and SDL.Time_Period_Assoc are required.
06_CALENDAR_HIERARCHY2	This job reduces the levels in the calender hierarchy2 (Day → Month → Quarter → Year) and stores the output in the Scratch.Calendar_Flatten_Hier2 table.	The tables SDL.Time_Period and SDL.Time_Period_Assoc are required.
07_TIME_DIM	This job populates the time dimension table DIM.Time_Dim.	The tables Scratch.Calendar_Flatten_Hier1 and Scratch.Calendar_Flatten_Hier2 are required.
08_CREATE_TIME_VIEWS	This job creates week, month, quarter, and year time views from the DIM.Time_Dim table.	The tables DIM.Time_Dim DIM.Week_Dim, DIM.Month_Dim, DIM.Quarter_Dim, DIM.Year_Dim are required.
09_CREATE_USER_TABLES	This job creates the following initial tables that are required for populating the user interface: • UIART.User_Mapping • TSDB.User_Preferences • TSDB.IO_Metrics_Alertsettings • TSDB.Scenarios • TSDB.Scenario_Params	None

Job Name	Job Description	Input Requirements
10_POPULATE_HPF_PREFERENCES	This job populates the tables Control.HPF_Preferences and Control.Forecast_GR_Seasonality. The Control.Forecast_GR_Seasonality table divides forecast groups on weekly, monthly, or quarterly basis.	None

Dimension Jobs

The following table provides an overview of the dimension jobs. These jobs must be run whenever the input tables that are required by these jobs are either loaded for the first time or updated later. The specific folder structure for these jobs is as follows:

SPO_JOBS/01_DIMENSION_JOBS

Table A2.3 *Dimension Jobs*

Job Name	Job Description	Input Requirements
FACILITY_DIM	This job populates the dimension table for a facility. The job extracts the latest records from the facility table, and then loads the extracted records in the Facility_Dim table.	The tables SDL.Facility, SDL.Location, and SDL.Organization are required.
ITEM_DIM	This job populates the dimension table for an item. The job extracts the latest records from the source item table, and then loads the extracted items in the Item_Dim table.	The tables SDL.Item and SDL.Item_Category are required.

Forecasting Jobs

Short-Term and New-Part Forecasting Jobs

The following table provides an overview of the short-term and new-part forecasting jobs. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/A_CREATE_DEMAND_ABT

Table A2.4 *Short-Term and New-Part Forecasting Jobs*

Job Name	Job Description	Input Requirements
1_NPF_DEMAND_ABT	This job generates the demand history data for new part forecasting process. The data consists of time series with less than 175 days of historical data. Successor items use the historical data of their predecessors. The combined history of the predecessor and successor are aggregated as demand history of the successor item. The predecessor and successor pair is network-specific.	The following tables are required: • SDL.Facility_Item_Demand • SDL.Item_Succession • DIM.Time_Dim • SDL.Forecast_Group_Item_Detail • SDL.Facility_Item_Ind_Variable • DIM.Item_Dim • DIM.Facility_Dim
2_SPLIT_NPF_ABT_DEPENDING_ON_HISTORY	This job divides the time series in the input table depending on history. The output table NPF_Demand_ABT_Less_History contains time series with history less than 175 days. This history data is either the history of the item at a particular facility or the data can be the combined history of the predecessor and the successor items. The second output table NPF_Demand_FCST_Dataset consists of time series that are qualified for forecasting. This history data can be the combined data of the predecessor and the successor items.	The ABT.NPF_FCST_Demand_ABT table is required.
3_STF_DEMAND_ABT	This job generates the demand history data for short-term forecasting. The data consists of time series with history greater than 175 days. If a time series has a successor item, then history of its predecessor item is not considered.	The following tables are required: • SDL.Facility_Item_Demand • SDL.Item_Succession • DIM.Time_Dim • SDL.Forecast_Group_Item_Detail • SDL.Facility_Item_Ind_Variable • DIM.Facility_Dim

Job Name	Job Description	Input Requirements
4_COMBINE_STF_NPF_ABT	This job combines the time series data from the analytical base tables for new-part forecasting and short-term forecasting. The combined data is stored in the table Forecast_Demand_Data_ABT that is used as an input for the forecasting batch process.	The following tables are required: • ABT.NPF_Demand_ABT_FCST_Dataset • ABT.STF_FCST_Demand_ABT

Initial Forecasting Jobs

The following table provides an overview of the initial forecasting jobs. These jobs must be run only once after you install the solution. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/B_SPLIT_DIAGNOSE/HPF/INITIAL_FORECASTING_JOBS

Table A2.5 *Initial Forecasting Jobs*

Job Name	Job Description	Input Requirements
1_DELETE_RESULT_FOLDERS	This job clears all the previously stored forecast results, that is, all the folders and subfolders that are present in the **HPF** folder are deleted. This job is to be run whenever the forecasting system is restarted.	None
2_DELETE_WORKFLOW_CONTROL_DATA	This job clears all the records from the table Control.FCST_Control_Data. This job is to be run whenever the forecasting system is restarted.	None
3_CREATE_HPF_PREFERENCES_UI_TABLE	This job first confirms whether the TSDB.HPF_Preferences_UI table exists or not. If the table does not exist, the job creates it. The job then populates the TSDB.HPF_Preferences_UI table. Data from this table is used to display on the user interface of the solution. This job also handles any newly introduced forecast groups.	None

Job Name	Job Description	Input Requirements
4_SPLIT_DIAGNOSE_SELECT_FORECAST	This job runs the HPFDIAGNOSE and HPFENGINE procedures and the QA algorithm for all the time series and populates the tables ABT.MTH_Forecast_ART, ABT.QTR_Forecast_ART, and ABT.WK_Forecast_ART with monthly, quarterly, and weekly forecast results. This is a one-time job unless new facility and item pairs are added to the system such that their demand history is more than 175 days. These facility and item pairs then qualify for the forecasting process. The model repository population, selection of model, and forecasting tasks are performed by this job.	The table ABT.Forecast_Demand_Data_ABT is required.
5_CREATE_HPF_ART	This job populates the ABT.HPF_Forecast_ART table by using data from the input tables. The job performs a simple aggregation of all the forecast results that are obtained at different intervals (weekly, monthly, quarterly).	The tables ABT.QTR_Forecast_ART, ABT.MTH_Forecast_ART, and ABT.WK_Forecast_ART are required.
6_HPF_VERSIONING_ART	This job maintains a history of the previously run forecasts in the ABT.HPF_Forecast_ART_History table.	The table ABT.HPF_Forecast_ART is required.
7_CREATE_HPF_OUTFOR_ART	This job generates the actual demand and the predicted demand data in the table HPF_Forecast_Outfor_ART. The actual demand data is provided for the period that you specify. The global symbol table stores this data as parameter GL_FCST_UI_Display_Values.	The tables QTR_Forecast_Outfor_ART, MTH_Forecast_Outfor_ART, and WK_Forecast_Outfor_ART are required.

Subsequent Forecasting Jobs

The following table provides an overview of the subsequent forecasting jobs. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/B_SPLIT_DIAGNOSE/HPF/SUBSEQUENT_FORECASTING

Table A2.6 *Subsequent Forecasting Jobs*

Job Name	Job Description	Input Requirements
1_CREATE_HPF_PREFERENCES_UI_TABLE	This job first confirms whether the TSDB.HPF_Preferences_UI table exists or not. If the table does not exist, the job creates it. The job then populates the TSDB.HPF_Preferences_UI table. Data from this table is used to display on the user interface of the solution. This job also handles any newly introduced forecast groups.	The Control.HPF_Preferences table is required.
2_SPLIT_DIAGNOSE_SELECT_FORECAST	This is a one-time job unless new facility and item pairs are added to the system such that their demand history is more than 175 days. These facility and item pairs then qualify for the forecasting process. After the initial run, this job considers only the incremental item and facility demand history. The model repository population, selection of model, and forecasting tasks are also performed by this job.	The table ABT.Forecast_Demand_Data_ABT is required.
3_FORECAST	This job runs the HPFENGINE procedure and the QA algorithm for all the time series and generates the forecast results in the ABT.MTH_Forecast_ART, ABT.QTR_Forecast_ART, and ABT.WK_Forecast_ART tables. After the initial run, this job must be run for all the subsequent periods. The HPFENGINE procedure runs for all time series in the control data. After the QA macros are run, depending on the classification of the forecast results into normal, revisited, and low accuracy forecasts, the post-QA task runs. Finally, the result tables are updated for all the different intervals (weekly, monthly, quarterly).	The table ABT.Forecast_Demand_Data_ABT is required.

Job Name	Job Description	Input Requirements
4_CREATE_HPF_ART	This job populates the table ABT.HPF_Forecast_ART by using data from the input tables. The job performs a simple aggregation of all the forecast results that are obtained at different intervals (weekly, monthly, quarterly).	The tables ABT.QTR_Forecast_ART, ABT.MTH_Forecast_ART, and ABT.WK_Forecast_ART are required.
5_HPF_VERSIONING_ART	This job maintains the history of previously run forecasts in the ABT.HPF_Forecast_ART_History table.	The table ABT.HPF_Forecast_ART is required.
6_CREATE_HPF_OUTFOR_ART	This job generates the actual demand and the predicted demand data in the table HPF_Forecast_Outfor_ART. The actual demand data is provided for the period that you specify. The global symbol table stores this data as parameter GL_FCST_UI_Display_Values.	The tables QTR_Forecast_Outfor_ART, MTH_Forecast_Outfor_ART, and WK_Forecast_Outfor_ART are required.

Clustering Jobs for Long-Term Forecasting

The following table provides an overview of the clustering jobs for long-term forecasting. These jobs must be run when you want to perform clustering. The specific folder structure for these jobs is as follows:

`SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/C_LTF/A_CLUSTERING`

Table A2.7 *Clustering Jobs for Long-Term Forecasting*

Job Name	Job Description	Input Requirements
01_SELECT_FACILITY_ITEM_FOR_LTF_CLUSTER	This job populates the table Scratch.LTF_Items_TMP with time series details for all forecast groups that are to be used for the clustering process for long-term forecasting.	The tables SDL.Forecast_Group and SDL.Forecast_Group_Item_Detail are required.

Job Name	Job Description	Input Requirements
02_CALCULATE_YEAR_DEMAND_HISTORY_LTF_CLUSTER	This job generates the demand history for all time series at a yearly interval in the ABT.LTF_Cluster_ABT table. The demand history for the complete planning horizon is then used for dividing facility and item pairs into clusters.	The following tables are required: • SDL.Facility_Item_Demand • Scratch.LTF_Items_TMP • DIM.Time_Dim • DIM.Year_Dim
03_LTF_GET_CLUSTER_PARAM	This job populates the parameters that are used while creating clusters for long-term forecasting in the ABT.LTF_Cluster_Param_ABT table.	The table SDL.Forecast_Group is required.
04_LTF_CLUSTERING_PROCESS	This is the actual clustering job. This job divides facility and item pairs into user-specified clusters that are based on demand in the complete planning horizon. The FASTCLUS procedure is used for clustering. This procedure uses demand data as the basis of clustering and divides the facility and item pairs. Before clustering, outliers (if any) are replaced with moving average values. Also, demand data is normalized to avoid any bias due to large or small values of demand for the facility and item pairs. The pattern of demand over the lifetime of the product is used for clustering and not the actual demand values. The table LTF_Cluster_ART stores the cluster mean values and the table LTF_Cluster_Member_ART stores information about the facility and item pairs that belong to different clusters.	The tables ABT.LTF_Cluster_Param_ABT, ABT.LTF_Cluster_ABT, and Forecast_Clustering (a user-written transform) are required.

Batch Process Jobs for Long-Term Forecasting

The following table provides an overview of the batch process jobs for long-term forecasting. These jobs must be run on a yearly basis. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/C_LTF/B_BATCH_PROCESS

Table A2.8 Batch Process Jobs for Long-Term Forecasting

Job Name	Job Description	Input Requirements
01_LTF DEMAND_DATA_AGGREGATION	This job performs aggregation of demand data for the facility and item pairs that need to be forecasted and stores the data in the ABT.LTF_Demand_Data_ABT table. The facility and item pairs are taken from the short-term forecasting table. If demand data for facility and item pairs is more than 40% of the planning horizon (as defined for the forecast group), then the facility and item pairs are considered for long-term or end-of-life forecasting. Demand for these facility and item pairs is aggregated at a yearly level and then used for forecasting in the subsequent job.	The tables ABT.Facility_Item_Forecast, ABT.STF_FCST_Demand_ABT, and SDL.Facility_Item_Demand are required.
02_LTF_PROCESS	This job runs the actual long-term forecasting process. Regression and Bass models are used for forecasting. The demand pattern for facility and item pairs is regressed against the mean values of all the clusters and clusters with minimum RMSE values are considered for further analysis. The forecast values are calculated based on demand pattern of selected clusters.	The tables ABT.LTF_Demand_Data_ABT, ABT.LTF_Cluster_Param_ABT, and ABT.LTF_Cluster_ART are required.

User Interface Table Creation Jobs

The following table provides an overview of the jobs for creating user interface tables. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/D_UI_TABLES

Table A2.9 *User Interface Table Creation Jobs*

Job Name	Job Description	Input Requirements
01_CREATE_UI_TABLES	This job populates the values of normal, revisited, and low accuracy forecast values in the TSDB.Forecast_Result table. This job presents the results of the quality analysis (QA) process in the format that is required by the user interface of the solution. The job calculates the number of time series that fall in normal, revisited, and low accuracy categories. The job also calculates the number of successor items and the number of time series with insufficient history.	The tables Control.FCST_Control_Data, SDL.Item_Succession, ABT.NPF_Demand_ABT_Less_History are required.
02_CREATE_DEMAND_PROJECTION_DETAIL	This job populates forecasted demand and other important statistics for facility and item pairs under consideration, for the date when the forecasting process was last run. The job updates the tables UIART.Demand_Projection_Detail, UIART.DP_Detail_Normal, UIART.DP_Detail_Revisited, UIART.DP_Detail_Low_Accuracy, and UIART.DP_Detail_Successor.	The following tables are required: • SDL.Facility_Item_Demand • Control.FCST_Control_Data • SDL.Facility • SDL.Item • SDL.Facility_X_Item • SDL.Employee
03_NO_HISTORY_DP_DETAIL	This job populates the actual demand quantity and the forecasted quantity that you provide for the date when the forecasting batch process was last run in the UIART.No_History_DP_Detail table. The table contains actual and predicted values for the facility and item pairs with insufficient available historical demand. HPF procedures are unable to produce forecasts correctly for these pairs.	The following tables are required: • SDL.Facility_Item_Demand • SDL.Facility • SDL.Item • SDL.Facility_X_Item • SDL.Employee

Job Name	Job Description	Input Requirements
04_ONE_TIME_BUY	This job summarizes long-term forecast values for all the facility and item combinations that fulfill long-term forecasting qualification test and populates the UIART.LTF_Forecast_UI_ART table.	The ABT.LTF_Forecast_ART, ABT.LTF_Forecast_Summary_ART tables are required.
05_CREATE_TIMESERIES_DETAIL	This job populates tables UIART.Timeseries_Detail, UIART.Timeseries_Detail_Normal, UIART.Timeseries_Detail_Revisited, UIART.Timeseries_Detail_Low_Accuracy, and UIART.Timeseries_Detail_Successor. The table UIART.Timeseries_Detail contains all the time series for all facility and item combinations. The table UIART.Timeseries_Detail_Normal contains time series details for the facility and item pairs with normal forecast results. Similarly, tables UIART.Timeseries_Detail_Revisited and UIART.Timeseries_Detail_Low_Accuracy contain time series details for the facility and item pairs with revisited and low accuracy forecast results, respectively. The table UIART.Timeseries_Detail_Successor contains time series details for facility and item pairs with items in succession.	The tables ABT.HPF_Forecast_ART and Control.FCST_Control_Data are required.
06_NO_HISTORY_TIMESERIES_DETAIL	This job populates actual and user-predicted demand values for time series for all the facility and item pairs with insufficient demand history in the table TSDB.No_History_Timeseries_Detail.	• ABT.NPF_Demand_ABT_Less_History • SDL.Facility • SDL.Item • SDL.Facility_X_Item • SDL.Employee

Job Name	Job Description	Input Requirements
07_FILTER_ATTRIBUTE_RANGE	This job populates the minimum and maximum values of MAPE for all the forecast result categories (normal, revisited, low accuracy, successor items, and no history) in the table UIART.Filter_Attribute_Range.	The table UIART.Demand_Projection_Detail is required.
08_NO_HIST_FILTER_ATTRIBUTE_RANGE	This job populates the maximum and minimum values of demand quantity, demand error, external demand variance, grouped by forecast groups in the table UIART.No_Hist_Filter_Attribute_Range.	The table UIART.No_History_DP_Detail is required.

Forecasting and Inventory Optimization Integration Jobs

The following table provides an overview of the jobs for integrating forecasting and inventory optimization process. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/E_FORECAST_IO_INTEGRATION

Table A2.10 *Forecasting and Inventory Optimization Integration Jobs*

Job Name	Job Description	Input Requirements
01_AGGREGATE_FCST_ART	This job aggregates data depending on the base period and populates the corresponding ABT (ABT.WK_Aggregated_Forecast_ART, ABT.MTH_Aggregated_Forecast_ART, ABT.QTR_Aggregated_Forecast_ART)	The tables ABT.WK_Forecast_Outfor_ART, ABT.MTH_Forecast_Outfor_ART, and ABT.QTR_Forecast_Outfor_ART are required.

Job Name	Job Description	Input Requirements
02_AGGREGATE_FCST_ART_FROM_NPF	This job aggregates data from the input tables depending on the base period and populates the following tables: • ABT.WK_Aggregated_Forecast_ART • ABT.MTH_Aggregated_Forecast_ART • ABT.QTR_Aggregated_Forecast_ART	The following tables are required: • ABT.NPF_DMD_Less_HIS_AGGR_ABT • TSDB.No_History_Time series_Detail • SDL.Facility_Item_Demand • SDL.Facility_X_Item

Intermittent Demand Job

The following table provides an overview of the job that obtains the intermittent demand. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/FORECASTING/F_INTERMITTENT_DEMAND

Table A2.11 *Intermittent Demand Job*

Job Name	Job Description	Input Requirements
GET_INTERMITTENT_DEMAND	This job identifies the intermittent demand series and retrieves the parameters that are required for the inventory optimization process. The job populates the table ABT.IO_**<BASE_PERIOD>**_IDM_ABT.	The ABT.Forecast_Demand_Data_ABT and ABT.IDM_Facility_Item tables are required.

Inventory Optimization Jobs

Input Data Preparation Jobs

The following table provides an overview of the jobs for preparing data for the inventory optimization process. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/A_CREATE_IO_ABT

Table A2.12 *Input Data Preparation Jobs*

Job Name	Job Description	Input Requirements
01_POPULATE_NODE_DATA	This job populates the tables ABT.IO_**<BASE_PERIOD>**_Node_Data_ABT and ABT.IO_**<BASE_PERIOD>**_Node_Data_Hist_ABT for the inventory optimization process.	The following tables are required: • SDL.Network_Model • SDL.Item • SDL.Item_Price • SDL.Facility_X_Item • SDL.Route • SDL.Network_X_Route • SDL.Route_X_Item • SDL.Facility_Item_X_Network • SDL.Item_Succession • SDL.BOM
02_POPULATE_ARC_DATA	This job populates the arc data in the table ABT.IO_**<BASE_PERIOD>**_ARC_Data_ABT for inventory optimization process.	The following tables are required: • SDL.Network_Model • SDL.Item • SDL.Facility_X_Item • SDL.Route • SDL.Network_X_Route • SDL.Route_X_Item • SDL.Facility_Item_X_Network • SDL.Item_Succession • SDL.BOM

Job Name	Job Description	Input Requirements
03_POPULATE_DEMAND_DATA	This job populates the demand data in the table ABT.IO_**<BASE_PERIOD>**_Demand_Data_ABT for inventory optimization process. If you want to adjust demand data for backorders, then in the user-written code for this job, specify the UseBackOrder parameter to GL_Short_Yes, else specify the parameter to GL_Short_No.	The following tables are required: • SDL.Network_Model • SDL.Item • SDL.Item_Price • SDL.Facility_X_Item • SDL.Facility_Item_X_Network • SDL.Item_Succession • SDL.Back_Order_Summary • SDL.BOM • ABT.**<BASE_PERIOD>**_Aggregated_Forecast_ART • DIM.Time_Dim
04_POPULATE_INVENTORY_DATA	This job populates the inventory data in the tables ABT.IO_**<BASE_PERIOD>**_Inventory_Data_ABT, ABT.IO_**<BASE_PERIOD>**_Opening_Inventory_ABT, and ABT.IO_**<BASE_PERIOD>**_Pipeline_Inventory_ABT for inventory optimization process. If you want to merge the inventory of the predecessor with that of the successor, then in the user-written code for this job, specify the UseItemSuccession parameter to GL_Short_Yes, else specify the parameter to GL_Short_No.	The following tables are required: • SDL.Network_Model • SDL.Item • SDL.Facility_X_Item • SDL.Facility_Item_Inventory • SDL.Pipeline_Inventory • SDL.Item_Succession • SDL.BOM • DIM.Time_Dim

Inventory Optimization Process Jobs

The following table provides an overview of the jobs that are used for running the inventory optimization process. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/B_IO_PROCESS

Table A2.13 *Inventory Optimization Process Jobs*

Job Name	Job Description	Input Requirements
01_POLICY_ORDER_KPI_1	This job runs the inventory optimization process before transshipment and populates the following tables: • ABT.MIRP_**<BASE_PERIOD>**_Node_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_ARC_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_Demand_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_Inventory_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_OPT_Message_ART • ABT.MIRP_**<BASE_PERIOD>**_OUT_Beforetrans	The following tables are required: • ABT.IO_**<BASE_PERIOD>**_Node_Data_ABT • ABT.IO_**<BASE_PERIOD>**_ARC_Data_ABT • ABT.IO_**<BASE_PERIOD>**_Demand_Data_ABT • ABT.IO_**<BASE_PERIOD>**_Inventory_Data_ABT • Control.IO_Batch_Attributes
02_TRANSSHIPMENT	This job runs the transshipment process and populates the following tables: • TRANS_**<BASE_PERIOD>**_Node_Data_ABT • TRANS_**<BASE_PERIOD>**_Nodearc_Data_ABT • TRANS_**<BASE_PERIOD>**_Alertdata_IN_ABT • TRANS_**<BASE_PERIOD>**_Alertdata_Type_ABT • Transshipment_**<BASE_PERIOD>**_ART • Transshipment_Cost_**<BASE_PERIOD>**_ART • Transshipment_**<BASE_PERIOD>**_Summary_ART • TRANS_**<BASE_PERIOD>**_Shipin_Period_Summary • TRANS_**<BASE_PERIOD>**_Shipout_Summary_ART • TRANSSHIP_**<BASE_PERIOD>**_Shipin_Summary	The following tables are required: • ABT.MIRP_**<BASE_PERIOD>**_Node_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_ARC_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_Demand_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_Inventory_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_OUT_Beforetrans

Job Name	Job Description	Input Requirements
03_POLICY_ORDER_KPI_2	This job runs the inventory optimization process after transshipment process and populates the tables ABT.MIRP_**<BASE_PERIOD>**_Predict_KPI_ART. and ABT.MIRP_**<BASE_PERIOD>**_Predict_KPI_HIST_ART	The following tables are required: • ABT.MIRP_**<BASE_PERIOD>**_Node_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_ARC_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_Demand_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_Inventory_Data_ABT • ABT.MIRP_**<BASE_PERIOD>**_OUT_Beforetrans • TRANS_**<BASE_PERIOD>**_Shipin_Period_Summary • TRANS_**<BASE_PERIOD>**_Shipout_Summary_ART

Inventory Optimization UIART Jobs

Attribute Table Creation Jobs

The following table provides an overview of the jobs that create the item and facility attribute tables. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/ATTRIBUTE_JOBS

Table A2.14 *Attribute Table Creation Jobs*

Job Name	Job Description	Input Requirements
FACILITY_ATTR	This job creates the facility attribute table UIART.Facility_ATTR that is used to display attribute details for facilities on the user interface. The job also loads code descriptions for the facilities from the lookup detail table.	The tables DIM.Facility_Dim and SDL.Lookup_Detail are required.

Job Name	Job Description	Input Requirements
ITEM_ATTRIBUTE	This job creates the item attribute table UIART.Item_ATTR that is used to display attribute details for items on the user interface. The job also loads code descriptions for the items from the lookup detail table.	The tables DIM.Item_Dim and SDL.Lookup_Detail are required.

Metric Calculation Jobs

The following table provides an overview of the jobs that calculate the metrics or key performance indicators (KPI). These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

`SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/02_METRIC_CALC`

Table A2.15 *Metric Calculation Jobs*

Job Name	Job Description	Input Requirements
01_GET_LEAD_TIME	This job loads the intermediate UIART tables Get_Lead_Time_ACT_TGT and Get_Lead_Time_Min_Max with the following lead time attributes from SDL and ABT sources: • ACTUAL_LEAD_TIME • TARGET_LEAD_TIME • LEAD_TIME_MAX • LEAD_TIME_MIN	The tables SDL.Purchase_Order, SDL.Receipts, and ABT.MIRP_WK/MTH/QTR_Node_Data_ABT are required.
02_GET_SL_DOWNSTREAM	This job loads the intermediate UIART table SL_Downstream with the following downstream service level attributes from SDL and ABT sources: • ACTUAL_SL_DOWNSTREAM • TARGET_SL_DOWNSTREAM	The tables SDL.Customer_Order, SDL.Dispatch, and ABT.IO_WK/MTH/QTR_Node_Data_ABT are required.

Job Name	Job Description	Input Requirements
03_GET_SL_UPSTREAM	This job loads the intermediate UIART table Get_SL_Upstream with following upstream service level attributes from SDL and ABT sources: • ACTUAL_SL_UPSTREAM • TARGET_SL_UPSTREAM	The tables SDL.Purchase_Order, SDL.Receipts, and ABT.IO_WK/MTH/QTR_Node_Data_ABT are required.
04_GET_DEMAND_ACT	This job loads the intermediate UIART table Get_Demand_ACT with the actual demand from the facility and item demand. The actual demand is calculated as sum of the order quantity from the Facility_Item_Demand table in the required time period.	The tables DIM.Time_Dim and SDL.Facility_Item_Demand are required.
05_GET_DEMAND_TGT	This job loads the intermediate UIART tables TGT_Demand_Temp and Get_Demand_TGT with the target demand from the ART source. The target demand KPI is calculated as the summation of the external and internal demand mean from the output of the MIRP procedure.	The table ABT.MIRP_WK/MTH/QTR_Predict_KPI_ART is required.
06_GET_COST_ACT	This job loads the intermediate UIART table Get_Inventory_Cost_ACT with the actual cost from the facility item inventory and item price sources. The inventory cost is calculated as (Inventory Quantity) * (Item Price Amount).	The tables DIM.Time_Dim, SDL.Facility_Item_Inventory, and SDL.Item_Price are required.
07_GET_COST_TGT	This job loads the intermediate UIART tables TGT_Cost_Temp and Get_Inventory_Cost_TGT with the target inventory cost from the SDL and ART sources. The target inventory cost is calculated as (On-hand Mean) * (Item Price Amount).	The tables ABT.MIRP_WK/MTH/QTR_Predict_KPI_ART and SDL.Item_Price are required.

Job Name	Job Description	Input Requirements
08_GET_KPI_PREDICT _min_max	This job loads the intermediate UIART tables Get_KPI_Min_Max_Period and Get_KPI_Min_Max_History with following upper and lower bound attributes from the intermediate UIART and ART sources: • SL_LB • SL_UB • Demand_UL • Demand_LL	The tables ABT.MIRP_WK/ MTH/QTR_Predict_KPI_ART is required.

Inventory Metrics Periodic Jobs

The following table provides an overview of the jobs that populate the inventory metrics periodic table. These jobs must be run as per the base period. This job runs as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/ 03_METRICS_BASE_PERIOD

Table A2.16 *Inventory Metrics Periodic Jobs*

Job Name	Job Description	Input Requirements
01_IO_METRICS_PERIODIC_ART_HIST	This job integrates all the metric calculations and stores the results in the IO_Metrics_Periodic_ART_HIST table. This table stores the history data, current data, and forecasted data for all periods. The following metrics are loaded in the target history table UIART.IO_Metrics_Periodic_ART_HIST: • LT_ACT • LT_TGT • DEMAND_ACT • DEMAND_FORECAST • SL_UPSTREAM_ACT • SL_UPSTREAM_TGT • SL_DOWNSTREAM_ACT • SL_DOWNSTREAM_TGT • COST_ACT • COST_TGT	The following tables are required: • ABT.IO_WK/MTH/QTR_NODE_DATA_ABT • UIART.GET_LEAD_TIME_act_tgt • UIART.SL_DOWNSTREAM • UIART.GET_SL_UPSTREAM • UIART.GET_DEMAND_ACT • UIART.GET_DEMAND_TGT • UIART.GET_INVENTORY_COST_ACT • UIART.GET_INVENTORY_COST_TGT
02_IO_METRICS_PERIODIC_ART_LOAD	This job loads data in the IO_Metrics_Periodic_ART_Temp table only for the required period in accordance with the global parameter GL_IO_Metrics_History_Period. This parameter specifies the number of periods of historical data to be displayed in the Inventory Metrics workspace.	The table UIART.IO_Metrics_Periodic_ART_HIST is required.

Job Name	Job Description	Input Requirements
03_IO_METRICS_PERIODIC_ART_REPLACE	This job replaces the IO_Metrics_Periodic_ART table by the _temp Periodic ART table that is already loaded. This replacement avoids any lock on the table when a query is issued from the user interface. The following metrics are loaded in the target table: • LT_ACT • LT_TGT • DEMAND_ACT • DEMAND_FORECAST • SL_UPSTREAM_ACT • SL_UPSTREAM_TGT • SL_DOWNSTREAM_ACT • SL_DOWNSTREAM_TGT • COST_ACT • COST_TGT	The table UIART.IO_Metrics_Periodic_ART_Temp is required.

Jobs for the Inventory Metrics View and Alert Settings

The following table provides an overview of the jobs that load the inventory metrics ARTs and the alert settings. These jobs must be run on a daily basis. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/04_METRICS_DAILY

Table A2.17 *Jobs for the Inventory Metrics View and Alert Settings*

Job Name	Job Description	Input Requirements
01_IO_METRICS_ALERT_SETTINGS_LOAD	This job integrates the minimum and maximum values from the intermediate UIART tables to load the IO_Metrics_Alert_Settings_Temp table. The job populates the upper and lower limit values for the required alert metrics. The job loads the following bound values in the UIART.IO_Metrics_Alertsettings_Temp table: • LT_UP_CNTRL_LIM • LT_LW_CNTRL_LIM • DEMAND_UP_CNTRL_LIM • DEMAND_LW_CNTRL_LIM • SL_UPSTREAM_UP_CNTRL_LIM • SL_UPSTREAM_LW_CNTRL_LIM • SL_DOWNSTREAM_UP_CNTRL_LIM • SL_DOWNSTREAM_LW_CNTRL_LIM	The following tables are required: • UIART.GET_LEAD_TIME_min_max • UIART.GET_KPI_min_max_period • ABT.IO_WK/MTH/QTR_NODE_DATA_ABT
02_IO_METRICS_ALERT_SETTINGS_REPLACE	This job replaces the UIART.IO_Metrics_Alertsettings_UIART table that is located on the table server by the corresponding temporary table that is already loaded. This replacement avoids any lock on the table when a query is issued from the user interface. The job only replaces those records in the TSDB.IO_Metrics_Alertsettings table where `spoid = 0` .	The table UIART.IO_Metrics_Alertsettings_Temp is required.

Job Name	Job Description	Input Requirements
03_IO_METRICS_ART_LOAD	This job loads the UIART.IO_Metrics_ART_Temp and UIART.IO_Metrics_ART_No_Vendor_Details tables and calculates the average of the metrics that are taken for the required period from the IO_Metrics_Periodic_ART table. The job also calculates the error high and error low columns by considering the upper control and lower control columns from the TSDB.IO_Metrics_Alertsettings table.	The following tables are required: • DIM.Item_Dim • DIM.Facility_Dim • SDL.Lookup_Detail • UIART.IO_Metrics_ART_Temp • TSDB.IO_Metrics_Alertsettings • SDL.Employee • UIART.IO_Metrics_Periodic_ART
04_IO_METRICS_ART_VENDOR_DETAILS	This job adds vendor details to the IO_Metrics_ART_No_Vendor_Details table to populate the UIART.IO_Metrics_ART_Temp_Vendor_DTLS table.	The following tables are required: • SDL.Facility_Item_X_Network • SDL.Vendor_Facility_Item • UIART.IO_Metrics_ART_No_Vendor_Details • SDL.Vendor
05_IO_METRICS_ART_REPLACE	This job replaces the UIART.IO_Metrics_ART table with the IO_Metrics_ART_Temp_Vendor_DTLS table that is already loaded to avoid any lock on the table when a query is issued from the user interface.	The table UIART.IO_Metrics_ART_Temp_Vendor_DTLS is required.

Job Name	Job Description	Input Requirements
06_IO_METRICS_ITEM_FACITY_NETWORK	This job creates the tables UIART.Item_Category_HIER<n> (Hierarchy tables are created in accordance with the global parameter GL_IO_Item_Cat_LVLNO), UIART.Network_Facility, and UIART.AllFacilities. These tables are required for displaying information in the Inventory Metrics view of the Inventory Analysis workspace. This job calls the following macros. • %IO_METRICS_ITEM_HIER_TABLES • %IO_METRICS_FACILITY_ALL • IO_METRICS_NETWORK_FACILITY The metrics are calculated for all the periods by using the input table.	The table UIART.IO_Metrics_ART is required.

Jobs for Scenario Development Workspace

The following table provides an overview of the job that loads the tables that are required to display information in the Scenario Development workspace. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/05_SCENARIO_BASE_PERIOD

Table A2.18 *The Job for the Scenario Development Workspace*

Job Name	Job Description	Input Requirements
SCENARIO_UI_TABLES	This job loads the following target output tables: • UIART.Facility_Face • UIART.Item_Vendor • UIART.I_X_F_ATTR • UIART.Item_Buyer_List • UIART.Buyers • UIART.Vendors These tables are required by the Scenario Development workspace with the SDL and ABT source tables.	The following tables are required: • SDL.Facility_X_Item • SDL.Facility_Item_X_Network • SDL.Vendor_Facility_Item • SDL.Employee • SDL.Vendor • ABT.IO_WK/MTH/QTR_Node_Data_ABT

Jobs for the Order Suggestions Workspace

The following table provides an overview of the jobs that load the tables that are used to display information in the Order Suggestions workspace. These jobs must be run as per the base period. The specific folder structure for these jobs is as follows:

SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/06_ORDER_BASE_PERIOD

Table A2.19 *Jobs for the Order Suggestions Workspace*

Job Name	Job Description	Input Requirements
01_CREATE_LOCK_PLAN_SETTING	This job creates a table TSDB.Lock_Plan_Setting to store settings that enable automatic locking of the replenishment plans.	The SDL.Facility_X_Item table is required.
02_CREATE_SKU_BUCKETS	This job classifies the facility and item pairs into the following categories for the Order Suggestions workspace: • Normal • Overstock • Primary • Primary and Alternative • Partial • Incomplete The job populates the following tables from the UIART library: • Order_Bucket_Normal_Stock • Order_Bucket_Over_Stock • Order_Bucket_Low_Stock • Order_Bucket_Primary • Order_Bucket_Primary_Alternate • Order_Bucket_Partial • Order_Bucket_Incomplete • Order_Bucket_Type	The input and output tables that are created by the Inventory Optimization process jobs and the following tables are required: • SDL.Lookup_Detail • SDL.Route_Type_Ref • SDL.Vendor • UIART.Item_ATTR • UIART.I_X_F_ATTR • UIART.Facility_ATTR • UIART.Buyers • TSDB.IO_Metrics_Alertsettings

Job Name	Job Description	Input Requirements
03_POPULATE_DRP_TABLE	This job populates the DRP tables UIART.Repl_By_Period_Normal, UIART.Repl_By_Period_Overstock, and UIART.Repl_By_Period_Lowstock for all the facility and item pairs to be displayed on the Order Suggestions workspace.	The input and output tables that are created by the Inventory Optimization Process jobs and the following tables are required: • SDL.Lookup_Detail • SDL.Route_Type_Ref • SDL.Vendor • UIART.Item_ATTR • UIART.I_X_F_ATTR • UIART.Facility_ATTR • UIART.Buyers • TSDB.IO_Metrics_Allsettings
04_POPULATE_ORDER_RESULT	This job populates the UIART.Order_Result table that is required for displaying information on the Order Suggestions workspace.	
05_POPULATE_REPL_BY_PERIOD_LOOKUP	This job populates the UIART_Repl_By_Period_Lookup lookup table that is required for displaying information on the Order Suggestions workspace.	
06_POPULATE_ORDER_DETAIL	This job populates the TSDB_Order_Detail table with details of each order.	
07_POPULATE_REPL_PLAN_DETAIL	This job populates the TSDB.Repl_Plan_Detail table with details of each replenishment plan.	
08_POPULATE_REPL_PLAN_METRICS	This job populates the UIART.Repl_Plan_Metrics table with different metrics such as projected transfer cost, projected holding cost, projected penalty cost, and so on, for displaying information on the Order Suggestions workspace.	
09_CREATE_REPL_PLAN_SUMMARY	This job creates replenishment plans summary table TSDB.Repl_Plan_Summary that is displayed on the Order Suggestions workspace.	

Job for Loading Order Details in the Stageout Library

The following table provides an overview of the job that loads the order details in the Stageout library. These jobs must be run on a daily basis. The specific folder structure for these jobs is as follows:

`SPO_JOBS/02_ANALYTICS_JOBS/IO/C_UI_ART_JOBS/07_ORDERS_DAILY`

Table A2.20 *Job for Loading the Order Details*

Job Name	Job Description	Input Requirements
LOAD_ORDER_DETAILS	This job loads the order details in the table Stageout.Order_Detail.	The TSDB.Order_Detail table is required.

Appendix 3

Metric Calculations

Calculations of Inventory Metrics and Thresholds

Lead Time

Overview

Lead time is the time to physically deliver an order (full or partial) from the source location to the destination location. The different values of lead time are explained in the following subsections.

Actual Lead Time

Actual lead time is the transit time between the primary supplying channel and the receiving location. An average of lead times is calculated over the period under consideration.

The derivation for the actual lead time is as follows:

Actual lead time = Average (Receipt date – Dispatch date) for the order of a facility and item pair, considering it is a regular order in the primary network.

Consider the order receipts between the start date and end date of the previous period. If there is no delivery during that period, then the actual lead time is missing. Actual lead time values are missing for current and future periods also.

Target Lead Time

Target lead time is the average transportation time that is measured over the past periods between the source and the destination facilities through primary channels. This value is obtained from the source system and is available in the node data.

Lead Time Minimum

Lead time minimum or lead time lower control limit is the minimum time (in days) for an item to be transported to a facility. This value is obtained from the source system and is available in the node data.

Lead Time Maximum

Lead time maximum or lead time upper control limit is the maximum time (in days) for an item to be transported to a facility. This value is obtained from the source system and is available in the node data.

Lead Time Error High

If the lead time actual value is greater than the lead time maximum value, then lead time error high = 1, else the value is 0.

Lead Time Error Low

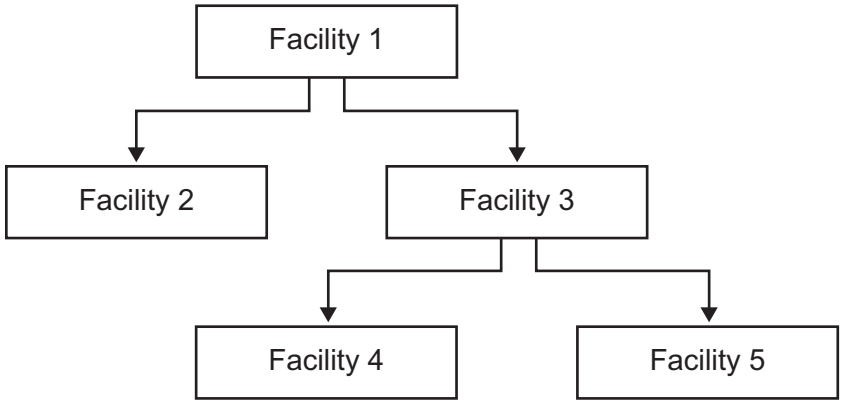
If the lead time actual value is less than the lead time minimum value, then lead time error low = 1, else the value is 0.

Downstream Service Level

Overview

Service level is a measure of the fulfillment of customer demand. Downstream service level is the service level provided by a facility to the downstream facilities or end customers. For example, consider the network in the following figure:

Figure A3.1 *A Sample Network*



The downstream service level for Facility 3 is the service level that is delivered to Facility 4 and Facility 5.

Actual Downstream Service Level

Actual downstream service level is the percentage service level that is calculated from the solution data layer for the facility. The average is calculated over the past period under consideration and is missing for the current and future periods.

The derivation for the actual downstream service level value is as follows:

- Consider all regular orders for an item in a facility where the delivery due date exists in the previous period. Receipt control type (possible values are LOOSE or TIGHT) is used to determine the period boundaries to calculate service level.

    - For facility and item pairs with loose receipt control type, the receipt date must be before or on the period end date of the previous period.
    - For facility and item pairs with tight receipt control type, the receipt date must be before or on the receipt due date.
- To calculate the downstream service level value for facility and item pairs for the period under consideration:
    - If the dispatch quantity is greater than or equal to the customer order quantity, then the downstream service level value = 1
    - If the dispatch quantity is less than the customer order quantity, then the downstream service level value equals the ratio of the dispatch quantity and customer order quantity.
- The actual downstream service level with customer-specific service type value is calculated as follows:
    - If service type = fill rate (FR), then actual service level = downstream service level value as calculated in earlier point.
    - If service type = ready rate (RR), and if downstream service level value = 1, then actual service level = 1; else for any other value of downstream service level, the actual service level = 0.
    - If service type = backorder ratio (BR), then actual service level = 1 - (downstream service level value)

Target Downstream Service Level

The target downstream service level is available in the solution data layer.

Downstream Service Level Upper Control Limit

The downstream service level upper control limit is required for setting alerts on the user interface of the solution. This value (also called upper bound for service level) is customizable.

The derivation for the downstream service level upper control limit is as follows:

Downstream service level upper control limit = (Service level value supplied by the source system) + (Service level threshold specified in the global parameter list)

Downstream Service Level Lower Control Limit

The downstream service level lower control limit is required for setting alerts on the user interface of the solution. This value (also called lower bound for service level) is customizable.

The derivation for the downstream service level lower control limit is as follows:

Downstream service level lower control limit = (Service level value supplied by the source system) - (Service level threshold specified in the global parameter list)

Downstream Service Level Error High

If the downstream service level actual value is greater than the downstream service level upper control limit, then the downstream service level error high value is stored as 1, else 0.

Downstream Service Level Error Low

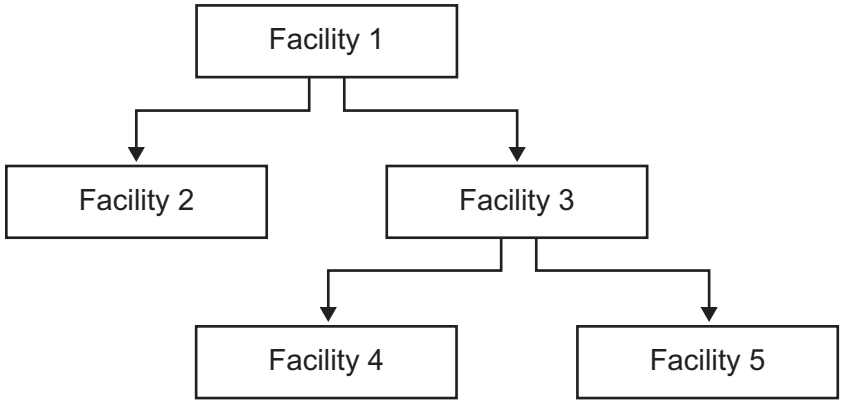
If the downstream service level actual value is less than the downstream service level lower control limit, then the downstream service level error low value is stored as 1, else 0.

Upstream Service Level

Overview

Service level is a measure of the fulfillment of customer demand. Service level value that is obtained from the supplying facility or external suppliers is stated as upstream service level. For example, consider the network in the following figure:

Figure A3.2 *A Sample Network*



The upstream service level for Facility 2 and Facility 3 is the service level that is obtained from Facility 1.

Actual Upstream Service Level

Actual upstream service level is the percentage service level that is calculated from the solution data layer for the facility. The average is calculated over the past period under consideration and is missing for the current and future periods.

The derivation for the actual upstream service level is as follows:

- Consider all regular orders for an item in a facility where the delivery due date exists between the start date and end date of the previous period. Receipt control type (possible values are LOOSE or TIGHT) is used to determine the period boundaries to calculate service level.
  - For facility and item pairs with loose receipt control type, the receipt date must be before or on the end date of the previous period.
  - For facility and item pairs with tight receipt control type, the receipt date must be before or on the receipt due date.
- The upstream service level value is calculated for facility and item pairs for the period under consideration as follows:
  - If the received quantity is greater than or equal to the purchase order quantity, then the upstream service level value = 1.
  - If the received quantity is less than the purchase order quantity, then the upstream service level value equals the ratio of the received quantity and purchase order quantity.
- The actual upstream service level with customer-specific service type value is calculated as follows:

- If service type = fill rate (FR), then actual service level = upstream service level value as calculated in the earlier point.
- If service type = ready rate (RR) and if upstream service level value = 1, then actual service level = 1; else for any other value of upstream service level, the actual service level = 0.
- If service type = backorder ratio (BR), then actual service level = 1 - (upstream service level value)

Target Upstream Service Level

Target upstream service level is obtained from the solution data layer.

Upstream Service Level Upper Control Limit

The upstream service level upper control limit is required for setting alerts on the user interface of the solution. This value (also called upper bound for service level) is customizable.

The derivation for the upstream service level upper control limit is as follows:

Upstream service level upper control limit = (Service level value supplied by the source system) + (Service level threshold specified in the global parameter list)

Upstream Service Level Lower Control Limit

The upstream service level lower control limit is required for setting alerts on the user interface of the solution. This value (also called lower bound for service level) is customizable.

The derivation for the upstream service level lower control limit is as follows:

Upstream service level lower control limit = (Service level value supplied by the source system) - (Service level threshold specified in the global parameter list)

Upstream Service Level Error High

If the upstream service level actual value is greater than the upstream service level upper control limit, then the upstream service level error high value is stored as 1, else 0.

Upstream Service Level Error Low

If the upstream service level actual value is less than the upstream service level lower control limit, then the upstream service level error low value is stored as 1, else 0.

Demand

Overview

Demand of an item at a facility is the quantity of that item that is ordered at that facility. Demand is defined as orders for the amount of a particular service part that a consumer or facility wants to purchase at a given price. Demand for an item or a service is determined by many different factors other than price, such as the price of substitute goods, complementary goods, and so on.

Actual Demand

Actual demand is the sum of the total order quantity that is received over the period under consideration for a facility and item pair. This demand value is calculated for all facility and item pairs from the source table FACILITY_ITEM_DEMAND. Demand values are not available for current and future periods.

The derivation for the actual demand is as follows:

Actual demand = Sum (Order quantity over the period under consideration)

Target Demand

Target demand is the sum of the total order quantity that is forecasted over the period under consideration. This demand value is calculated by using the output of the MIRP procedure, that is external demand mean and internal demand mean.

The derivation for the target demand is as follows:

Target demand = (External demand mean) + (Internal demand mean)

Demand Upper Limit

Demand upper limit (also called demand upper bound) is derived from the output of the MIRP procedure for the period under consideration. The upper bound value is calculated by using the customer-specific global parameter GL_DEMAND_THRESHOLD_COEFF. This global parameter specifies the coefficient value (a positive number < 3) that is used to calculate the lower and upper bounds for forecasted demand.

The derivation for the demand upper limit is as follows:

Demand upper limit = (EXTERNAL_DEMAND_MEAN +INTERNAL_DEMAND_MEAN) + (GL_DEMAND_THRESHOLD_COEFF. * (SQRT(EXTERNAL_DEMAND_VAR + INTERNAL_DEMAND_VAR)))

Demand Lower Limit

Demand lower limit (also called demand lower bound) is derived from the output of the MIRP procedure predict KPI for the period under consideration. The lower bound value is calculated by using the customer-specific global parameter GL_DEMAND_THRESHOLD_COEFF. This global parameter specifies the coefficient value (a positive number < 3) that is used to calculate the lower and upper bounds for forecasted demand.

The derivation for the demand lower limit is as follows:

Demand lower limit = (External demand mean + Internal demand mean) - (GL_DEMAND_THRESHOLD_COEFF * (SQRT(External demand variance + Internal demand variance)))

Demand Error High

If the actual demand value is greater than the demand upper control limit, then value for the demand error high is stored as 1, else 0.

Demand Error Low

If the actual demand value is less than the demand lower control limit, then value for the demand error low is stored as 1, else 0.

Inventory Cost

Overview

Inventory cost is defined as the cost of the closing inventory for the previous period. This value is the total aggregated cost of the inventory for all facility and item pairs.

Actual Inventory Cost

Actual inventory cost is an average cost that is calculated over the period under consideration.

The derivation for the actual inventory cost is as follows:

Actual inventory cost = (Average of closing inventory quantity in the period) * (Unit cost (for the previous periods) taken from item price)

Target Inventory Cost

Target inventory cost is the predicted or forecasted inventory cost for the current and future periods.

The derivation for the target inventory cost is as follows:

Target inventory cost = (On-hand mean * Unit cost) where on-hand mean is calculated by the MIRP procedure and the unit cost is used from the item price.

Calculations of Replenishment Plan Metrics

The replenishment plan metrics are displayed in the Order Suggestions workspace.

Here is a list of the metrics and their calculations:

Total Order Amount

Total order amount = Total amount of suggested orders from primary channel and alternative channels

Total Number of Orders

Total number of orders = Total number of orders from primary channel and alternative channels, where suggested order quantity > 0

Total Projected Cost

Total projected cost = Projected transfer cost + Projected holding cost + Projected penalty cost

Projected Holding Cost

Projected holding cost = (Projected on-hand after lateral transshipment + Projected future delivery) * Unit holding cost

Projected Transfer Cost

Projected transfer cost = Fixed ordering cost + Total transfer cost from primary channel and alternate channels

Projected Penalty Cost

Projected penalty cost = Inventory shortage * Unit penalty cost

Projected Service Level

Projected service level is based on the demand during projection intervals (DDPI).

DDPI = Lead time + Period between replenishments (PBR), if PBR > 1 and policy type is base stock

For other conditions, DDPI = Lead time + 1

If echelon = 1:

Projected service level = Average of the service level over the period DDPI, where the service level is not missing

If echelon > 1:

Projected service level = Average of the service level over the period DDPI excluding period 1, where the service level is not missing

Target Service Level

Target service level = Service level value as specified in the SDL

Service Level Upper Bound

Service level upper bound = Upper threshold value for the service level at a facility and item pair

Service Level Lower Bound

Service level lower bound = Lower threshold value for the service level at a facility and item pair

Calculations of the Scenario-Specific Metrics

The scenario-specific metrics are displayed in the Scenario Development workspace.

The following table lists the metrics, their calculations, and the scenario types where the metrics are used.

Table A3.1 *Scenario-Specific Metric Calculations*

Metric	Metric Calculation	Associated Scenario Types
Inventory units	Inventory units = Order-up-to level	• Customer-facing facility analysis • Internal facility service level analysis • Ad hoc analysis
Inventory Cost	Inventory cost = Order-up-to level * Item price	Customer-facing facility analysis
On-Hand Cost	On-hand cost = On-hand mean * Item price	• Internal facility service level analysis • Ad hoc analysis
On-Hand Holding Cost	On-hand holding cost = On-hand mean * Holding cost amount	• Internal facility service level analysis • Ad hoc analysis
Pipeline Cost	Pipeline cost = Pipeline mean * Pipeline cost amount	Ad hoc analysis
Total Cost	Total cost = On-hand cost + On-hand holding cost + Pipeline cost	• Service level sensitivity analysis • Internal facility service level analysis • Ad hoc analysis

Glossary

backorder ratio
the ratio of average backorders divided by average demand.

base period
the interval of time in which one inventory replenishment order is allowed.

clustering
the process of dividing a data set into mutually exclusive groups such that the observations for each group are as close as possible to one another, and different groups are as far as possible from one another.

control table
a table containing parameter values that are used for the forecast and inventory optimization analyses. These values are customizable.

fill rate
a service measure that indicates the fraction of demand that is satisfied from on-hand inventory.

holding cost
the cost of keeping items in inventory, which includes the expense that is incurred in running a warehouse, handling inventory, and counting inventory. Holding costs might also include the cost of special storage requirements, deterioration of stock, damage, theft, obsolescence, insurance, taxes, or the opportunity cost of money invested. Also called carrying cost.

MAPE
See mean absolute percent error.

mean absolute percent error
the average of the absolute percentage errors. Short form: MAPE.

multi-echelon network
the distribution network that has at least one facility and item pair with more than one echelon level. The echelon level of a facility and item pair represents its relative position in a network. The echelon level of a pair is equal to the maximum echelon level of all its successor facility and item pairs plus one. If a facility and item pair does not have successors, its echelon level is one.

order-up-to level
the target inventory level.

penalty cost
the cost that is incurred when a stockout occurs. This cost might include the cost of emergency shipments, the cost of substituting a less profitable item, or the cost of lost goodwill.

planning horizon
the number of periods into the future for which predictions are made.

policy type
the type of replenishment policy that is used to manage inventory. Policies help in determining when and in what quantities, orders should be placed to restock inventory. SAS Service Parts Optimization supports two policy types¬Base-stock policy (BS) and Min-max policy (SS).

ready rate
the probability that the on-hand inventory level at the end of a review time period is positive.

reorder level
the inventory level at which a replenishment order should be placed.

RMSE
See root mean square error.

root mean square error
the square root of the mean square error. It is used as an estimate of the standard deviation of the response variable. Short form: RMSE.

solution data layer
an intermediate layer of tables provided by the solution to save the customer source data in the required manner.

Index

Your Turn

We welcome your feedback.

- If you have comments about this book, please send them to **yourturn@sas.com**. Include the full title and page numbers (if applicable).
- If you have comments about the software, please send them to **suggest@sas.com**.

SAS® Publishing Delivers!

Whether you are new to the work force or an experienced professional, you need to distinguish yourself in this rapidly changing and competitive job market. SAS® Publishing provides you with a wide range of resources to help you set yourself apart. Visit us online at support.sas.com/bookstore.

SAS® Press

Need to learn the basics? Struggling with a programming problem? You'll find the expert answers that you need in example-rich books from SAS Press. Written by experienced SAS professionals from around the world, SAS Press books deliver real-world insights on a broad range of topics for all skill levels.

support.sas.com/saspress

SAS® Documentation

To successfully implement applications using SAS software, companies in every industry and on every continent all turn to the one source for accurate, timely, and reliable information: SAS documentation. We currently produce the following types of reference documentation to improve your work experience:

- Online help that is built into the software.
- Tutorials that are integrated into the product.
- Reference documentation delivered in HTML and PDF – **free** on the Web.
- Hard-copy books.

support.sas.com/publishing

SAS® Publishing News

Subscribe to SAS Publishing News to receive up-to-date information about all new SAS titles, author podcasts, and new Web site features via e-mail. Complete instructions on how to subscribe, as well as access to past issues, are available at our Web site.

support.sas.com/spn

sas | THE POWER TO KNOW®