

# **Sterling Business Intelligence™**

---

## **Implementation Guide**

**Release 8.5**

*Last Updated HF5*

**May 2011**

**Sterling Commerce**  
An IBM Company

© Copyright 2009 Sterling Commerce, Inc. All rights reserved.

Additional copyright information is located on the documentation library:  
<http://www.sterlingcommerce.com/Documentation/MCSF85/CopyrightPage.htm>

---

# Contents

Getting Started . . . . .	6
Intended Audience . . . . .	6
Implementing Sterling Business Intelligence. . . . .	6
Configure Sterling Business Intelligence . . . . .	7
Configure the Staging Project . . . . .	7
Define the Hold Types . . . . .	7
Configure the Data Mart Project . . . . .	8
Configure the Install Locale . . . . .	8
Define a Catalog . . . . .	8
Configure the Sterling Business Intelligence Application . . . . .	9
Map Address to a Region . . . . .	9
Convert Long Zip Code to Short Zip Code . . . . .	9
Configure the Purge Criteria . . . . .	9
Order Timestamp Tag Purge . . . . .	10
Shipment Timestamp Tag Purge . . . . .	11
Purge for Staging Tables . . . . .	13
Load Management. . . . .	14
Staging Database . . . . .	14
Data Mart Database . . . . .	14
Change Data Capture Filters . . . . .	14
Load Interval . . . . .	15
Generate the Staging Project Package File . . . . .	15
Define a DB-Alias File . . . . .	15
Staging Entity Metadata File. . . . .	16
Generate the Staging Project Package File . . . . .	17
Load Data Into Data Mart . . . . .	18
Create a Backup Database . . . . .	19
Initial Data Load . . . . .	20
Load Initial Data in Sterling Business Intelligence for the Staging Project . . . . .	20
Load Initial Data in Sterling Business Intelligence for the Data Mart Project . . . . .	24
Incremental Data Load . . . . .	26
Load Incremental Data in Sterling Business Intelligence for Staging Project. . . . .	26
Load Incremental Data in Sterling Business Intelligence for Data Mart Project . . . . .	29
Purge the Staging Tables . . . . .	30
Enhancing the Performance of the Sterling Business Intelligence Dashboards and Reports By Using Aggregate Fact Tables . . . . .	32
Extend and Customize Sterling Business Intelligence . . . . .	34
Extend the Adaptive Warehouse Content. . . . .	34
Extend the Adaptive Analytics Reports. . . . .	35

Extend the Dashboards . . . . .	35
Customize the User Interface . . . . .	36
Create a New Screen . . . . .	36
Customize the Staging Project . . . . .	40
Add a New Entity . . . . .	41
Add Filters to an Existing Entity . . . . .	44
Configure Load Partitions . . . . .	45
Deploy the Customization . . . . .	46
The Sterling Business Intelligence Model . . . . .	47
Warehouse Model Star Schemas . . . . .	47
SO Line Star Schema . . . . .	47
Shipment Line Star Schemas . . . . .	49
SO Payment Star Schema . . . . .	50
SO Hold Star Schema . . . . .	51
SO Modification Star Schema . . . . .	52
SO Modification Summary Star Schema . . . . .	53
SO Payment Summary Star Schema . . . . .	54
SO Hold Summary Star Schema . . . . .	55
SO Daily Category Summary Star Schema . . . . .	56
SO Daily Region Summary Star Schema . . . . .	57
Shipment Summary Star Schema . . . . .	58
SO Daily Summary Star Schema . . . . .	59
SO Customer Monthly Summary Star Schema . . . . .	60
SO Monthly Summary Star Schema . . . . .	61
Target Model Star Schemas . . . . .	62
SO Line Star Schema Grouping . . . . .	62
Shipment Line Star Schema Grouping . . . . .	64
. . . . .	65
SO Payment Star Schema Grouping . . . . .	66
SO Hold Star Schema Grouping . . . . .	67
SO Modification Star Schema Grouping . . . . .	67
SO Modification Summary Star Schema Grouping . . . . .	68
SO Payment Summary Star Schema Grouping . . . . .	68
SO Hold Summary Star Schema Grouping . . . . .	69
SO Daily Category Summary Star Schema Grouping . . . . .	70
SO Daily Region Summary Star Schema Grouping . . . . .	71
Shipment Summary Star Schema Grouping . . . . .	71
SO Daily Summary Star Schema Grouping . . . . .	72
SO Customer Monthly Summary Star Schema Grouping . . . . .	75
SO Monthly Summary . . . . .	76
Target Model Dimensions . . . . .	77
Enterprise . . . . .	77
Ship Node . . . . .	77
Seller . . . . .	78
Buyer . . . . .	78
Payment Type . . . . .	78
Hold Type . . . . .	79
Channel . . . . .	79
Fulfillment Type . . . . .	79
Delivery Method . . . . .	80
Product . . . . .	80

Product Category Hierarchy . . . . .	81
Customer . . . . .	82
Customer Hierarchy . . . . .	82
Customer Contact. . . . .	83
Address Region . . . . .	83
Address Region Hierarchy . . . . .	84
Document Type . . . . .	84
User Ranges. . . . .	85
Calendar . . . . .	85
Parameter Maps . . . . .	87
Troubleshooting. . . . .	88
Load Management Process is Stalled. . . . .	88
Resolving the Problem . . . . .	88
Report Execution Displays an RSV-BBP-0022 Error . . . . .	88
Resolving the Problem . . . . .	88
Schema Not Displayed When Creating an Adaptive Warehouse Project. . . . .	89
Resolving the Problem . . . . .	89
ETL Script Running in CLI Stalls in an IBM AIX Environment. . . . .	89
ETL Process Does Not Load All the Data into the Data Mart Database. . . . .	89
Resolving the Problem . . . . .	89
<b>Index</b>	<b>90</b>

---

---

## Getting Started

The Sterling Business Intelligence solution can be extended and customized to meet specific business requirements. Sterling Business Intelligence provides a predefined set of content, including the Adaptive Warehouse model, Adaptive Analytics reports, and dashboards for sales order business processes. You can analyze your business requirements and customize each of this content according to individual requirements.

## Intended Audience

This document is intended for use by application developers, analytic system administrators, and analytic users who implement the Sterling Business Intelligence solution.

## Implementing Sterling Business Intelligence

Ensure that you have installed Sterling Business Intelligence, and that the content is deployed as per the instructions provided in the *Sterling Business Intelligence Installation Guide*.

Use the following process to perform the implementation of the Sterling Business Intelligence solution:

1. Review the reports and dashboards provided in Sterling Business Intelligence and understand whether they meet your business requirements. For more information about the reports and dashboards, refer to the *Sterling Business Intelligence User Guide*.
2. Make appropriate configuration changes to the model. For more information about configuring Sterling Business Intelligence, refer to the topic “[Configure Sterling Business Intelligence](#)”.
3. Perform the initial data load, which helps load the existing data into the data mart database. For more information about initial data load, refer to the topic “[Load Management](#)”.
4. Perform incremental data load, which helps load the transactional data into the data mart database. For more information about incremental data load, refer to the topic “[Load Management](#)”.
5. Extend and customize Sterling Business Intelligence to meet your business requirements. For more information about extending and customizing, refer to the topic “[Extend and Customize Sterling Business Intelligence](#)”.

---

## Configure Sterling Business Intelligence

You can configure Sterling Business Intelligence to restrict the data that is loaded into the data warehouse from the online transaction processing (OLTP) system.

Prior to configuring Sterling Business Intelligence, ensure that you have performed all the steps mentioned in the *Sterling Business Intelligence Installation Guide*. For details about setting up the Adaptive Warehouse Content, refer to the *Sterling Business Intelligence Installation Guide*.

To configure Sterling Business Intelligence, perform the following tasks:

- ◆ [Configure the Data Mart Project](#)
- ◆ [Configure the Sterling Business Intelligence Application](#)
- ◆ [Configure the Purge Criteria](#)

### Configure the Data Mart Project

After setting up the Adaptive Warehouse Content, perform the following configurations to restrict the data that is loaded into the data mart schema.

#### Configure the Install Locale

To configure the install locale, perform the following tasks:

1. Launch the IBM<sup>®</sup> Cognos<sup>®</sup> Adaptive Warehouse application.
2. Open Sterling Data Source project.
3. Click the **Source** tab.
4. Click **Parameter Maps** and select **Locale Code Map**.

The Parameter Map Definition screen is displayed.

5. Change the values for the `Country` and `Language` keys. By default, `US` and `en` are the values respectively for the `Country` and `Language` keys. For example, if your base locale is `ja_JP`, the `Country` and `Language` values must be changed to `ja` and `JP` respectively.
6. Click the **Warehouse** tab.
7. Click **Save**.

#### Define a Catalog

You must determine a catalog for analytical purpose. Only items belonging to this catalog will be considered for analysis.

**Note:** When an item belongs to multiple categories in a catalog, the item is displayed as a part of one category only.

To define a catalog, perform the following tasks:

1. Launch the IBM Cognos Adaptive Warehouse application.

2. Open Sterling Data Source project.
3. Click the **Source** tab.
4. Navigate to **Source > Product Category Hierarchy source > Product Category Hierarchy Product Category Tree Input Source**.
5. Double-click **Category Domain Filter**.  
The Filter Definition screen is displayed.
6. Click the **Parameter** tab.
7. Select the required catalog code from **Category Domain Map**.
8. Click **OK**.
9. Click the **Warehouse** tab.
10. Double-click the **Product Category Hierarchy** dimension.
11. Click the **Dataflow** tab.  
For each dataflow, apply the category domain filter.
12. Click **Save**.

## Configure the Sterling Business Intelligence Application

After setting up the Adaptive Warehouse Content, perform follow the additional configurations to restrict the data that is loaded into the staging schema.

### Map Address to a Region

You must define a region schema for analytical purpose. This region schema is used to determine the region associated with the customer's address and to perform analysis based on regions.

To define a region schema for analytics purpose, perform the following tasks:

1. Launch Application Manager.
  1. From the **Application Rules Side** panel, select **Analytics > Region Usage For Analytics**.  
The Region Usage For Analytics window is displayed.
  2. From the **Schema for Analytics** panel, select the region schema, you want to use for the data warehouse reports.
  3. Click **Save**.

### Convert Long Zip Code to Short Zip Code

Short zip code is used to obtain the best match region for a customer's address. It is a simplified format of an address' zip code. Short zip code regex expression defined for a country is used to convert long zip codes to a simplified format. You can also implement `YCPGetShortZipCodeUE` user exit to convert long zip code into short zip code.

If you have installed Sterling Multi-Channel Fulfillment Solution, Release 8.5, in upgrade mode, you must follow the instructions provided in the section "Business Intelligence Enhancements: Best Match Region"



in the *Selling and Fulfillment Foundation: Upgrade Guide*. If you have installed Sterling Multi-Channel Fulfillment Solution, Release 8.5, in the nonupgrade mode, refer to the section “Update Best Match Region” in the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

## **Configure the Purge Criteria**

Purge transactions determine when a record has to be purged, by determining the current date and subtracting the retention days specified in the purge criteria. If the timestamp on the table is less than or equal to the value of current days minus retention days, the record is purged.

You can define purge criteria rules for data purges not related to specific document types. Purge is the process by which old data is removed from the system database. A purge minimizes the number of unused database records to increase search efficiency and reduce the size of the required physical disk.

For more information about configuring a document’s purge criteria, refer to the *Sterling Distributed Order Management: Configuration Guide*.

## Order Timestamp Tag Purge

The Order Timestamp Tag (YFS\_ORDER\_TS\_TAG) table maintains a record of each order that is added or updated in the online transaction processing (OLTP) system. The ETL\_TS column of this table is used as a Change Data Capture filter for all the order-related dataflows. Based on the data volumes, this table can grow rapidly. To avoid this, a purge agent is provided to purge the records that are older than the configured retention days. The records are purged only if the data has been loaded into the staging system.

### Purge Criteria

The following table displays the purge rule for the Order Timestamp Tag Purge:

Rule	Description	Retention Days
SBIORDERTAGTS	Order Tag Timestamp Purge	7

### Attributes

The following table displays the attributes for the Order Timestamp Tag Purge transaction:

Attribute	Value
Base Transaction ID	ORDER_TS_TAG_PRG
Agent Criteria ID	ORDERTSTAGPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

### Criteria Parameters

The following table displays the criteria parameters for the Order Timestamp Tag Purge transaction:

Criteria Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information about the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
EnterpriseCode	Optional. The organization for which the Alert Purge has to be run. If it is not passed, all the enterprises are monitored.

Criteria Parameter	Description
Live	Optional. Mode in which to run. Valid values are: Y - Deletes qualifying records from the YFS_ORDER_TS_TAG table. N - Determines the rows that are deleted from the YFS_ORDER_TS_TAG table.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria. Purge code is SBIORDERTAGTS.

### Statistics Tracked

None.

### Events Raised

None.

### Tables Purged

YFS\_ORDER\_TS\_TAG.

## Shipment Timestamp Tag Purge

The Shipment Timestamp Tag (YFS\_SHIPMENT\_TS\_TAG) table maintains a record of the orders that are shipped. The ETL\_TS column of this table is used as a CDC filter for all the shipment-related dataflows. Based on the data volumes, this table can grow rapidly.

### Attributes

The following table displays the attributes for the Shipment Timestamp Tag Purge transaction:

Attribute	Value
Base Transaction ID	SHIPMENT_TS_TAG_PRG
Agent Criteria ID	SHIPMENTTSTAGPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs Called	None
User Exits Called	YFSBeforePurgeUE

## Purge Criteria

The following table displays the purge rule for the Shipment Tag Timestamp Purge:

Rule	Description	Retention Days
SBISHIPMENTTAGTS	Shipment Tag Timestamp Purge	7

## Criteria Parameters

The following table displays the criteria parameters for the Shipment Timestamp Tag Purge transaction:

Criteria Parameter	Description
Action	Required. Triggers the transaction. If left blank, it defaults to <i>Get</i> , the only valid value.
CollectPendingJobs	If this parameter is set to <i>N</i> , the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the System Management Console.
Number of RecordsTo Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 ( <i>zero</i> ), it defaults to 5000.
EnterpriseCode	Optional. The organization for which the <i>Alert Purge</i> has to be run. If it is not passed, all the enterprises are monitored.
Live	Optional. Mode in which to run. Valid values are: Y - Default value. Moves qualifying records from the regular tables listed under <i>Tables Purged</i> to the corresponding history tables. N - Test mode. Determines the rows that are being purged without actually purging them.
PurgeCode	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the <i>PurgeCode</i> used in <i>Business Rules Purge Criteria</i> . Purge Code is <i>SBISHIPMENTTAGTS</i> .

## Statistics Tracked

None

## Events Raised

None.

## Tables Purged

YFS\_SHIPMENT\_TS\_TAG

## Purge for Staging Tables

The Staging database is used to stage the data before loading it into the data mart database. To keep the staging tables manageable, timely clean up has to be carried out. The purge is based on retention days, and only records older than the retention days are deleted. Retention days should be set based on the volumes of data and the frequency of the incremental load. To ensure that only data that is no longer useful is cleaned up, the data is cleaned only if the ETL job of the Sterling Data Source project has successfully run within the last 'n' days. Purging of transaction tables pertaining to YFS\_ORDER\_HEADER is done only when the order is in Closed status. Master data and configuration data is not cleaned up unless you manually run load management, requesting a refresh of the master data and the configuration data.

To run the purge for staging tables, perform the following tasks:

1. Rename `<AC_INSTALL_DIR>/bin/dbconnection.properties.sample` as `<AC_INSTALL_DIR>/bin/dbconnection.properties`.
2. Configure the `dbconnection.properties` file. Set the following properties:

```
db.driver=<path of the db driver jar>
purge.days=<Retention Days>
jdbc.driver.class=<Db Driver Class Name>
db.vendor=<Db Vendor Name>
staging.jdbc.url= <JDBC URL of staging DB>
staging.jdbc.catalog= <DB Name of staging DB>
staging.jdbc.user= <User Name of staging DB>
staging.jdbc.password=<Password of staging DB>
```

```
target.jdbc.url=<JDBC URL of datamart DB>
target.jdbc.catalog=<DB Name of datamart DB>
target.jdbc.user=<User Name of datamart DB>
target.jdbc.password=<Password of datamart DB>
```

3. Edit the `purge_staging.cmd` file and set the path to `<AC_INSTALL_DIR>`.
4. From the Analytics Content Installation directory, `<AC_INSTALL_DIR>/bin`, run the following command:

```
purge_staging.cmd (For Windows)
purge_staging.sh (For UNIX)
```

**Note:** This command deletes the records from the staging database. You will, therefore, not be able to retrieve these records.

---

## Load Management

Load management extracts, transforms, and loads (ETL) data from the online transaction processing (OLTP) database and other data sources into your data warehouse. Load management synchronizes the source model and target model to the warehouse model and updates the data mart database to reflect the changes in the warehouse model. Load management also publishes the source and target packages to the IBM® Cognos® 8 portal. You can use load management differently, depending on the environment that you work in.

**Note:** The ETL process may not load some data into the data mart if the system time of the Cognos server is ahead of the OLTP database server. To avoid this, prior to starting the ETL process, you must ensure that the time in the Cognos Server is set to at least one minute slower than the OLTP database server. In case the platform considerations do not allow for time syncing, the Cognos server should run one minute slower than the OLTP server.

## Staging Database

Staging serves as a gateway for integrating data from multiple schemas, and delivering the same to the data mart database. Staging is required for an ETL process to deliver functionally correct data with the least impact on the source system. The advantages of the staging process are as follows:

- ◆ Staging reduces the load on the production server by ensuring that the incremental data is first copied to a staging database, and then transforms are applied from the staging database to the data mart database.
- ◆ In a multischema mode, staging helps obtain data from multiple colonies into a single data mart database.

## Data Mart Database

Sterling Data Mart database is a dimensionally modeled relational database. The reports in Sterling Business Intelligence are based on this database.

The Sterling Data Source project extracts data from the staging database and performs the necessary transformations and loads them into the warehouse database. This project contains all the metric calculations and star schema definitions. The Sterling Data Source project contains dataflows that extract data such as Configuration, Master, and Transaction.

## Change Data Capture Filters

Change Data Capture (CDC) filters determine the updates to your OLTP source system from one load to another. You can set a date range to control the amount of data that is loaded.

CDC filters determine the updates to your data warehouse from one load to another. You can set a date range to control the amount of data that is loaded. Let us for example, consider that you have previously loaded data for the period 2000-2006, and now want to load only the data pertaining to 2007. You must create a CDC filter to add only the new or changed data and avoid reloading data from 2000-2006.

## Load Interval

You can set a load interval for a CDC filter. Specifying a load interval reduces the amount of data being read and processed by the Adaptive Warehouse load engine at one time by dividing the data into a number of smaller batches. This ensures that the server does not run out of memory and that the load operation completes successfully.

For example, for the initial load of your data warehouse, will probably have historical data from many years that you want to load. Calculate the number of batches for the filter by dividing the difference between the From and To dates by the number of load intervals. The Adaptive Warehouse load engine creates a maximum of 99 batches. Based on the volume of the data, you must find a balance between the batch limit and the memory capacity of your machine.

The load interval is measured in days for loading a fact with a change date. For example, if the date range for a CDC filter is from January 1, 2000, to June 15, 2007, and you specify a load interval of 30. When load management is run, multiple batches are generated by the Adaptive Warehouse load engine. Each batch consists of 30 days. The first batch extracts data from January 1, 2000, to January 30, 2000, the second batch extracts data from January 31, 2000, to March 1, 2000, and so on. In this example, the Adaptive Warehouse load engine generates 90 batches.

For more information about load management and optimizing performance, refer to the *IBM® Cognos® 8 Analytic Applications Installation and Configuration Guide* and the *IBM® Cognos® 8 Analytic Applications Adaptive Warehouse User Guide*.

## Generate the Staging Project Package File

To load data from the OLTP database to the Staging database, you require the Staging Project Package file that can be generated by the Staging Project Package file generation tool. To run the Staging Project Package file generation tool, you must perform the following tasks:

1. [Define a DB-Alias File](#)
2. [Staging Entity Metadata File](#)
3. [Generate the Staging Project Package File](#)

The Staging Project Package file generation tool uses the data source connection information from the DB-Alias file and stores it in the generated Staging Project Package file. The Staging Entity Metadata file contains information about the entities required to be loaded into staging database. The tool gets the entity information such as entity name, supported table types of entity, columns associated with the entity, extended columns, and so on from the entity repository. An ETL job takes the required information from the Staging Project Package file to load the data into the staging database.

**Note:** Ensure that you perform all these tasks from the server in which Selling and Fulfillment Foundation is installed.

### Define a DB-Alias File

A DB-Alias file provides information about the data source connections.

To define a DB-Alias file:

1. Create the `<db_alias>.txt` file.

2. Add the connection information for the required data sources in <db\_alias>.txt file. The syntax for a connection string is as follows:

```
<PRODUCTION> '<data_source>' <catdb_driver> '<catdb_items>'
```

- ◆ <data\_source> is the name of the data source. For staging and datamart schemas, the data source names should be STERLING\_STAGING\_DATA\_SOURCE and STERLING\_DATA\_SOURCE respectively.

You must start all the transaction schemas data source names with TRANSACTION and the Configuration schema names with CONFIGURATION. Ensure that the complete value for this variable is defined in single quotes. It is assumed that master and transaction schemas are shared. Therefore, it is not required to define data source details of master data schema.

**Note:** In a multischema deployment, you must define the data source information of transaction schemas in all the colonies that are required to be loaded.

- ◆ <catdb\_driver> specifies the database driver used to connect to the catalog containing the Job Stream. The value of this property depends on the database being used.

The database name and its corresponding <catdb\_driver> values are as follows:

- ◆ Oracle: ORACLE
- ◆ DB2: DB2
- ◆ Microsoft SQL: OLEDB
- ◆ <catdb\_items> specifies the connection string that is required to connect to the data source. Ensure that the value for this variable is defined in single quotes.

The database name and its corresponding <catdb\_items> values are as follows:

- ◆ Oracle: '<userid>/<password>@<SID>'
- ◆ DB2: '<DSN=<database>;UID=<userid>;PWD=<password>'
- ◆ Microsoft SQL:  
'<User Id=<userid>;Password=<password>;Database=<database>;  
Data Source=<server\_name>;'

A sample DB-Alias file is as follows:

```
<PRODUCTION> 'STERLING_STAGING_DATA_SOURCE' ORACLE  
'staging_test/staging_test@ORCL'  
  
<PRODUCTION> 'STERLING_DATA_SOURCE' ORACLE 'target_test/staging_test@ORCL'  
  
<PRODUCTION> 'CONFIGURATION-1' ORACLE 'oltp_config/oltp-config@ORCL'  
  
<PRODUCTION> 'TRANSACTION-1' ORACLE 'oltp_colony_1/oltp_colony_1@ORCL'  
  
<PRODUCTION> 'TRANSACTION-2' ORACLE 'oltp_colony_2/oltp_colony_2@ORCL'
```

## Staging Entity Metadata File

The Staging Entity Metadata file contains information about the following:

- ◆ Entities that need to be loaded into the staging database.
- ◆ Entity information such as table name, table type, and so on.



- ◆ Relationship between the transaction entities, the driver tables, and filters.

The data is loaded into the staging database from all the columns of an entity including the extended columns. You should not change this file. If you want to add new entities, you need to define the extended Staging Entity Metadata file before generating the Staging Project Package file. For more information about customizing the staging project, refer to the [Extend and Customize Sterling Business Intelligence](#) topic.

## Generate the Staging Project Package File

Execute the Staging Project Package file generation tool to generate the Staging Project Package file. This tool gets the required information from the DB-Alias file, Staging metadata entity file, and the entity repository.

To generate the Staging Project Package file:

1. Navigate to the <INSTALL\_DIR>/bin folder.

Here, <INSTALL\_DIR> is the directory in which the Selling and Fulfillment Foundation is installed.

2. Set or export (set for Windows and export for Linux/UNIX) the value of the ANT\_OPTS environment variable depending on your operating system as described in the following table:

Operating System	Command Parameters
Solaris 10	-Xmx1024m -Xms512m -XX:MaxPermSize=256m
Windows 2008, Windows XP	-Xmx1024m -Xms512m -XX:MaxPermSize=256m
IBM AIX 6.1 SP10	-Xmx1024m -Xms512m
HP-UX 11i v3	-Xmx1024m -Xms512m -XX:MaxPermSize=256m
Red Hat Enterprise Linux	-Xmx1024m -Xms512m -XX:MaxPermSize=256m

For example, in a Red Hat Enterprise Linux environment, set the ANT\_OPTS environment variable as follows:

```
export ANT_OPTS=-Xmx1024m -Xms512m -XX:MaxPermSize=256m
```

3. Execute either of the following commands:

- ◆ To generate the initial load package file:

For Windows

```
sci_ant.cmd -f generate_staging_package.xml initial  
-Dfile_name=<package_file_name> -Ddb_alias=<db_alias_file_path>
```

For Linux or UNIX

```
sci_ant.sh -f generate_staging_package.xml initial  
-Dfile_name=<package_file_name> -Ddb_alias=<db_alias_file_path>
```

Here, <package\_file\_name> is the name of the generated Staging Project Package file. If you do not pass any argument, the Staging Project Package file is generated with `Staging_Initial_Load_Package.pkg` as the name.

<db\_alias\_file\_location> is the path in which the DB-Alias file is defined. This argument is mandatory.

- ◆ To generate the incremental load package file:

For Windows

```
sci_ant.cmd -f generate_staging_package.xml incremental
-Dfile_name=<package_file_name> -Ddb_alias=<db_alias_file_path>
```

For Linux or UNIX

```
sci_ant.sh -f generate_staging_package.xml incremental
-Dfile_name=<package_file_name> -Ddb_alias=<db_alias_file_path>
```

Here, <package\_file\_name> is the name of the generated Staging Project Package file. If you do not pass any argument, the Staging Project Package file is generated with `Staging_Incremental_Load_Package.pkg` as the name.

<db\_alias\_file\_path> is the path in which the DB-Alias file is defined. This argument is mandatory.

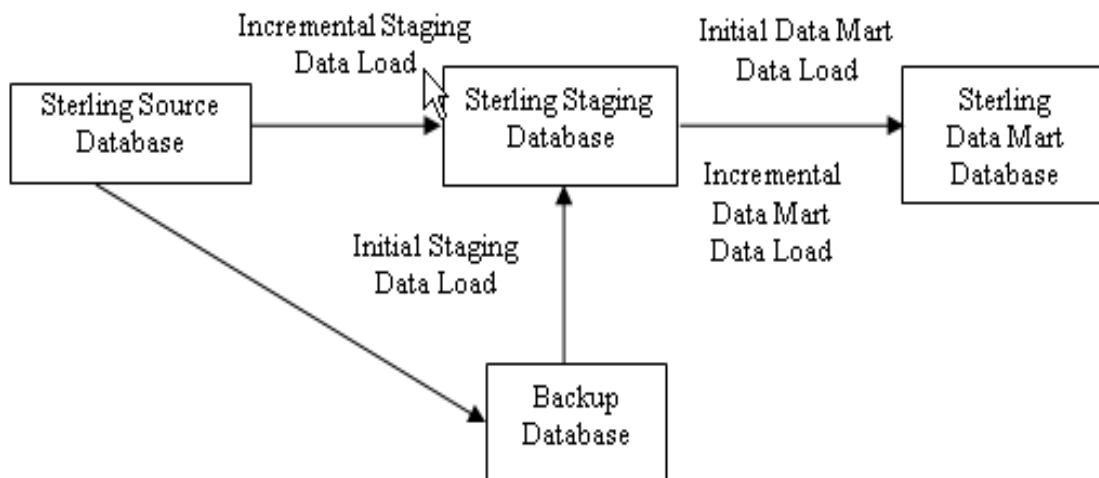
The package files generated are located in the <INSTALL\_DIR>/bin folder, where <INSTALL\_DIR> is directory in which the Sterling and Fulfillment Foundation is installed.

For more information about load management, refer to the following:

- ◆ [Load Initial Data in Sterling Business Intelligence for the Staging Project](#)
- ◆ [Load Incremental Data in Sterling Business Intelligence for Staging Project](#)

## Load Data Into Data Mart

The load management process in Sterling Business Intelligence is illustrated here:



1. Back up the OLTP source database before you go live (typically two weeks earlier) with Sterling Business Intelligence.
2. Initial data load from the backup database to the Sterling staging database
3. Initial data load from the Sterling staging database to the Sterling data mart database
4. Incremental data load from the Sterling source database to the Sterling staging database
5. Incremental data load from the Sterling staging database to the Sterling data mart database

### Create a Backup Database

It is recommended that you take a backup of your OLTP source database along with the history data as required (typically, 2 years) before you go live with Sterling Business Intelligence. Restore this backup into a different database and use this to perform initial data load.

For more information about taking a backup and restoring your database, refer to your database vendor documentation.

## Initial Data Load

Initial load dataflows enable you to load existing data into the staging database. These dataflows load data from transaction tables and their history tables. You can set properties such as CDC filters and Load Interval, to improve the performance of the ETL process that occurs during load management. Initial data load is used to bring data into the data warehouse the first time.

Because Initial Data Load is a long-running and memory-intensive process, you must control the amount of data that is loaded in order to avoid running into out-of-memory exceptions. This is done using the CDC filters and Load Interval properties.

**Note:** When an order is purged, the quantity in YFS\_ORDER\_RELEASE\_STATUS record is set to zero. These records are not considered in the Volume Analysis metrics calculations.

## Load Initial Data in Sterling Business Intelligence for the Staging Project

The staging project loads the data from the following configuration, master, and transaction tables:

### ◆ Configuration Tables

- ◆ YFS\_ORGANIZATION
- ◆ YFS\_COMMON\_CODE
- ◆ YFS\_PAYMENT\_TYPE
- ◆ YFS\_HOLD\_TYPE
- ◆ YFS\_STATUS\_MODIFICATION\_TYPE
- ◆ YFS\_PERSON\_INFO\_CONFIG
- ◆ YFS\_REGION
- ◆ YFS\_DOCUMENT\_PARAMS
- ◆ YFS\_LOCALIZED\_STRINGS
- ◆ YFS\_REGION\_BEST\_MATCH
- ◆ YFS\_CATEGORY\_DOMAIN
- ◆ YFS\_REGION\_SCHEMA
- ◆ YFS\_REGION\_USAGE
- ◆ YFS\_CAT\_DOMAIN\_LOCALE
- ◆ YFS\_DATA\_SECURITY\_GROUP
- ◆ YFS\_CHARGE\_CATEGORY

### ◆ Master Tables

- ◆ YFS\_ITEM
- ◆ YFS\_CATEGORY
- ◆ YFS\_CATEGORY\_ITEM
- ◆ YFS\_CUSTOMER

- ◆ YFS\_CONSUMER
- ◆ YFS\_CUSTOMER\_CONTACT
- ◆ YFS\_USER
- ◆ YFS\_USER\_GROUP\_LIST
- ◆ YFS\_ITEM\_LOCALE
- ◆ YFS\_CATEGORY\_LOCALE
- ◆ **Transaction Tables**
  - ◆ YFS\_ORDER\_HEADER
  - ◆ YFS\_ORDER\_LINE
  - ◆ YFS\_ORDER\_RELEASE\_STATUS
  - ◆ YFS\_ORDER\_DATE
  - ◆ YFS\_ORDER\_LINE\_SCHEDULE
  - ◆ YFS\_CHARGE\_TRANSACTION
  - ◆ YFS\_PAYMENT
  - ◆ YFS\_ORDER\_HOLD\_TYPE
  - ◆ YFS\_ORDER\_HOLD\_TYPE\_LOG
  - ◆ YFS\_ORDER\_AUDIT
  - ◆ YFS\_ORDER\_AUDIT\_LEVEL
  - ◆ YFS\_SHIPMENT
  - ◆ YFS\_SHIPMENT\_LINE
  - ◆ YFS\_OPPORTUNITY
  - ◆ YFS\_PERSON\_INFO
  - ◆ YFS\_LINE\_CHARGES

To load the initial data in Sterling Business Intelligence for the staging project, perform the following steps:

1. Copy the generated initial load Staging Project Package file from the <INSTALL\_DIR>/bin folder to the SBI Adaptive Analytics Runtime (server 6) in the Sterling Business Intelligence application ecosystem.

Here, <INSTALL\_DIR> is the directory in which the Selling and Fulfillment Foundation is installed.

2. For Windows, copy the staging\_load\_management.cmd file from the <AC\_INSTALL\_DIR>/bin/ folder to the SBI Adaptive Analytics Runtime (server 6) in the Sterling Business Intelligence application ecosystem.

For Linux or UNIX, copy the staging\_load\_management.sh file from the <AC\_INSTALL\_DIR>/bin/ folder to the he SBI Adaptive Analytics Runtime (server 6) in the Sterling Business Intelligence

application ecosystem. After copying the file, provide execute permission to the `staging_load_management.sh` file.

**Notes:**

- ◆ If Oracle is the database for your datamart and you are using a 64 bit client on UNIX, before running the command, ensure that the 32 bit client libraries are present in the shared libraries path you provide.

Set the environment variables as follows:

- ◆ Set the value of the `COGNOS_AW_HOME` environment variable to the install directory of Adaptive Analytics Runtime.
- ◆ Set the value of the `DB_ALIAS_DRIVER` environment variable to the database name.

The database name and its corresponding values are as follows:

- ◆ Oracle: ORACLE
- ◆ DB2: DB2
- ◆ Microsoft SQL: OLEDB
- ◆ Set the value of the `DB_ALIAS_ITEM` environment variable to the connection string information of Sterling Staging database. This value must be passed in double quotes.

Database name and its corresponding values are as follows:

- ◆ Oracle: "`<userid>/<password>@<SID>`"
- ◆ DB2: "`DSN=<database>;UID=<userid>;PWD=<password>`"
- ◆ Microsoft SQL:  
"`User Id=<userid>;Password=<password>;Database=<database>;  
Data Source=<server_name>;`"
- ◆ Set the value of the `LOG_DIR` environment variable to the directory in which the ETL log files are stored. The default location of the ETL log files is `<AW_INSTALL_DIR>\datamanager\log`.

3. To create the staging catalog tables, DDL, and CDC tables, run the following commands respectively:

For Windows

```
staging_load_management.cmd setup <initial_load_package_file>
```

For Linux or UNIX

```
staging_load_management.sh setup <initial_load_package_file>
```

Here, the `<initial_load_package_file>` variable indicates the path and the name of Sterling Staging Initial Package file.

4. To load data from all the configuration and master tables, run the following command:

For Windows

```
staging_load_management.cmd Configuration_Master <initial_load_package_file path>
```

For Linux or UNIX

```
staging_load_management.sh Configuration_Master <initial_load_package_file path>
```

5. To load data from the transaction tables, run the following command:

For Windows

```
staging_load_management.cmd Initial_Load_Bulk_Data  
<initial_load_package_file path> <start_date> <end_date> <table_name>
```

For Linux or UNIX

```
staging_load_management.sh Initial_Load_Bulk_Data  
<initial_load_package_file path> <start_date> <end_date> <table_name>
```

**Note:** Here <table\_name> is an optional parameter. If you do not add <table\_name> parameter to the commands above, then data from all the transaction tables are loaded.

Also, pass the values for the <start\_date> and <end\_date> variables in the

'YYYY-MM-DD' format. <start\_date> and <end\_dates> are the dates in which the initial load data is distributed. For example, if the initial load data is distributed from 01/01/2007 to 12/31/2009 then run the following command to load initial data:

```
staging_load_management.cmd/sh Initial_Load_Bulk_Data  
<initial_load_package_file path> '2007-01-01' '2009-12-31'
```

**Note:** You can view the log files in the

<AW\_INSTALL\_DIR>\datamanager\log\<etl\_execution\_time> directory to find if the initial data load job stream is successful. If any build stream fails to execute, you must truncate the corresponding table in the staging database and execute the build stream again. To execute a build stream, run the following command:

```
staging_load_management.cmd/sh Initial_Load_Bulk_Data  
<initial_load_package_file path> <start_date> <end_dates> <table_name>
```

## Load Initial Data in Sterling Business Intelligence for the Data Mart Project

The data mart project contains dataflows that extract data from the configuration, master, and the transaction tables. Following are the dataflows pertaining to initial data load for the data mart project:

◆ Dataflows that extract configuration data and master data are:

- ◆ Enterprise Dataflow
- ◆ Ship Node Dataflow
- ◆ Seller Dataflow
- ◆ Buyer Dataflow
- ◆ Payment Type Dataflow
- ◆ Hold Type Dataflow
- ◆ Channel Dataflow
- ◆ Fulfillment Dataflow
- ◆ Delivery Method Dataflow
- ◆ Product Dataflow
- ◆ Product Category Tree Dataflow
- ◆ Modification Type Dataflow
- ◆ Customer Hierarchy Dataflow
- ◆ Customer Contact Dataflow
- ◆ Address Dataflow
- ◆ Address Region Hierarchy Dataflow
- ◆ Document Type Dataflow
- ◆ User Ranges Dataflow
- ◆ Time Period Dataflow

◆ Dataflows that extract transaction data are:

- ◆ SO Line Initial Load Dataflow
- ◆ Shipment Line Initial Load Dataflow
- ◆ SO Payment Initial Load Dataflow
- ◆ SO Hold Initial Load Dataflow
- ◆ SO Modification Initial Load Dataflow

To load the initial data in Sterling Business Intelligence for the data mart project, perform the following steps:

1. Launch the IBM Cognos Adaptive Warehouse application.
2. Open the Sterling Data Source project.



3. From the **Actions** menu, select **Load Management**.  
The Load Management screen is displayed with the available dataflows.
4. Select the required initial dataflow and click **CDC Filters**.  
The Change Data Capture Parameters screen is displayed.
5. Enter the time period range in the **From Value** field and **To Value** field to load data in each filter item.
6. Repeat steps 3 to 5, for the other required dataflows.
7. Click **Close**.
8. Click **Save**.
9. From the **Actions** menu, select **Load Management**.  
The Load Management screen is displayed with all the available dataflows.
10. Click **Deselect All**.
11. Select all the initial dataflows to load the transaction data, for example, SO Payment Initial Load Dataflow.  
**Note:** When you are loading data for the first time, you must also select the dataflows that load the configuration data and master data. At the time of loading, whenever there is a change in the configuration data and the master data, select the dataflows that load the configuration data and the master data.
12. Click **Script**.  
The Script dialog box is displayed.
13. Enter `\cognos\ap\bin\ in the Script Location field.`
14. Click **Save**.
15. Click **OK**.
16. Click **Close**.

**Notes:**

- ◆ If the IBM Cognos Adaptive Application Framework (AAF) Runtime and Cognos Adaptive Application Framework (AAF) applications are installed on different servers, copy the generated script to the appropriate location.

17. To run the load management scripts, execute the following command from the `\cognos\ap\bin` folder:

```
pwexec.cmd <path\filename>.xml (For Windows)
```

```
pwexec.sh <path\filename>.xml (For UNIX)
```

- Note:** It is recommended that you truncate all the transaction tables in the staging database before starting the incremental load.

## Incremental Data Load

Incremental data load refers to changes in the OLTP source system. Every change in the OLTP source system must be reflected in the data mart system, and occurs through incremental load (basically, synchronizing the data between the OLTP source system and the data mart system). You must also add the configuration data and master data because they do not change frequently. It is recommended that the incremental load scripts are run as scheduled jobs, and the configuration data and master data are reloaded periodically.

**Note:** The CDC filters for the incremental data load use the **ETL\_TS** column in the YFS\_ORDER\_TS\_TAG table or the YFS\_SHIPMENT\_TS\_TAG table.

## Load Incremental Data in Sterling Business Intelligence for Staging Project

The staging project contains tables that extract data from the configuration, master, and transaction tables. For incremental data load, the tables that extract the configuration data and the master data are the same as that of the initial data load. The tables that extract transaction data are:

- ◆ YFS\_ORDER\_HEADER
- ◆ YFS\_ORDER\_LINE
- ◆ YFS\_ORDER\_RELEASE\_STATUS
- ◆ YFS\_ORDER\_DATE
- ◆ YFS\_ORDER\_LINE\_SCHEDULE
- ◆ YFS\_CHARGE\_TRANSACTION
- ◆ YFS\_PAYMENT
- ◆ YFS\_ORDER\_HOLD\_TYPE
- ◆ YFS\_ORDER\_HOLD\_TYPE\_LOG
- ◆ YFS\_ORDER\_AUDIT
- ◆ YFS\_ORDER\_AUDIT\_LEVEL
- ◆ YFS\_SHIPMENT
- ◆ YFS\_SHIPMENT\_LINE
- ◆ YFS\_OPPORTUNITY
- ◆ YFS\_ORDER\_TS\_TAG
- ◆ YFS\_SHIPMENT\_TS\_TAG
- ◆ YFS\_PERSON\_INFO
- ◆ YFS\_LINE\_CHARGES
- ◆ YFS\_OPPORTUNITY\_TS\_TAG

**Note:** Ensure that the Staging database schema is refreshed after loading initial load data into staging and datamart database and before performing the following task.

To load the incremental data in Sterling Business Intelligence for the staging project, perform the following steps:

1. Copy the generated incremental load Staging Project Package file from the <INSTALL\_DIR>/bin folder to the SBI Adaptive Analytics Runtime (server 6) in the Sterling Business Intelligence application ecosystem.

Here, <INSTALL\_DIR> is the directory in which the Selling and Fulfillment Foundation is installed.

2. For Windows, copy the staging\_load\_management.cmd file from the <AC\_INSTALL\_DIR>/bin/ folder to the SBI Adaptive Analytics Runtime (server 6) in the Sterling Business Intelligence application ecosystem.

For Linux or UNIX, copy the staging\_load\_management.sh file from the <AC\_INSTALL\_DIR>/bin/ folder to the he SBI Adaptive Analytics Runtime (server 6) in the Sterling Business Intelligence application ecosystem. After copying the file, provide execute permission to the staging\_load\_management.sh file.

**Notes:**

- ◆ If Oracle is the database for your datamart and you are using a 64 bit client on UNIX, before running the command, ensure that the 32 bit client libraries are present in the shared libraries path you provide.

Set the environment variables as follows:

- ◆ Set the value of the COGNOS\_AW\_HOME environment variable to the install directory of Adaptive Analytics Runtime.
- ◆ Set the value of the DB\_ALIAS\_DRIVER environment variable to the database name.

The database name and its corresponding values are as follows:

- ◆ Oracle: ORACLE
- ◆ DB2: DB2
- ◆ Microsoft SQL: OLEDB
- ◆ Set the value of the DB\_ALIAS\_ITEM environment variable to the connection string information of Sterling Staging database. This value must be passed in double quotes.

Database name and its corresponding values are as follows:

- ◆ Oracle: "<userid>/<password>@<SID>"
- ◆ DB2: "DSN=<database>;UID=<userid>;PWD=<password>"
- ◆ Microsoft SQL:  
"User Id=<userid>;Password=<password>;Database=<database>;  
Data Source=<server\_name>;"
- ◆ Set the value of the LOG\_DIR environment variable to the directory in which the ETL log files are stored. The default location of the ETL log files is <AW\_INSTALL\_DIR>\datamanager\log.

3. To create the staging catalog tables, DDL, and CDC tables, run the following commands:

For Windows

```
staging_load_management.cmd setup <incremental _load_package_file>
```

For Linux or UNIX

```
staging_load_management.sh setup <incremental _load_package_file>
```

Here, the <incremental \_load\_package\_file> variable indicates the path and name of the Sterling Staging Incremental Package file.

4. You can load data from the configuration and master tables at the same time. Alternatively, you can also choose to load data from the configuration and master tables one after the other.

- ◆ To load data from the configuration and master tables at the same time, run the following command:

For Windows

```
staging_load_management.cmd Configuration_Master  
<incremental_load_package_file path>
```

For Linux or UNIX

```
staging_load_management.sh Configuration_Master  
<incremental_load_package_file path>
```

- ◆ To load data from configuration tables only, run the following command:

For Windows

```
staging_load_management.cmd Configuration <incremental_load_package_file  
path>
```

For Linux or UNIX

```
staging_load_management.sh Configuration <incremental_load_package_file  
path>
```

- ◆ To load data from master tables only, run the following command:

For Windows

```
staging_load_management.cmd Master <incremental_load_package_file path>
```

For Linux or UNIX

```
staging_load_management.sh Master <incremental_load_package_file path>
```

5. To load the incremental data, run the following command:

For Windows

```
staging_load_management.cmd ALL_Incremental <incremental_load_package_file  
path>
```

For Linux or UNIX

```
staging_load_management.sh ALL_Incremental <incremental_load_package_file  
path>
```

## Load Incremental Data in Sterling Business Intelligence for Data Mart Project

In the incremental data load process for the data mart project, the dataflows that extract the configuration data and the master data are the same as that of the initial data load. Following are the dataflows that extract transaction data pertaining to incremental data load for the data mart project:

- ◆ SO Line Dataflow
- ◆ Shipment Line Dataflow
- ◆ SO Payment Dataflow
- ◆ SO Hold Dataflow
- ◆ SO Modification Dataflow

To load the incremental data in Sterling Business Intelligence for the data mart project, perform the following steps:

1. Launch the IBM Cognos Adaptive Warehouse application.
2. Open the Sterling Data Source project.
3. From the **Actions** menu, select **Load Management**.

The Load Management screen is displayed with all the available dataflows.

4. Click **Deselect All**.
5. Select all noninitial dataflows. The noninitial dataflows have names that do not end with Initial Dataflow, for example, SO Line Dataflow.

### Notes:

- ◆ When you are loading data for the first time, you must also select the dataflows that load the configuration data and master data. At the time of loading, whenever there is a change in the configuration data and the master data, select the dataflows that load the configuration data and the master data.
- ◆ In a single schema deployment, do not select the YFS\_PERSON\_INFO\_CONFIG Dataflow.

6. Click **Script**.

The Script dialog box is displayed.

7. Enter `\cognos\ap\bin\ in the Script Location field and click Save.`
8. Click **OK**.
9. Click **Close**.

### Notes:

- ◆ If the IBM Cognos Adaptive Application Framework (AAF) Runtime and Cognos Adaptive Application Framework (AAF) applications are installed on different servers, copy the generated script to the appropriate location.

10. To run the load management scripts, execute the following command from the `\cognos\ap\bin` folder:

```
pwexec.cmd <path\filename>.xml (For Windows)
```

```
pwexec.sh <path\filename>.xml (For UNIX)
```

## Purge the Staging Tables

The Staging database is used to stage the data before loading it into the data mart database. To keep the staging tables manageable, timely clean up has to be carried out. The purge is based on retention days, and only records older than the retention days are deleted. Retention days should be set based on the volumes of data and the frequency of the incremental load. To ensure that only data that is no longer useful is cleaned up, the data is cleaned only if the ETL job of the Sterling Data Source project has successfully run within the last 'n' days. Purging of transaction tables pertaining to YFS\_ORDER\_HEADER is done only when the order is in Closed status.

**Note:** The command to purge the staging tables deletes the records from the staging database permanently.

To purge the staging tables, perform the following tasks:

1. In the SBI Adaptive Analytics Runtime (server 6) of the Sterling Business Intelligence application ecosystem, navigate to the folder in which the `staging_load_management.cmd/.sh` file is copied. Ensure that the environment variables are set as follows:

- ◆ Set the value of the `COGNOS_AW_HOME` environment variable to the install directory of Adaptive Analytics Runtime.
- ◆ Set the value of the `DB_ALIAS_DRIVER` environment variable to the database name.

The database name and its corresponding values are as follows:

- ◆ Oracle: ORACLE
- ◆ DB2: DB2
- ◆ Microsoft SQL: OLEDB
- ◆ Set the value of the `DB_ALIAS_ITEM` environment variable to the connection string information of Sterling Staging database. This value must be passed in double quotes.

Database name and its corresponding values are as follows:

- ◆ Oracle: "`<userid>/<password>@<SID>`"
- ◆ DB2: "`DSN=<database>;UID=<userid>;PWD=<password>`"
- ◆ Microsoft SQL:  
"`User Id=<userid>;Password=<password>;Database=<database>;  
Data Source=<server_name>;`"
- ◆ Set the value of the `LOG_DIR` environment variable to the directory in which the ETL log files are stored. The default location of the ETL log files is `<AW_INSTALL_DIR>\datamanager\log`.

2. To purge data from the transaction tables in the staging database, run the following command:

For Windows:

```
staging_load_management.cmd Purge <incremental_load_package_file>  
<retention_days>
```

For Linux or UNIX:

```
staging_load_management.sh Purge <incremental_load_package_file>  
<retention_days>
```

Here, the `<retention_days>` variable indicates the number of days for which you want to retain the data in staging. For example, if the value of this variable is set to 7, the records older than 7 days are considered for purge.

---

## Enhancing the Performance of the Sterling Business Intelligence Dashboards and Reports By Using Aggregate Fact Tables

Sterling Business Intelligence provides various dashboards and reports in which the metrics are associated with a time period. These metrics are stored in fact tables that are located at the centre of the corresponding star schemas. The fact tables are stored at the lowest level of granularity in a data mart. A report takes a long period of time to fetch data from the fact tables because the calculations and summarization of data is performed when executing the report. Because of this, the Sterling Business Intelligence user interface takes more time to display the dashboards and reports. To enhance the performance of the dashboards and reports, the pre-summarized or pre-aggregated data should be stored at a higher level of granularity. For example, Sterling Business Intelligence provides out-of-the-box aggregate fact tables that contain new metrics derived either from one or more aggregate functions such as COUNT, MIN, MAX, and so on. The new metrics in an aggregate fact table are stored and maintained in special fact tables at the grain of aggregation in the data mart. You can create an aggregate table by performing the following steps when summarizing a fact table:

◆ Excluding one or more dimensions. For example, the SO Payment Summary aggregate fact table is derived from the SO Payment fact table by excluding certain dimensions. The SO Payment fact table consists of the following dimensions:

- ◆ Channel
- ◆ Calendar
- ◆ Payment Type
- ◆ Enterprise
- ◆ Seller
- ◆ Buyer
- ◆ Document Type

The SO Payment Summary aggregate fact table consists of the following dimensions:

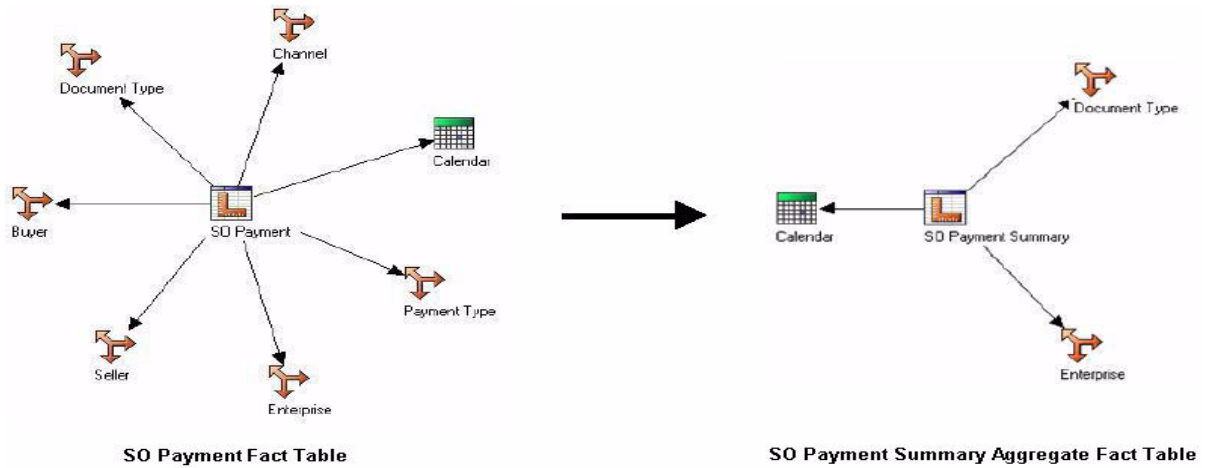
- ◆ Enterprise
- ◆ Document Type
- ◆ Calendar

◆ Replacing one or more dimensions by their shrunken or rolled versions. For example, the SO Customer Monthly Summary aggregate fact table is created by replacing the dimensions by their shrunken and rolled versions. The daily-grain dimensions in the SO Daily Summary aggregate fact table have been replaced by the monthly-grain dimensions in the SO Customer Monthly Summary aggregate fact table.

Though maintaining aggregate fact tables in a data mart increases the time taken to complete an ETL process, but the time taken to display the dashboards and reports is remarkably decreased. For example, the SO Payment Summary aggregate fact table has been created from the SO Payment fact table. Because of this, the report execution time for the Percentage Authorization Failure and Percentage Settlement Failure metrics in the Perfect Order dashboard has been reduced approximately by a factor of 10.



The following figure illustrates the SO Payment fact table being aggregated to the SO Payment Summary aggregate fact table:



Based on your business requirements, you can analyze the metrics required by a dashboard or report and the existing grain to create aggregate fact tables to enhance the performance of the dashboard or report. When creating an aggregate fact table, consider the following:

- ◆ Number of records: As compared to a fact table, decrease the total number of records in an aggregate table by ten to twenty times.
- ◆ Storage: As compared to a fact table, increase the size of a record at the most by a factor of two.
- ◆ Granularity: Select a grain that addresses the maximum number of metrics. You can create a daily, monthly, or yearly summary aggregate tables based on the time dimension. However, it is recommended to create a daily summary aggregate tables that addresses most of the metrics for a report.

---

## Extend and Customize Sterling Business Intelligence

Sterling Business Intelligence can be customized to meet your business requirements. Prior to extending and customizing Sterling Business Intelligence, you must perform the following tasks:

1. Analyze the reports, model, and dashboards that are shipped along with Sterling Business Intelligence.
2. Based on your business requirements, identify the customizations that have to be performed.
3. Identify the components that have to be customized to meet your business requirements.

Based on your analysis, perform the following tasks:

- ◆ [Extend the Adaptive Warehouse Content](#)
- ◆ [Extend the Adaptive Analytics Reports](#)
- ◆ [Extend the Dashboards](#)
- ◆ [Customize the User Interface](#)
- ◆ [Customize the Home Page](#)
- ◆ [Customize the Staging Project](#)

**Note:** This version of Sterling Business Intelligence is shipped with the Order data mart. All the extensions should be limited to bringing in data that is related to orders or shipments because incremental change data capture is supported only for the order and shipment objects.

### Extend the Adaptive Warehouse Content

The Sterling Business Intelligence model can be customized based on your business requirements. A model is a set of related warehouse objects such as dimensions, facts, and calendars required for one or more related reporting applications. To customize your model, you must identify the warehouse objects required to meet your business requirements, create the warehouse model, and publish it.

**Prerequisites:** To extend the Adaptive Warehouse content, you must be familiar with developing metadata model using the IBM<sup>®</sup> Cognos<sup>®</sup> Adaptive Warehouse and the IBM<sup>®</sup> Cognos<sup>®</sup> Framework Manager applications.

To extend and customize the Adaptive Warehouse content, perform the following tasks:

- ◆ Modify the existing dimensions, facts, and add new warehouse objects
- ◆ Add new facts, dimensions, and create a new star schema
- ◆ Add custom hierarchy, target model filters, and new target model calculation
- ◆ Add custom hierarchy, hierarchy levels to existing hierarchy, source and target model filters, filters on warehouse objects, and new target model calculation

When extending and customizing the Adaptive Warehouse content, you must not perform the following tasks:

- ◆ Delete existing warehouse objects, including facts, dimensions, and references
- ◆ Modify the existing target model calculations, filters, and hierarchy

To extend the Adaptive Warehouse content, modify the existing Adaptive Warehouse projects that are shipped with Sterling Business Intelligence, namely Sterling Data Source (data mart project).

For more information about working with warehouse objects, refer to the *IBM® Cognos® 8 Analytic Applications Adaptive Warehouse User Guide*.

## Extend the Adaptive Analytics Reports

Based on your business requirements, you can add new reports in Sterling Business Intelligence. For customizing the reports, you must determine the metric types, hierarchy references, and the corresponding packages, create a deployment archive, and publish the reports.

**Prerequisites:** To extend the reports in Sterling Business Intelligence, you must be familiar with developing reports in the IBM® Cognos® Adaptive Analytics application.

**Note:** When extending and customizing reports using Sterling Business Intelligence, you can add new reports, but you should not delete or modify the existing reports.

To create a new analytic report, follow the process listed here:

1. Open the Sterling Data Source Adaptive Analytics project.
2. Create a Report Pack within a metric type. The metric type defines what you are reporting on.
3. Select an Hierarchy to report.
4. Create an Analytic Type. The Analytic Type defines the style of the report.
5. Optionally, set other properties as follows:
  - ◆ Define additional measures or levels (hierarchy)
  - ◆ Define filters

**Note:** All adaptive analytic reports must contain a time reference. If the time hierarchy is not present in the report, you must select an available prompt parameter in the Parameter Configuration screen and set the time reference properties to *Yes*.

For more information about generating analytic reports, refer to the *IBM® Cognos® 8 Analytic Applications Adaptive Analytics User Guide*.

## Extend the Dashboards

Based on your business requirements, you can add dashboards in Sterling Business Intelligence. For customizing the dashboards, you must clearly understand the business requirements and the layout of the dashboards. Ensure that the metrics and the dimensions required are available in the model. If the metrics and dimensions are not available, modify the model to meet your business requirements.

**Prerequisites:** To extend and customize the dashboards, you must be familiar with the task of authoring reports using IBM® Cognos® 8 BI Report Studio Professional.

## Notes:

- ◆ You must not modify or delete the existing dashboards. If you want to modify the existing dashboards, ensure that you make a copy of the dashboard and modify it in the same folder where the existing dashboards are present.
- ◆ All customization of the dashboards must be saved in the `Analytic Applications/Info Packs/Sterling Analytics Package/Metric Type Report Packs/Sterling TOC /Dashboards` folder in the Content Store.

For more information about creating reports, refer to the *IBM® Cognos® 8 BI Report Studio Professional Authoring User Guide*.

## Customize the User Interface

You can customize the Sterling Business Intelligence user interface by using the Designer Workbench application, which allows you to add new screens.

**Prerequisites:** To customize the Sterling Business Intelligence user interface, you must be familiar with the task of developing an user interface using the Sterling Web UI Framework.

In Sterling Business Intelligence, you can perform the following customizations:

- ◆ **Create new screens:** You must create a new screen if you are adding a new report or a dashboard and want to deliver this using the Web UI Framework. You can create new screens using the Sterling Designer Workbench application.  
**Note:** You cannot modify an existing screen. You can only add new screens. When customizing the user interface, copy the standard resources of the application and then modify your copy. Do not modify the standard resources of the application.
- ◆ **CSS:** Icons and font sizes in the user interface can be customized by modifying the CSS entries in the CSS file. You can create a new CSS entry, and extend and override the existing CSS entries. This is done by creating a file in the `extn` folder with the same structure as the `sbi` folder. For more information about customizing a CSS file, refer to the *Web UI Framework Extensibility Guide*.
- ◆ **Menu:** The menu bar and the menu options in the interface can be customized. You can customize the sequence of the menu options and add new menu options. For more information about customizing menu options, refer to the *Web UI Framework Extensibility Guide*.

For more information about extending the Web user interface, refer to the *Customizing the Web UI Framework Guide*.

## Create a New Screen

1. Install the Sterling Business Intelligence application and build a WAR file for it.
  - a. Deploy the WAR file on the server in the exploded format.
  - b. After the deployment is completed, start the application server.
2. Copy the `SBISample.json` file from the `<web_ui_extn>/sbi/bi/json` folder to the appropriate folder and rename it as `<screenname>.json`.
3. Open the `<screenname>.json` file in Designer Workbench.

4. Change the properties of the screen that you want to create as follows:
    - ◆ `sciId` as `<screen name >`
    - ◆ `class name` as `sc.bi.<screenname>`
    - ◆ `regxtype` as `sc.bi.<screenname>`
    - ◆ Source namespaces as the namespace names as required by the added widgets in the screen.
  5. Change the value of the hidden parameter `report_name` to `<cognos report name>`.
  6. Add the required widgets in the **SearchPanel**. For details about adding a widget to a screen, refer to the *Customizing the Web UI Framework Guide*.
- Note:** If you do not require the Time Period drop-down list in a screen, remove the `TimePanel` from the screen.
7. If the custom report has a new parameter, which needs to be displayed as a list of values, for example, as a check box or Combo box, from the datamart database, then you must change the value of a hidden parameter named `QueryIds`. The `QueryIds` should be same as the parameter in the report. Change the value of the hidden parameter `QueryIds` as required, using the added widgets in the screen.

**Note:**

- ◆ The `sbi_query.xml` file defines a list of queries used by the `entityLookup` API. Each query defined in `sbi_query.xml` is identified by a `QueryId`. For details about the `entityLookup` API, refer to API javadocs.
  - ◆ If you are adding a new query to populate a widget in a screen, you must define a new entry in a new file named `sbi_query.xml` inside the `<Runtime>/repository/xapi/template/merged/metadata/extn` folder.
  - ◆ For new `QueryIds`, you must extend and add new corresponding entries pertaining to the mashups. The `id` attribute of the mashup entry should be `get<queryid>`. The value of `QueryIds` must be a coma separated value and each value must be same as defined in the `sbi_query.xml` file.
  - ◆ To extend the `sbi_mashups.xml` file, create a new file, `<filename>.xml`, in the `<Runtime>/repository/eardata/sbi/war/mashupxmls/extn` folder.
8. Navigate to **Screen > SearchPanel > Buttons > Btn\_Search**.
  9. Change the value of `Handler` property to `sc.bi.utils.generateReportForPrompt`.
  10. Click **Save**.

The `<screen name>_config.js` and `<screen name>.js` files are created.

11. Add the following code in the `<screen name>.js` file within the `<screen name>` class:

```
getViewID :function(){
    return "<screenname>";
},
getScreenContext : function(){
    return "<screenname>";
}
```

12. Copy the <screen name>\_config.js and <screen name>.js files, created in step 9 to <INSTALL\_DIR>/extensions/sbi/webpages/bi/screens folder. If this folder does not exist, create the same.
13. Create a Java Script builder file sbi.jsb in the <INSTALL\_DIR>/extensions/<sbi>/webpages/bi/screens folder. The folder contains entries for the newly created screens.

**Note:** If you have chosen to retain the **Time Period** drop-down list, add the entry in the same target element of sbi.jsb file.

Following is the sample code for JSB:

```
<?xml version="1.0" encoding="utf-8"?>
<project name="sbi" author="Sterling Commerce Pvt.Ltd" version="1.0"
copyright="$projectName&#xD;&#xA;Copyright(c) 2006,
$author.&#xD;&#xA;&#xD;&#xA;This code is licensed under BSD license. Use
it as you wish, &#xD;&#xA;but keep this copyright intact."
source="False" source-dir="$output/source" minify="True"
min-dir="$output/build" doc="False" doc-dir="$output/docs" master="true"
master-file="$output/yui-ext.js" zip="false"
zip-file="$output/yuo-ext.$version.zip">
    <target name="<screenname>"
file="/extn/bi/scripts/<screenname>Lib.js" allowDynamicLoad="true"
debug="true" shorthand="false" shorthand-list="">
        <include name="/extn/bi/screens/<screenname>_config.js" />
        <include name="/extn/bi/screens/<screenname>.js" />
        <include name="/bi/common/TimeScreen_config.js" />
        <include name="/bi/common/TimeScreen.js" />
    </target>
</project>
```

14. Create a new servlet to register the new JSB and mashup file.

Following is a sample code for creating servlet:

```
import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import
com.sterlingcommerce.ui.web.framework.helpers.SCUIJSLibraryHelper;
import
com.sterlingcommerce.ui.web.framework.helpers.SCUI MashupHelper;
public class CreateServlet extends HttpServlet {
private static final long serialVersionUID =
4693417985837892469L;
public synchronized void init(final ServletConfig config)
throws ServletException {
//loads the JSB specified at the path
SCUIJSLibraryHelper.loadJSLibraryXml
("extn/bi/screens/sbi.jsb", config.getServletContext());
//loads the mashup XML specified at the path
SCUI MashupHelper.loadMashupXml
("/extn/bi/screens/<filename.xml>", config.getServletContext());
}
}
```

15. Package the servlet into a JAR file.

16. Create a new web.xml file by copying the original

<INSTALL\_DIR>/repository/eardata/sbi/extn/web.xml.sample file in the same directory, and add the entry for the servlet in web.xml file as follows:

```
<servlet>
    <display-name><servletName></display-name>
    <servlet-name><servletName></servlet-name>
    <servlet-class><full servlet class name> </servlet-class>
    <load-on-startup>1500</load-on-startup>
</servlet>
```

17. Run the following command in the <INSTALL\_DIR>/bin directory so that the JAR file containing the servlet is added to the WAR file when the WAR file is generated:

```
install3rdParty.sh <vendorName> <vendorVersion> -j <full path of the JAR
file> -targetJVM APP (for UNIX)
install3rdParty.cmd <vendorName> <vendorVersion> -j <full path of the JAR
file> -targetJVM APP (for Microsoft Windows)
```

18. Create a minified file using the following command:

```
sci_ant.sh -f jsUtil.xml minify-js -Dgis.install=<Install> -Dminify=true  
(for UNIX)
```

```
sci_ant.cmd -f jsUtil.xml minify-js -Dgis.install=<Install> -Dminify=true  
(for Microsoft Windows)
```

For more details about compiling and minimizing JavaScript files, refer to *Customizing Web UI Framework Guide*.

19. Create the resource jar file using the following command:

```
<INSTALL_DIR>/bin/deployer.sh -t resourcejar (for UNIX)
```

```
<INSTALL_DIR>/bin/deployer.cmd -t resourcejar (for Microsoft Windows)
```

20. Create a new menu item using Application Manager, and attach the Resource ID. For more information about configuring presentation components, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
21. Create user permissions for the resources that have been added. For more information about user permissions, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
22. Re-create the `smcfs.ear` file.

For more information about recreating the EAR file, refer to the *Selling and Fulfillment Foundation: Installation Guide*. Create a resource with `resourceId` as `<screenname>` for the new screen. For more information about creating a resource, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

## Customize the Home Page

You can customize the Sterling Business Intelligence home page by defining the custom dashlets in the `extended_dashlets.xml`.

You can perform the following actions to customize the home page:



1. Extended dashlets.xml and dashboard.xml must be copied to the following location in the Selling and Fulfillment Foundation runtime folder:

/extensions/sbi/webpages/bi/home/metadata/

Modify the existing dashlets.xml and dashboard.xml that resides in the <INSTALL\_DIR>/repository/eardata/sbi/war/bi/home/metadata/ folder.

It is recommended that you copy the existing dashlets.xml and dashboard.xml from the <INSTALL\_DIR>/repository/eardata/sbi/war/bi/home/metadata/ folder and make changes to the same.

The following example is a snippet from dashlet.xml where all the new reports are included:

```
<Dashlets
<Dashlet id="new_dashlet" title=" new_dashlet " pure="false" height="130">
  <Url>/bi/home/dashletLink.jsp?did= new_dashlet </Url>
  <Config numberOfColumns="2" >
  </Config>
  <Group>
<Section Id="dashboardlinks" Title="" IconCls="sbi-clean-1"
Icon="bi/img/clean.gif">
  <Link LinkId="EXTENDED_PerfectOrderDashboard"
Name="b_PerfectOrderDashboard"
URL="report.do?ResourceId=EXTNDED_SBIPftOrderDashboard&DocumentType=
0001" ResourceID="SBIPftOrderDashboard" ActionNs="/sbi/bi"/>
</Section>
</Dashlet>
</Dashlets>
```

2. sbi\_bundle.js and servlet to load the sbi.jsb must be extended to display the Dashlet title.

To extend the sbi\_bundle.js, perform the following steps:

- a. Create a new sbi\_bundle.js file and copy it in the Selling and Fulfillment Foundation runtime folder, extensions\sbi\webpages\bi\screens.

- b. The entry in the sbi\_bundle.js can be as follows;

```
Ext.override(sc.plat.ui.Screen, {
  "new_dashlet": "CUSTOM REPORTS",
});
```

- c. Create the following new target in the extended sbi.jsb;

```
<target name="extensionSBI" file="/bi/scripts/extensionSBILib.js"
loadAfter="sbiHome" allowDynamicLoad="true" debug="true"
depends="bicore" shorthand="false" shorthand-list="">
<include name="/extn/bi/screens/sbi_bundle.js"/>
</target>
```

This ensures that the new entries are retrieved from the newly created sbi\_bundle.js along with the original entries.

3. Rebuild the application EAR and deploy it on the application server.

## Customize the Staging Project

You can customize the staging project by creating a new Staging Entity Metadata file. In the Staging Entity Metadata file, you can add new filters to an existing entity if you want to exclude certain records when analyzing the data. You can also add a new entity and configure load partitions by modifying the NumberOfLoadIntervals attribute.

You can perform the following tasks when customizing the Staging Entity Metadata file:

- ◆ [Add a New Entity](#)
- ◆ [Add Filters to an Existing Entity](#)
- ◆ [Configure Load Partitions](#)
- ◆ [Deploy the Customization](#)

**Note:** You cannot perform the following tasks when customizing the Staging Entity Metadata file:

- ◆ Change the value of the DriverEntity attribute
- ◆ Change the existing filter of an entity
- ◆ Add a new relationship to an existing entity

## Add a New Entity

To add a new entity:

1. Create the `sbi_staging_entities.xml` file, if it does not already exist, in the `<INSTALL_DIR>/extensions/global/template/metadata` directory.
2. Create a new element, Entity. Consider the following structure of a sample Entity element when creating a new entity:

```
<DBSchema>
  <Entities>
    <Entity TableName="YFS_ORDER_HOLD_TYPE_LOG"
DriverEntity="YFS_ORDER_TS_TAG" NumberOfLoadIntervals="4" >
      <Attributes>
        <Attribute ColumnName="STATUS" AttributeQuery="NFLIKE">
          <AttributeValues>
            <AttributeValue Value="1300"/>
          </AttributeValues>
        </Attribute>
      </Attributes>
      <Relationships>
        <Relationship ForeignEntity="YFS_ORDER_HOLD_TYPE">
          <Attribute Name="ORDER_HOLD_TYPE_KEY"
ParentName="ORDER_HOLD_TYPE_KEY"/>
        </Relationship>
        <Relationship ForeignEntity="YFS_ORDER_TS_TAG">
          <Attribute Name="ORDER_HEADER_KEY"
ParentName="ORDER_HEADER_KEY"/>
        </Relationship>
      </Relationships>
    </Entity>
  </Entities>
</DBSchema>
```

3. Set the value of the XML attributes.

Refer to the following table for more information about defining the values of the otherXML attributes:

Component	Component Type	Component Location	Description
TableName	Attribute	Entities/Entity	Indicates the table name of the entity that is required to be loaded into staging database.

Component	Component Type	Component Location	Description
DriverEntity	Attribute	Entities/Entity	<p>Indicates the tablename of the driver entity that is related with this entity. The system supports the following driver tables:</p> <ul style="list-style-type: none"> <li>◆ YFS_ORDER_TS_TAG</li> <li>◆ YFS_SHIPMENT_TS_TAG</li> <li>◆ YFS_OPPUTUNITY_TS_TAG</li> </ul> <p>Based on the entity being defined, you must provide the correct driver entity value that is related with this entity. For example, for the YFS_ORDER_HOLD_TYPE_LOG entity that is related to ORDER, set the value of the DriverEntity attribute to YFS_ORDER_TS_TAG. The DriverEntity attribute is defined only for transaction tables. However, certain transaction tables such as YFS_PERSON_INFO do not contain any driver entity. Therefore, the value of the DriverEntity attribute should be left blank.</p>
NoOfLoadIntervals	Attribute	Entities/Entity	<p>This attribute is used to generate the initial load package file. For example, if you want to load data for one year from the YFS_ORDER_HEADER table and the value of the NumberOfLoadIntervals attribute is set to 4, four build streams are created for the YFS_ORDER_HEADER table when loading data in to Staging database and each build stream loads data of 3 months.</p>
Attributes	Element	Entities/Entity	<p>This element contains the filter information such as the column name on which a filter is applied, the filter operation, and the applicable filter attribute values.</p>
ColumnName	Attribute	Entities/Entity/Attributes/Attribute	<p>Indicates the column name of the table on which a filter is applied.</p>

Component	Component Type	Component Location	Description
AttributeQuery	Attribute	Entities/Entity/Attributes/Attribute	<p>Indicates the filter operation. The system supports the following filter operations:</p> <ul style="list-style-type: none"> <li>◆ EQ</li> <li>◆ NE</li> <li>◆ GT</li> <li>◆ GE</li> <li>◆ LT</li> <li>◆ LE</li> <li>◆ IS</li> <li>◆ ISNOT</li> <li>◆ BETWEEN</li> <li>◆ FLIKE</li> <li>◆ ELIKE</li> <li>◆ NFLIKE</li> <li>◆ NELIKE</li> <li>◆ IN</li> <li>◆ NOTIN</li> </ul>
Value	Attribute	Entities/Entity/Attributes/Attribute	Indicates the value of the filter attribute.
Relationship	Element	Entities/Entity/Relationships	<p>This element defines the relationship of an entity with its corresponding driver entity. It must be defined in the order in which the entities are related. For example, the YFS_ORDER_HOLD_TYPE_LOG entity is related with the YFS_ORDER_HOLD_TYPE driver entity and the YFS_ORDER_HOLD_TYPE entity is related with the YFS_ORDER_TS_TAG driver entity. Therefore, the relationship between the YFS_ORDER_HOLD_TYPE_LOG entity and YFS_ORDER_HOLD_TYPE driver entity must be defined first and then the relationship between the YFS_ORDER_HOLD_TYPE entity and the YFS_ORDER_TS_TAG driver entity must be defined.</p>
ForeignEntity	Attribute	Entities/Entity/Attributes/Relationships/Relationship	Indicates the foreign entity.
Name	Attribute	Entities/Entity/Relationships/Relationship	Indicates the column name of the entity.

Component	Component Type	Component Location	Description
ParentName	Attribute	Entities/Entity/Relation Ships/RelationShip	Indicates the column name of the ForeignEntity that has a foreign key element relationship.

## Add Filters to an Existing Entity

You can add new filters to an existing entity if you want to exclude certain records when analyzing the data. For example, the model that is shipped along with Sterling Business Intelligence consists of all the hold types, including the system-specific hold types. You can exclude specific hold types and perform the analysis.

To add filters to an existing entity:

1. Create the `sbi_staging_entities.xml` file, if it does not already exist, in the `<INSTALL_DIR>/extensions/global/template/metadata` directory.
2. Create a structure of the existing entity. For example, create a structure as follows:

```
<DBSchema>
  <Entities>
    <Entity TableName="YFS_HOLD_TYPE" OverrideDefaultFilters="Y">
      </Entity>
    </Entities>
  </DBSchema>
```

**Note:** For Entities which are included in the `sbi_staging_entities.xml` file provided with Sterling Business Intelligence, if you want to override the filters applied, set the attribute `OverrideDefaultFilters="Y"`. Setting this property would mean that the default filters would not be applied. If you want to add additional filters to the default filters, set the attribute `OverrideDefaultFilters="N"`. Please be aware that additional filters are “AND”, not “OR”.

For example, by default the `YFS_ORDER_DATE` entity is filtered based on `DATE_TYPE_ID = 'YCD_FTC_FIRST_PROMISE_DATE'`. To override the default behaviour and filter the `YFS_ORDER_DATE` based on `CustomDateType`, create a structure as follows:

```
<Entity TableName="YFS_ORDER_DATE" OverrideDefaultFilters="Y">
  <Attributes>
    <Attribute ColumnName="DATE_TYPE_ID" AttributeQuery="EQ">
      <AttributeValues>
        <AttributeValue Value="<CustomDateType>" />
      </AttributeValues>
    </Attribute>
  </Attributes>
</Entity>
```

3. Create an element, `Attribute`, as a child element of the root element, `Attributes`.
4. Set the value of the `ColumnName` attribute.
5. Set the value of the `AttributeQuery` attribute.
6. Create an element, `AttributeValues`, as child element of the root element, `Attribute`.

7. Set the value of the Value attribute.
8. To add multiple filters, repeat steps 3 through 7.

Following is the sample of the Staging Entity Metadata file that contains new filters for an existing entity:

```
<DBSchema>
  <Entities>
    <Entity TableName="YFS_HOLD_TYPE" >
      <Attributes>
        <Attribute ColumnName="HOLD_TYPE" AttributeQuery="IN">
          <AttributeValues>
            <AttributeValue Value="Fraud Check"/>
            <AttributeValue Value="Credit Fraud"/>
          </AttributeValues>
        </Attribute>
      </Attributes>
    </Entity>
  </Entities>
</DBSchema>
```

## Configure Load Partitions

You can configure the load partitions by modifying the `NumberOfLoadIntervals` attribute in the Staging Entity Metadata file. The `NumberOfLoadIntervals` attribute is used to load initial data. For example, if you want to load data for one year from the `YFS_ORDER_HEADER` table and the value of the `NumberOfLoadIntervals` attribute is set to 4, four build streams are created for the `YFS_ORDER_HEADER` table when loading data into the Staging database and each build stream loads data of 3 months. The default value of this attribute is 2. By increasing the value of this attribute, the number of processes and database connections are also increased. Set this value based on the number of connections and processes that the environment can support.

To modify the `NumberOfLoadIntervals` attribute in an existing entity:

1. Create the `sbi_staging_entities.xml` file, if it does not already exist, in the `<INSTALL_DIR>/extensions/global/template/metadata` directory.
2. Create an element, `Entity`.
3. Set the value of the `TableName` attribute.

4. Set the value of the NumberOfLoadIntervals attribute.

Following is the sample of the Staging Entity Metadata file that contains a modified NumberOfLoadIntervals attribute in an existing entity:

```
<DBSchema>
  <Entities>
    <Entity TableName="YFS_ORDER_RELEASE_STATUS" NumberOfLoadIntervals="4" >
  </Entities>
</DBSchema>
```

## Deploy the Customization

After performing the required customizations in a Staging Entity Metadata file, perform the following steps:

1. To create the resource jar file, run the following command:

For Windows:

```
<INSTALL_DIR>/bin/deployer.cmd -t resourcejar
```

For Linux or UNIX:

```
<INSTALL_DIR>/bin/deployer.sh -t resourcejar
```

2. Re-create the `smcfs.ear` file. For more information about recreating the EAR file, refer to the *Selling and Fulfillment Foundation: Installation Guide*.
3. Generate the staging project package file. For more information about generating the staging project package file, see *Generate the Staging Project Package File* on page 15.



---

# The Sterling Business Intelligence Model

This topic explains the data model and the schemas pertaining to Sterling Business Intelligence.

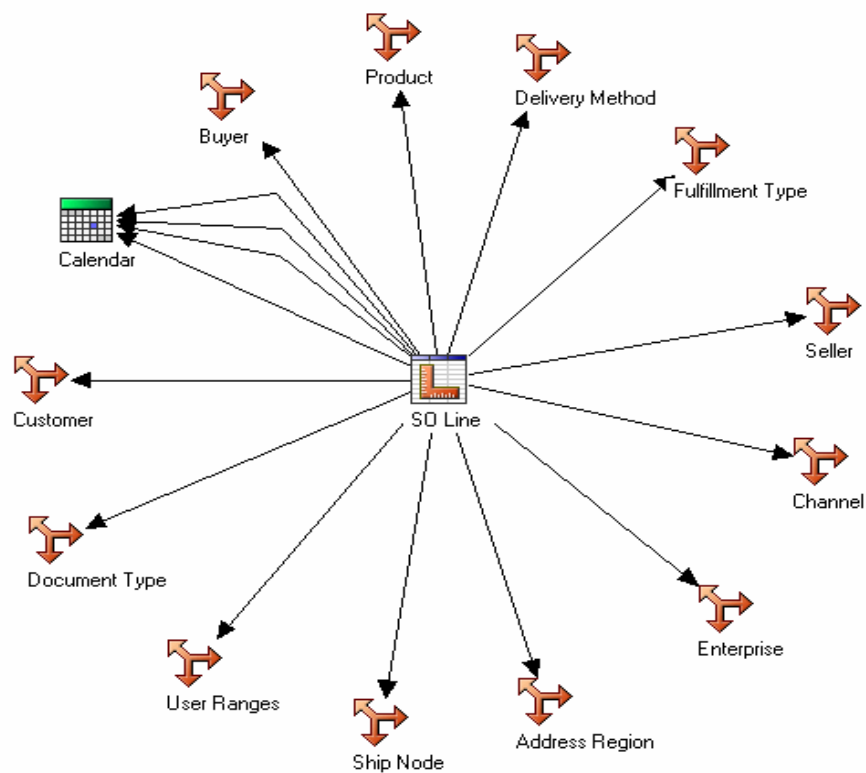
## Warehouse Model Star Schemas

Sterling Business Intelligence provides metadata to deliver the warehouse content in the warehouse model using the star schemas shown in the following illustration.

The tables in this section list the dimensions and the corresponding role names for each star schema. A role is the role name given when multiple references exist between a fact and a dimension.

### SO Line Star Schema

The SO Line star schema is illustrated here.

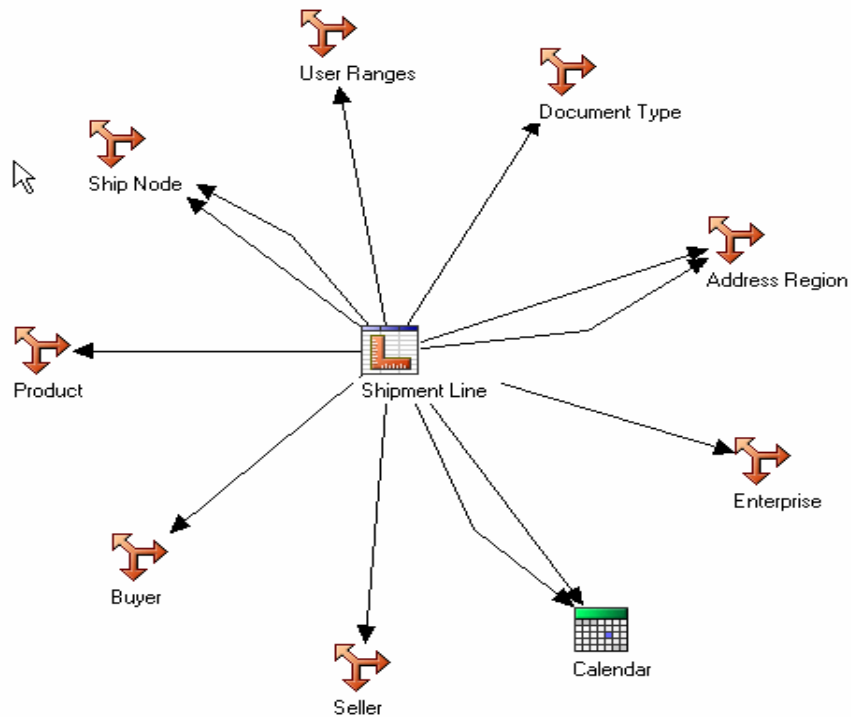


The SO Line star schema consists of SO Line measures and references to the dimensions listed in the following table:

<b>Dimension</b>	<b>Role</b>
Enterprise	Default
Address Region	Default
Calendar	Actual Ship Date-SO Line Released Date Confirmed Date Promised Ship Date
Ship Node	Default
User Ranges	Default
Buyer	Default
Seller	Default
Customer	(Default)
Product	Default
Delivery Method	Default
Fulfillment Type	Default
Document Type	Default
Channel	Default

## Shipment Line Star Schemas

The Shipment Line star schema is illustrated here.



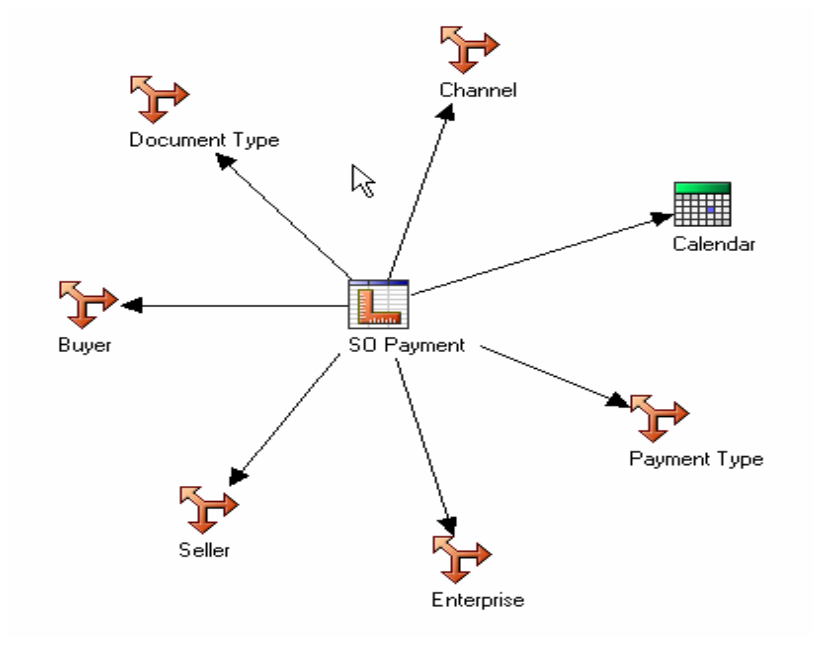
The Shipment Line star schema consists of the Shipment Line measures and references to the dimensions listed in the following table:

Dimension	Role
Enterprise	Default
Address Region	Ship To Bill To
Calendar	Actual Ship Date-Shipment Line Expected Shipment Date
Ship Node	Default Procurement Node
User Ranges	Default
Buyer	Default

Dimension	Role
Seller	Default
Product	Default
Document Type	Default

## SO Payment Star Schema

The SO Payment star schema is illustrated here.

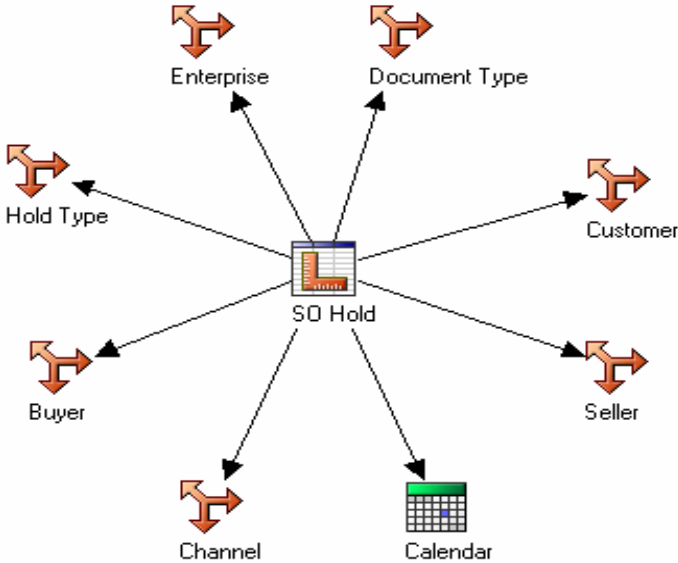


The SO Payment star schema consists of the SO Payment measures and references to the dimensions listed in the following table:.

Dimension	Role
Calendar	Charge Date
Channel	Default
Document Type	Default
Seller	Default
Buyer	Default
Enterprise	Default
Payment Type	Default

# SO Hold Star Schema

The SO Hold star schema is illustrated here.

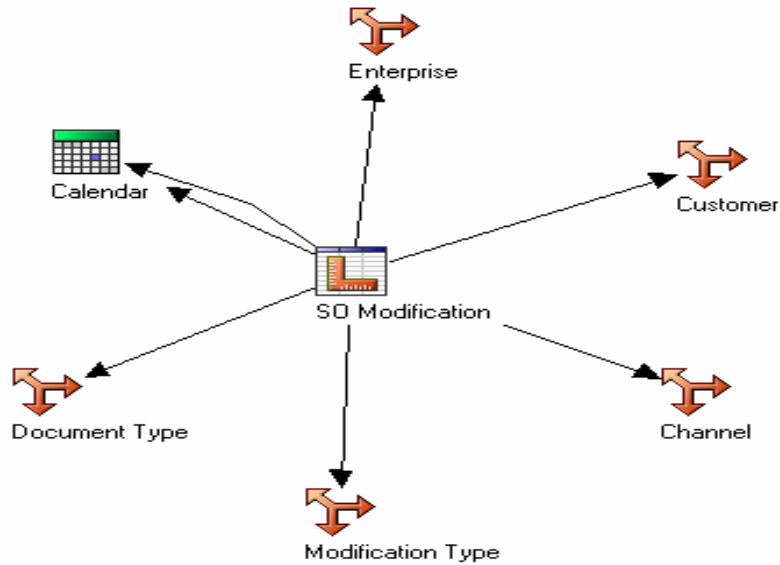


The SO Hold star schema consists of the SO Hold measure and references to the dimensions listed in the following table:.

Dimension	Role
Enterprise	Default
Customer	Default
Document Type	Default
Calendar	Hold Date
Seller	Default
Channel	Default
Buyer	Default
Hold Type	Default

## SO Modification Star Schema

The SO Modification star schema is illustrated here.

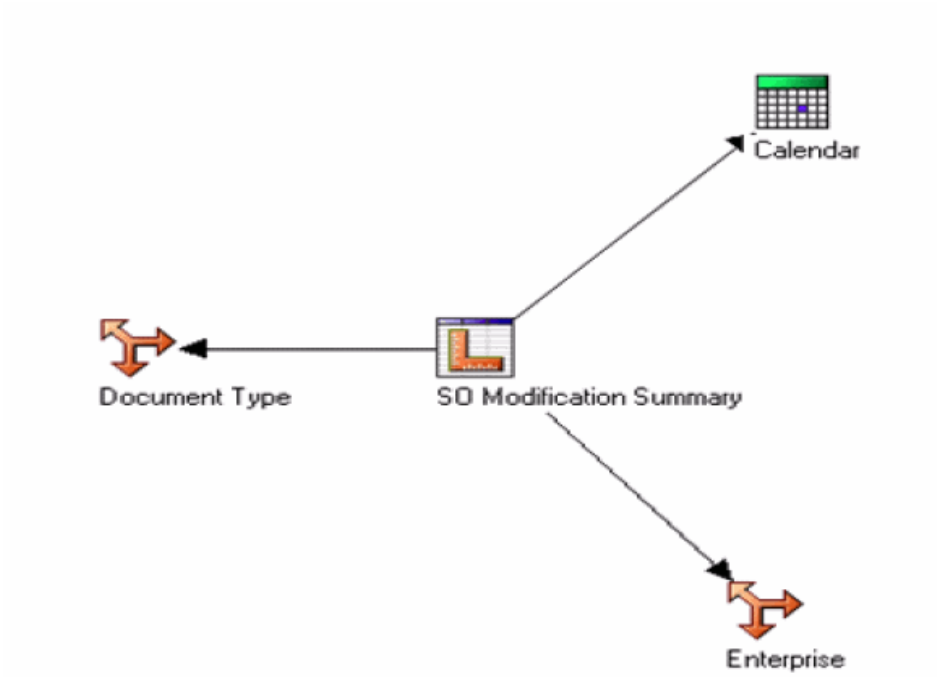


The SO Modification star schema consists of the SO Modification measure and references to the dimensions listed in the following table:.

Dimension	Role
Enterprise	Default
Customer	Default
Channel	Default
Modification Type	Default
Document Type	Default
Calendar	Confirmed Date Modify Date

# SO Modification Summary Star Schema

The SO Modification Summary star schema is illustrated here:

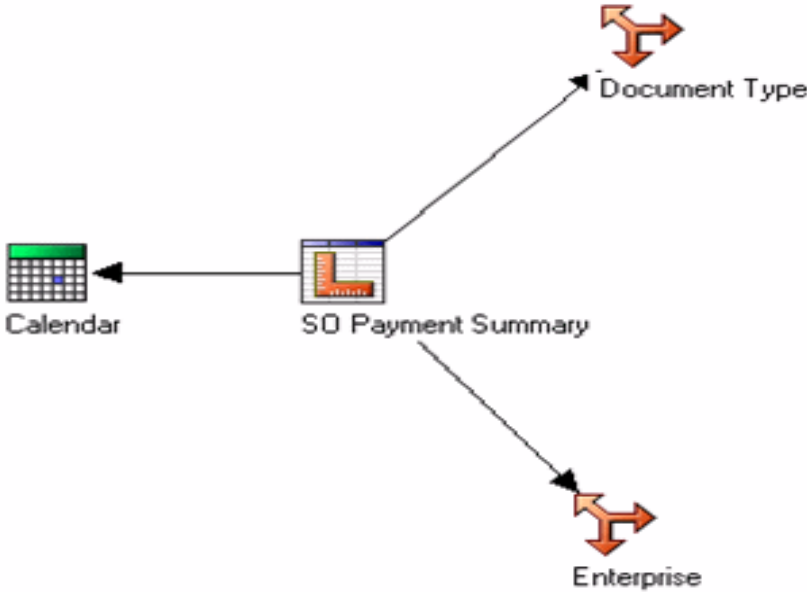


The SO Modification Summary star schema consists of the SO Modification Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Calendar	Default
Enterprise	Default
Document Type	Default

# SO Payment Summary Star Schema

The SO Payment Summary star schema is illustrated here:



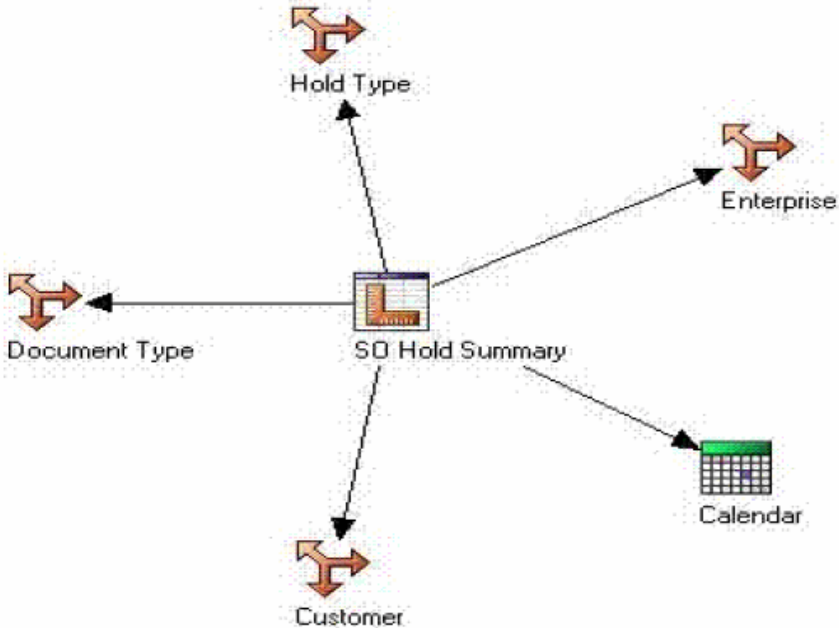
The SO Payment Summary star schema consists of the SO Payment Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Calendar	Default
Enterprise	Default
Document Type	Default



# SO Hold Summary Star Schema

The SO Hold Summary star schema is illustrated here:

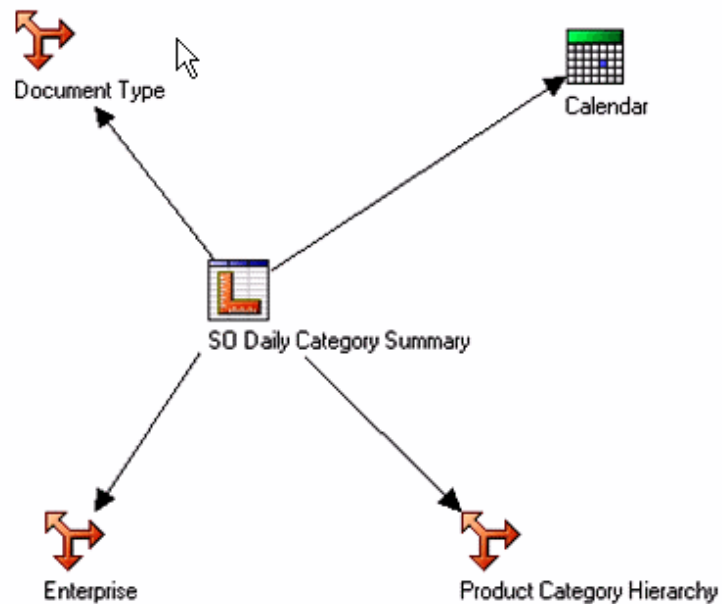


The SO Hold Summary star schema consists of the SO Hold Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Hold Type	Default
Document Type	Default
Calendar	Default
Enterprise	Default

## SO Daily Category Summary Star Schema

The SO Daily Category Summary star schema is illustrated here:

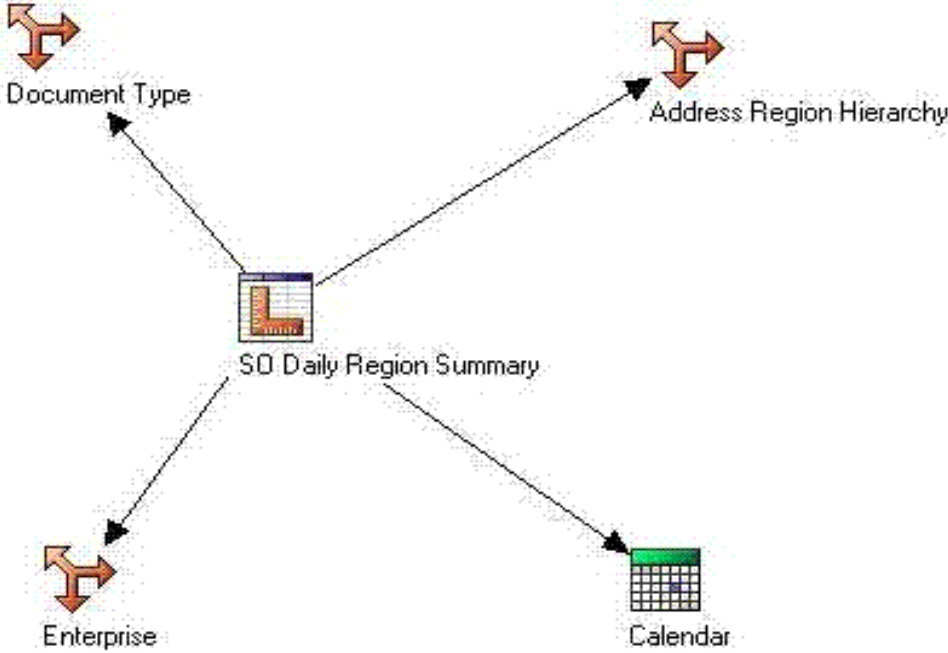


The SO Daily Category Summary star schema consists of the SO Daily Category Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Calendar	Default
Product Category Hierarchy	Default
Enterprise	Default
Document Type	Default

# SO Daily Region Summary Star Schema

The SO Daily Region Summary star schema is illustrated here:

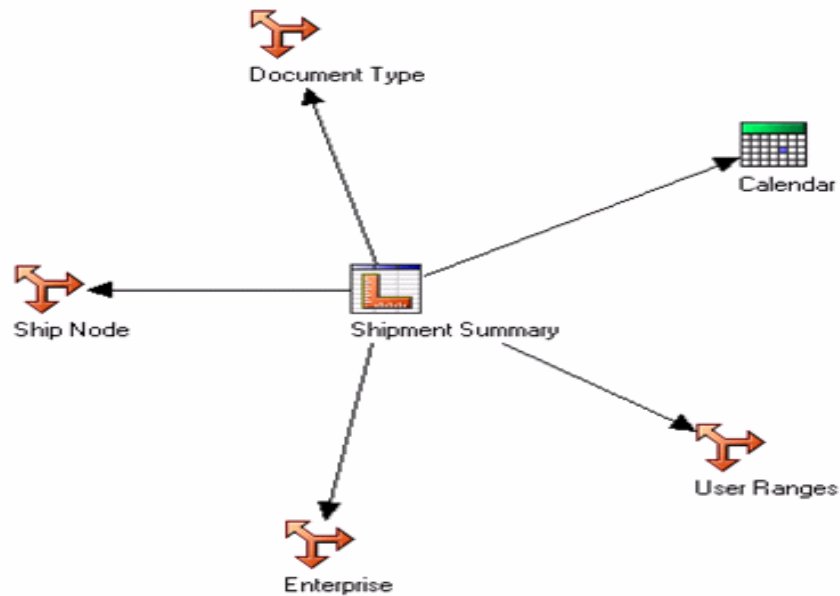


The SO Daily Region Summary star schema consists of the SO Daily Region Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Address Region Hierarchy	Default
Calendar	TranDate
Enterprise	Default
Document Type	Default

## Shipment Summary Star Schema

The Shipment Summary star schema is illustrated here:

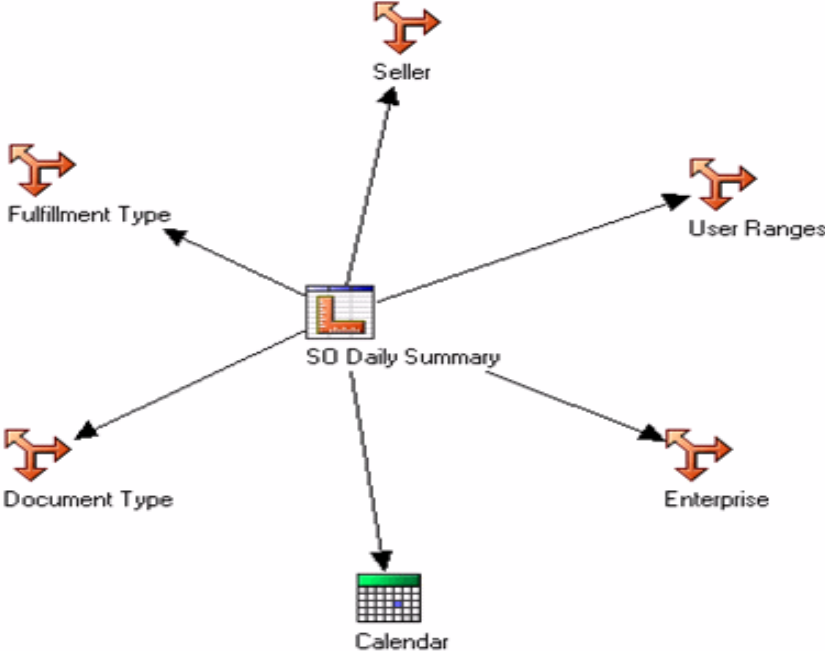


The Shipment Summary star schema consists of the Shipment Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Calendar	Default
User Ranges	Default
Enterprise	Default
Ship Node	Default
Document Type	Default

# SO Daily Summary Star Schema

The SO Daily Summary star schema is illustrated here:

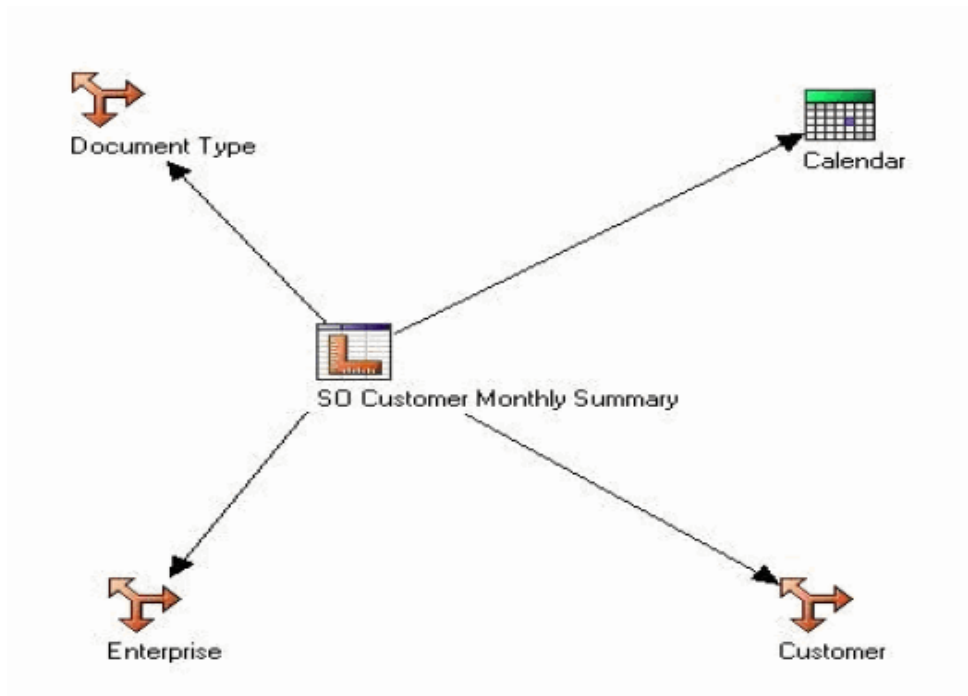


The SO Daily Summary star schema consists of the SO Daily Summary measure and references to the dimensions listed in the following table:

Dimension	Role
Seller	Default
User Ranges	Default
Enterprise	Default
Calendar	TranDate
Document Type	Default
Fulfillment Type	Default

## SO Customer Monthly Summary Star Schema

The SO Customer Monthly Summary star schema is illustrated here:

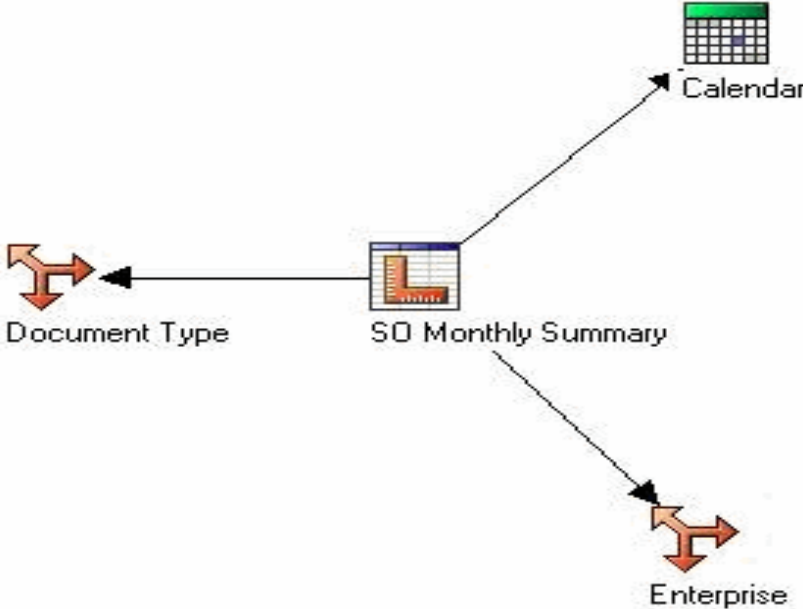


The SO Customer Monthly Summary star schema consists of the SO Customer Monthly Summary measure and references to the dimensions listed in the following table:.

Dimension	Role
Calendar	Default
Enterprise	Default
Customer	Default
Document Type	Default

# SO Monthly Summary Star Schema

The SO Monthly Summary star schema is illustrated here:



The SO Monthly Summary star schema consists of the SO Monthly Summary measure and references to the dimensions listed in the following table:.

<b>Dimension</b>	<b>Role</b>
Calendar	Default
Enterprise	Default
Document Type	Default

## Target Model Star Schemas

The target model generated by the Adaptive Warehouse applies the best practices for reporting. The target model provides a business view of the information in the source model, and is used to simplify the building of reports, analyses, and queries. The following star schema groupings are delivered in the target model of the Sterling Business Intelligence Warehouse content.

### SO Line Star Schema Grouping

The SO Line star schema grouping consists of the SO Line measures and the following dimensions.

#### Dimensions

- ◆ Enterprise
- ◆ Buyer
- ◆ Seller
- ◆ Channel
- ◆ Calendar
- ◆ Delivery Method
- ◆ Fulfillment Type
- ◆ Product
- ◆ Product Category Hierarchy
- ◆ Address Region
- ◆ Ship Node
- ◆ Address Region Hierarchy
- ◆ Customer
- ◆ Customer Hierarchy
- ◆ Document Type
- ◆ User Ranges



## Measures

The following table provides a list of the SO Line measures:

<b>Measure</b>	<b>Description</b>
Total Line Charges	This is the total amount pertaining to an order line.
Is Shipped	This measure indicates whether the order line is shipped.
Elapsed Time Actual Ship Promised	The time that has elapsed between the actual shipment time and the promised time (in minutes).
Elapsed Time Actual Ship Released	The time that has elapsed between the actual shipment time and the released time (in minutes).
Elapsed Time Confirm Release	The time that has elapsed between the released time and confirmed time (in minutes)
Shipped Quantity	This is the shipped quantity of the order lines.
Is ASAP Order	This measure indicates whether the order is an ASAP order.
Ordered Quantity	This is the ordered quantity of the order lines.
Capture to Shipment Time for ASAP Order	This is the time between the shipped time and the captured time for ASAP orders.
Is Delayed	This measure indicates whether the order line has been delayed.
Is BackOrdered	This measure indicates whether the order line has been backordered.
Is BackOrdered and Delayed	This measure indicates whether the order line has been backordered and delayed.
Is Shipped On Time	This measure indicates whether the order line has been shipped on time.
Elapsed Time Actual Ship Confirmed	The time that has elapsed between the actual shipment time and the confirmed time (in minutes).
Elapsed Lead Time ASAP	The time that has elapsed for the ASAP orders (in minutes).
Sales Order Line Count	This is the total number of order lines.
Fulfilled Revenue (Currency)	This is the fulfilled revenue of an order line.
Is Released	This measure indicates whether the order line has been released.
Booked Value (Currency)	This measure indicates the booked value.
Is Shipped ASAP Order	This measure indicates whether the shipped order is an ASAP order.
Not Cancelled	This measure indicates whether the order line has been cancelled.

## Grouped Measures

The following table provides a list of the SO Line group measures:

Measure	Description
Percentage Shipped on Time	This is the percentage of order lines shipped on or before the first promised date.
No. of Orders Shipped	This is the total number of orders shipped.
Avg. Exp. Lead Time (ASAP Orders)	This is the time between the promised date and the confirmed date for ASAP orders.
Avg. Capture to Ship Time (ASAP)	This is the average amount of time between the stage an ASAP order line is captured and the time it is shipped.
Avg. Lead Time to Ship	This is the average time required to move the order lines from the release stage to the ship stage.
Avg. Value of an Order	This is the average booked value of confirmed orders.
No. of Order Lines Confirmed	This is the total number of order lines confirmed.
Percentage BackOrdered	This is the percentage of shipment lines backordered from a node.
No. of Order Lines Shipped	This is the total number of order lines shipped.
Avg. Capture to Ship Time (ASAP)	This is the average amount of time between the stage when an ASAP order line is captured and the time it is shipped.
Avg. Order to Release Time	This is the average time required to move the order lines from the order capture stage to the order release stage.
Avg. Order Lines per Order	This is the average number of order lines in a confirmed order.

## Shipment Line Star Schema Grouping

The Shipment Line star schema grouping consists of the Shipment Line measures and following dimensions.

### Dimensions

- ◆ Seller
- ◆ Ship Node
- ◆ Enterprise
- ◆ Buyer
- ◆ Address Region
- ◆ Product
- ◆ Product Category Hierarchy
- ◆ Address Region Hierarchy
- ◆ Calendar
- ◆ Document Type

## Measures

The following table provides a list of the Shipment Line measures:

<b>Measure</b>	<b>Description</b>
Shipped Quantity	This measure indicates the quantity that has been shipped.
Elapsed Time Expected Ship	This measure indicates the difference between the shipped time and the expected time (in minutes).
Is Procured	This measure indicates whether the shipment line has been procured.
Is Shipped	This measure indicates whether the shipment has been shipped.
Is Delayed	This measure indicates whether a shipment line has been delayed.
Delay in Shipment per Line	This measure indicates the delay in shipment for each shipment Line.
Elapsed Time Released Ship	This measure indicates the difference between the shipment time and the released time (in minutes).
Is Shipped on Time	This measure indicates whether the shipment has been shipped on time.
Is Backordered from Node	This measure indicates whether the order line has been backordered from a node.
Ordered Quantity	This measure provides the ordered quantity.
Is Fulfillment Partner	This measure indicates whether the ship node of a shipment line is a fulfilled partner.
Delay in Shipment	This measure indicates the delay in shipment for each shipment.

## Grouped Measures

The following table provides a list of the Shipment Line group measures:

<b>Measure</b>	<b>Description</b>
Percentage Shipped on Time	This is the percentage of order lines that are shipped on or before the first promised date.
No. of Shipment Lines Shipped	This is the total number of shipment lines shipped.
Avg. Release to Ship Time	This is the average time required to move the shipment lines from the release stage to the ship stage.
Percentage Procured	This is the percentage of shipment lines procured for a node.
No. of Shipments Shipped	This is the total number of shipment shipped.
Avg. Value of an Order	This is the average booked value of the confirmed orders.
Avg. Delay in Shipment	This is the average delay in the delayed shipments shipped.

## SO Payment Star Schema Grouping

The SO Payment star schema grouping consists of the SO Payment measures and the following dimensions.

### Dimensions

- ◆ Enterprise
- ◆ Calendar (Charge Date)
- ◆ Buyer
- ◆ Seller
- ◆ Channel
- ◆ Payment Type
- ◆ Document Type

### Measures

The following table provides a list of the SO Payment measures:

Measure	Description
Authorization Count	This measure indicates whether the transaction has been authorized.
Authorization Failure Count	This measure indicates whether the authorization of transaction has failed.
Credit Amount	This is the amount that is actually charged or refunded to the buyer.
Authorized Amount	This is the amount that is authorized for an order.
Settlement Count	This measure indicates whether the transaction settlement has been completed or not.
Settlement Failure Count	This measure indicates whether the settlement of transaction has failed.
Request Amount	This is the amount that has been requested for settlement or authorization.
Settled Amount	This measure is the amount for which a settlement has been requested.

### Grouped Measures

The following table provides a list of the SO Payment grouped measures:

Measure	Description
Percentage Authorization Failures	This is the percentage of payment instruments that failed during the authorization of payment.
Percentage Settlement Failures	This is the percentage of settlements that failed during the settlement cycle.

## SO Hold Star Schema Grouping

The SO Hold star schema grouping consists of the SO Hold measure and following dimensions.

### Dimensions

- ◆ [Calendar \(Hold Date\)](#)
- ◆ [Enterprise](#)
- ◆ [Seller](#)
- ◆ [Buyer](#)
- ◆ [Channel](#)
- ◆ [Hold Type](#)
- ◆ [Document Type](#)
- ◆ [Customer](#)
- ◆ [Customer Hierarchy](#)

### Measures

The SO Hold measure is Hold Duration. This field indicates the duration of hold applied on an order (in minutes).

### Grouped Measures

The following table provides a list of the SO Hold grouped measures:

Measure	Description
No. of Orders on Hold	This is the total number of orders on hold.
Avg. Hold Duration	This is the average time the orders are kept on hold.

## SO Modification Star Schema Grouping

The SO Modification star schema grouping consists of the SO Modification measure and following dimensions.

### Dimensions

- ◆ [Calendar \(Modify Date\)](#)
- ◆ [Channel](#)
- ◆ [Modification Type](#)
- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Customer](#)
- ◆ [Customer Hierarchy](#)

### Measures

The SO Modification measure is Is Manually Modified. This field indicates whether the order is manually modified.

## Grouped Measures

The following table provides a list of the SO Modification grouped measures:

Measure	Description
Orders from SO Modification	This is the total number of orders manually modified after confirmation.
Orders from SO line	This is the total number of orders confirmed.
Percentage of Orders Manually Modified	This is the percentage of orders that have been manually modified after confirmation.

## SO Modification Summary Star Schema Grouping

The SO Modification Summary aggregate fact table is derived from the SO Line and SO Modification facts and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Calendar](#) (Confirmed Date)

### Measures

The following table provides a list of the SO Modification Summary measures:

Measure	Description
Modified Order	Indicates the total number of modified orders.
Confirmed Order	Indicates the total number of confirmed orders.

## Grouped Measures

The SO Modification Summary grouped measure is % Orders Manually Modified. It indicates the percentage of the orders that were manually modified.

## SO Payment Summary Star Schema Grouping

The SO Payment Summary aggregate fact table is derived from the SO Payment fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Calendar](#) (Execution Date)

## Measures

The following table provides a list of the SO Payment Summary measures:

Measure	Description
Authorized Amount	Indicates the amount that has been authorized for a payment.
Credit Amount	Indicates the amount that has been allowed for credit in a payment.
Settled Amount	Indicates the amount that has been settled for a payment.
Request Amount	Indicates the amount that has requested for a payment.
Authorization Count	Indicates the number of payments that have been authorized.
Settlement Count	Indicates the number of settlements.
Authorization Failure Count	Indicates the number of authorizations that have failed.
Settlement Failure Count	Indicates the number of settlements that have failed.

## Grouped Measures

The following table provides a list of the SO Payment Summary grouped measures:

Measure	Description
Percentage Authorization Failures	Indicates the percentage of the failed authorizations.
Percentage Settlement Failures	Indicates the percentage of the failed settlements.

## SO Hold Summary Star Schema Grouping

The SO Hold Summary aggregate fact table is derived from the SO Hold fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Hold Type](#)
- ◆ [Calendar](#) (Hold Date)

### Measures

The following table provides a list of the SO Hold Summary measures:

Measure	Description
Hold Duration	Indicates the duration for which an order is held.
Order Count	Indicates the number of orders that are on hold.

## Grouped Measures

The following table provides a list of the SO Hold Summary grouped measures:

Measure	Description
No. of Orders on Hold	Indicates the number of orders on hold.
Avg. Hold Duration	Indicates the average duration for which an order is kept on hold.

## SO Daily Category Summary Star Schema Grouping

The SO Daily Category Summary aggregate fact table is derived from the SO Line fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Product Category Hierarchy](#) (Category)
- ◆ [Calendar](#) (Transaction Date)

### Measures

The following table provides a list of the SO Category Summary measures:

Measure	Description
Shipped Count	Indicates the number of orders that have been shipped.
Shipped On Time Count	Indicates the number of orders that have been shipped on time.
Shipped Quantity	Indicates the shipped quantity.
B2C Fulfilled Revenue	Indicates the fulfilled revenue of B2C orders.
B2B Fulfilled Revenue	Indicates the fulfilled revenue of B2B orders.
Backordered Count	Indicates the number of orders that have been backordered.

## Grouped Measures

The following table provides a list of the SO Category Summary grouped measures:

Measure	Description
B2C Fulfilled Revenue (Currency)	Indicates the fulfilled revenue of a B2C order.
B2B Fulfilled Revenue (Currency)	Indicates the fulfilled revenue of a B2B order.
Percentage Shipped On Time	Indicates the percentage of the orders that have been shipped on time.



## SO Daily Region Summary Star Schema Grouping

The SO Daily Region Summary aggregate fact table is derived from the SO Line fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Address Region Hierarchy](#)
- ◆ [Calendar](#) (Transaction Date)

### Measures

The following table provides a list of the SO Daily Region Summary measures:

Measure	Description
Shipped Count	Indicates the number of orders that have been shipped.
Shipped On Time Count	Indicates the number of orders that have been shipped on time.
Shipped Quantity	Indicates the shipped quantity.
B2C Fulfilled Revenue	Indicates the fulfilled revenue of B2C orders.
B2B Fulfilled Revenue	Indicates the fulfilled revenue of B2B orders.

### Grouped Measures

The SO Daily Region Summary grouped measure is B2C Fulfilled Revenue (Currency). It indicates the fulfilled revenue of B2C orders.

## Shipment Summary Star Schema Grouping

The Shipment Summary aggregate fact table is derived from the Shipment Line fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Calendar](#) (Actual Ship Date)
- ◆ [User Ranges](#)
- ◆ [Ship Node](#)

### Measures

The following table provides a list of the Shipment Summary measures:

Measure	Description
Procured Count	Indicates the number of shipments that have been procured.

<b>Measure</b>	<b>Description</b>
BackOrdered from Node Count	Indicates the number of shipments that have been backordered from a node.
Shipped Count	Indicates the number of shipments that have been sent.
Ordered Quantity	Indicates the ordered quantity.
Delay in Shipment per Line	Indicates the delay in shipment per line.
Elapsed Time Released Ship	Indicates the time elapsed between the released date and the actual shipment date.
Shipment Number	Indicates the shipment number.
Shipped On Time Count	Indicates the number of shipments that have been sent on time.

### **Grouped Measures**

The following table provides a list of the Shipment Summary grouped measures:

<b>Measure</b>	<b>Description</b>
No. of Shipment Lines Shipped	Indicates the number of shipment lines that have been shipped.
Percentage Backordered From Node	Indicates the percentage of orders that have been backordered from a given node.
Percentage Procured	Indicates the percentage of the orders that have been procured.
No. of Shipments Shipped	Indicates the number of shipments that have been shipped.
Avg. Delay in Shipment	Indicates the average amount of delay that occurred when sending shipments.

### **SO Daily Summary Star Schema Grouping**

The SO Daily Summary aggregate fact table is derived from the SO Line fact and consists of the following dimensions:

#### **Dimensions**

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Seller](#)
- ◆ [Fulfillment Type](#)
- ◆ [Calendar](#) (Transaction Date)

## Measures

The following table provides a list of the SO Daily Summary measures:

<b>Measure</b>	<b>Description</b>
ASAP Order Count	Indicates the number of orders that are to be shipped as soon as possible.
BackOrdered Count	Indicates the number of orders that have been backordered.
BackOrdered and Delayed Count	Indicates the number of orders that have been backordered and delayed.
Released Count	Indicates the number of orders that have been released.
Shipped Count	Indicates the number of orders that have been shipped.
Shipped On Time Count	Indicates the number of orders that have been shipped on time.
Elapsed Lead Time ASAP	Indicates the time that has elapsed between the promised time and the confirmed time (in minutes).
Elapsed Time Actual ship Confirmed	Indicates the time that has elapsed between the actual shipment time and the confirmed time (in minutes).
Elapsed Time Actual ship Promised	Indicates the time that has elapsed between the actual shipment time and the promised time (in minutes).
Elapsed Time Actual ship Released	Indicates the time that has elapsed between the promised time and the released time (in minutes).
Elapsed Time Confirm Release	Indicates the time that has elapsed between the released time and confirmed time (in minutes).
Fulfilled Revenue	Indicates the fulfilled revenue of an order.
Shipped Quantity	Indicates the shipped quantity.
Promised Count	Indicates the number of orders that have been promised to be delivered.
Confirmed Order Lines	Indicates the confirmed order lines.
Ordered Quantity	Indicates the ordered quantity.
Confirmed Orders	Indicates the confirmed orders.
Booked Value	Indicates the booked value of the confirmed order.
B2B Booked Value	Indicates the booked value of the B2B orders.
B2C Booked Value	Indicates the booked value of the B2C orders.
B2B Fulfilled Value	Indicates the fulfilled value of the B2B orders.
B2C Fulfilled Value	Indicates the fulfilled value of the B2C orders.
B2B Shipped Quantity	Indicates the shipped quantity of the B2B orders.
B2C Shipped Quantity	Indicates the shipped quantity of the B2C orders.
B2B Ordered Quantity	Indicates the ordered quantity of the B2B orders.
B2C Ordered Quantity	Indicates the ordered quantity of the B2C orders.

<b>Measure</b>	<b>Description</b>
B2B Orders Confirmed	Indicates the B2B orders that have been confirmed.
B2C Orders Confirmed	Indicates the B2C orders that have been confirmed.
B2B Order Lines Confirmed	Indicates the B2B order lines that have been confirmed.
B2C Order Lines Confirmed	Indicates the B2C order lines that have been confirmed.
B2B Shipped Count	Indicates the number of B2B orders lines that have been shipped.
B2C Shipped Count	Indicates the number of B2C orders lines that have been shipped.
B2B Shipped Order Count	Indicates the number of B2B orders that have been shipped.
B2C Shipped Order Count	Indicates the number of B2C orders that have been shipped.
B2B Order Count	Indicates the number of B2B orders.
B2C Order Count	Indicates the number of B2C orders.
Order Delay	Indicates the time taken to ship an order.
Capture to Ship Time	Indicates the amount of time between the stage an order is captured and the time it is shipped.
Delayed Count	Indicates the number of orders that have been delayed.
Shipped ASAP Order Count	Indicates the number of ASAP orders that have been shipped.

### **Grouped Measures**

The following table provides a list of the SO Daily Summary grouped measures:

<b>Measure</b>	<b>Description</b>
Percentage Shipped on Time	Indicates the percentage of orders that have been shipped on time.
Percentage BackOrdered	Indicates the percentage of orders that have been backordered.
No. of Order Lines Shipped	Indicates the number of order lines that have been shipped.
Avg. Exp. Lead Time (ASAP Orders)	Indicates the average expected lead time for ASAP orders.
Avg. Capture to Ship Time (ASAP)	Indicates the average amount of time between the stage an ASAP order line is captured and the time it is shipped.
Avg. delay for Delayed Orders	Indicates the average amount of delay occurred in the delayed orders.
Avg. Order to Release time	indicates the average time required to move the order lines from the order capture stage to the order release stage.
Avg. Lead Time to Ship	Indicates the average time required to move the order lines from the release stage to the ship stage.
Avg. Order Lines per Order	Indicates the average number of order lines in a confirmed order.
Avg. Value of an Order	Indicates the average booked value of confirmed orders.

<b>Measure</b>	<b>Description</b>
No. of Orders Confirmed	Indicates the number of orders that have been confirmed.
No. of Order Lines Confirmed	Indicates the number of order lines that have been confirmed.
No. of Order Lines Shipped (B2B)	Indicates the number of shipped order lines pertaining to a B2B order.
No. of Order Lines Shipped (B2C)	Indicates the number of shipped order lines pertaining to a B2C order.
B2B Booked Value (Currency)	Indicates the booked value of a B2B order.
B2C Booked Value (Currency)	Indicates the booked value of a B2C order.
B2B Fulfilled Value (Currency)	Indicates the fulfilled value of a B2B order.
B2C Fulfilled Value (Currency)	Indicates the fulfilled value of a B2C order.
Avg. Value of an Order B2B	Indicates the average value of a B2B order.
Avg. Value of an Order B2C	Indicates the average value of a B2C order.
Avg. Order Lines per Order B2B	Indicates the average number of order lines confirmed per a B2B order.
Avg. Order Lines per Order B2C	Indicates the average number of order lines confirmed per a B2C order.
No. of Order Lines Confirmed B2B	Indicates the average number of B2B order lines confirmed.
No. of Order Lines Confirmed B2C	Indicates the average number of B2C order lines confirmed.
No. Of B2B Orders Shipped	Indicates the number of B2B orders that have been shipped.
No. Of B2C Orders Shipped	Indicates the number of B2C orders that have been shipped.

## SO Customer Monthly Summary Star Schema Grouping

The SO Customer Monthly Summary aggregate fact table is derived from the SO Line fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Customer](#)
- ◆ [Calendar](#) (Month)

## Measures

The following table provides a list of the SO Customer Monthly Summary measures:

Measure	Description
Fulfilled Revenue	Indicates the fulfilled revenue by a customer in a month.
Order Count	Indicates the total number of orders placed by a customer.

## Grouped Measures

The SO Customer Monthly Summary grouped measure is Average Order Value. It indicates the average lifetime order value for a customer.

## SO Monthly Summary

The SO Monthly Summary aggregate fact table is derived from the SO Line fact and consists of the following dimensions:

### Dimensions

- ◆ [Enterprise](#)
- ◆ [Document Type](#)
- ◆ [Calendar](#) (Month)

## Measures

The following table provides a list of the SO Monthly Summary measures:

Measure	Description
Fulfilled Revenue	Indicates the fulfilled revenue of a customer in a month.
Order Count	Indicates the total number of orders placed by a customer.
Customer Count	The number of customers for the Enterprise.

## Grouped Measures

The following table provides a list of the SO Monthly Summary grouped measures:

Measure	Description
Average Customer Revenue	Indicates the amount of revenue generated by an average customer. An average customer is the customer with whom other customers are compared in terms of revenue generated
Customer Spent	Indicates the total amount that a customer has spent.
Customer Spent Ratio	Indicates the ratio of the amount spent by a customer by the total amount.

## Target Model Dimensions

Dimensions reflect the attributes of a business, such as enterprise, channel, seller, buyer, and so on. Each dimension consists of one or more hierarchy that typically contain several levels. For example, the enterprise dimension contains the enterprise hierarchy and the corresponding levels such as Enterprise (All) and Enterprise Code.

Dimensions also provide contextual information to facts. For example, order being shipped is a fact, However, data such as the order ship date, ship to address, ship from address, and so on are contextual information that is captured by a dimension. Dimensions can be used to filter data and arrive at an appropriate data set. Sterling Business Intelligence contains the following target model dimensions.

### Enterprise

An enterprise is an organization that brokers business. An enterprise represents the organization that owns and controls all transactions. An enterprise controls the flow of documents such as a sales order, and is considered the owner of the documents. Most business rules and fulfillment processes pertaining to an order are defined by an enterprise.

#### Dimension Hierarchy

The dimension hierarchy related to enterprise is Enterprise.

The levels associated with enterprise are:

- ◆ Enterprise (All)
- ◆ Enterprise Code

Location in the target model: Enterprise

#### Filter

The filter associated with the enterprise dimension is Enterprise Code.

### Ship Node

This dimension contains all the ship nodes. Ship node is the warehouse or distribution center that is shipping the shipment or products.

#### Dimension Hierarchy

The dimension hierarchy related to ship node is Ship Node.

The levels associated with ship node are:

- ◆ Ship Node (All)
- ◆ Ship Node Code

Location in the target model: Ship Node

#### Filter

The filter associated with the ship node dimension is Ship Node Code.

#### Dimension Role Play

The dimension role play related to ship node is Ship Node (Procurement Node).

## Seller

This dimension contains all the sellers. A seller is an organization that supplies products to an enterprise or other buyer organizations.

### **Dimension Hierarchy**

The dimension hierarchy related to seller is Seller.

The levels associated with seller are:

- ◆ Seller (All)
- ◆ Seller Code

Location in the target model: Seller

### **Filter**

The filter associated with the seller dimension is Seller Code.

## Buyer

This dimension contains all the buyers. A buyer is an organization that purchases product from an enterprise or other seller organizations.

### **Dimension Hierarchy**

The dimension hierarchy related to buyer is Buyer.

The levels associated with buyer are:

- ◆ Buyer (All)
- ◆ Buyer Code

Location in the target model: Buyer

### **Filter**

The filter associated with the buyer dimension is Buyer Code.

## Payment Type

This dimension contains the credit card payment types supported by various organizations. Payment type is the type of payment that a customer uses to pay for an order, for example, check, credit card, store value card, and so on.

### **Dimension Hierarchy**

The dimension hierarchy related to payment type is Payment Type.

The levels associated with payment type are:

- ◆ Payment Type (All)
- ◆ Enterprise Code
- ◆ Payment Type Code

Location in the target model: Payment Type



## **Filter**

There is no filter associated with the payment type dimension.

## **Hold Type**

Holds are applied on orders because of exception conditions and can be used to stop order processing. For example, an order can be put on fraud hold. Holds can be applied at either the order level or the order line level that meeting a certain condition.

### **Dimension Hierarchy**

The dimension hierarchy related to hold type is Hold Type.

The levels associated with hold type are:

- ◆ Hold Type (All)
- ◆ Enterprise Code
- ◆ Hold Type Code

Location in the target model: Hold Type

## **Filter**

There is no filter associated with the hold type dimension.

## **Channel**

This dimension contains the channel. Orders are placed through various channels, for example, Web channel, catalog, call center, so on.

### **Dimension Hierarchy**

The dimension hierarchy related to channel is Channel.

The levels associated with channel are:

- ◆ Channel (All)
- ◆ Channel Code

Location in the target model: Channel

## **Filter**

The filter associated with the channel dimension is Channel Code.

## **Fulfillment Type**

An enterprise can define the list of valid values for fulfillment type, and specify them on the order line. Fulfillment type specifies how the order line will be sourced.

### **Dimension Hierarchy**

The dimension hierarchy related to fulfillment type is Fulfillment Type.

The levels associated with fulfillment type are:

- ◆ Fulfillment Type (All)

- ◆ Enterprise Code
- ◆ Fulfillment Type Code

Location in the target model: Fulfillment Type

#### **Filter**

There is no filter associated with the fulfillment type dimension.

### **Delivery Method**

This dimension contains all the delivery methods. This specifies how an order is delivered.

#### **Dimension Hierarchy**

The dimension hierarchy related to delivery method is Delivery Method.

The levels associated with delivery method are:

- ◆ Delivery Method (All)
- ◆ Enterprise Code
- ◆ Delivery Method Code

Location in the target model: Delivery Method

#### **Filter**

There is no filter associated with the delivery method dimension.

### **Product**

This dimension contains the products. A product is an item that has at least one characteristic that is different from all other products, and which requires a unique identifier such as a Stock Keeping Unit (SKU) or part number. A product is the most basic part of a catalog hierarchy.

**Note:** When an item has multiple units of measures (UOM) defined, only one unit of measure is considered for analysis in Sterling Business Intelligence.

## **Dimension Hierarchy**

The dimension hierarchy related to product are:

### ◆ Product

The levels associated with Product are:

- ◆ Product (All)
- ◆ Catalog Organization Code
- ◆ Product Code
- ◆ Unit of Measure Code

### ◆ Products

The levels associated with Products are:

- ◆ Product (All)
- ◆ Product Code

Location in the target model: Product

## **Filter**

There is no filter associated with the product dimension.

## **Product Category Hierarchy**

Items are classified based on categories. You can define an hierarchy of categories for a product.

## **Dimension Hierarchy**

The dimension hierarchy related to product category hierarchy are:

### ◆ Categories (Product)

The levels associated with Categories (Product) are:

- ◆ Categories (Product) (All)
- ◆ Top Level Categories (Product)
- ◆ Second Level Categories (Product)
- ◆ Third Level Categories (Product)
- ◆ Fourth Level Categories (Product)
- ◆ Fifth Level Categories (Product)

### ◆ Product Category Hierarchy

The levels associated with Products are:

- ◆ Product Category Hierarchy (All)
- ◆ Catalog Organization Code
- ◆ Category Identifier

Location in the target model: Product Category Hierarchy

## **Filter**

The filter associated with the product category hierarchy dimension is Product Category.

## **Customer**

This dimension contains a repository of all customers. Customers are further categorized into Business Customers (Organizations or Businesses) and Consumer Customers (Individual).

### **Dimension Hierarchy**

The dimension hierarchy related to customer is Customer.

The levels associated with customer are:

- ◆ Customer (All)
- ◆ Customer Organization Code
- ◆ Customer Identifier

Location in the target model: Customer

### **Filters**

There is no filter associated with the customer dimension.

## **Customer Hierarchy**

This dimension contains the entire customer hierarchy. It signifies the hierarchy of customers. Customers are further categorized into Business Customers (Organizations or Businesses) and Consumer Customers (Individual). Sterling Business Intelligence supports up to five levels of customer hierarchy.

### **Dimension Hierarchy**

The dimension hierarchy related to customer hierarchy is Customer Hierarchy.

The levels associated with customer hierarchy are:

- ◆ Customer Hierarchy

The levels associated with customer hierarchy are:

- ◆ Customer Hierarchy (All)
- ◆ Customer Organization Code
- ◆ Customer Identifier Customers

- ◆ Customers

The levels associated with customers are:

- ◆ Customers (All)
- ◆ Top Level Customers
- ◆ Second Level Customers
- ◆ Third Level Customers
- ◆ Fourth Level Customers

- ◆ Fifth Level Customers

Location in the target model: Customer Hierarchy

### **Filters**

The filters associated with the customer hierarchy dimension are Business Customer and Consumer Customer.

## **Customer Contact**

Customer contacts represent individuals at a customer location. You can define multiple contacts for each customer.

### **Dimension Hierarchy**

The dimension hierarchy related to customer contact is Customer Contact.

The levels associated with customer contact are:

- ◆ Customer Contact (All)
- ◆ Customer Identifier
- ◆ Customer Contact Identifier
- ◆ Customer Organization Code

Location in the target model: Customer Contact

### **Filter**

There is no filter associated with the Customer Contact dimension.

## **Address Region**

This dimension contains the address of a customer.

### **Dimension Hierarchy**

The dimension hierarchy related to address region is Address Region.

The levels associated with address region are:

- ◆ Address Region (All)
- ◆ Address Identifier

Location in the target model: Address Region

### **Filter**

There is no filter associated with the address region dimension.

### **Dimension Role Play**

The dimension role plays related to address region are:

- ◆ Address Region (Ship To)
- ◆ Address Region (Bill To)

## Address Region Hierarchy

This dimension contains the region hierarchy corresponding to an address. A typical region hierarchy would be Country > Regions (Territory) > State > City > Zip Code.

### Dimension Hierarchy

The dimension hierarchy related to address region hierarchy is Address Region Hierarchy.

The levels associated with address region hierarchy are:

- ◆ Address Region Hierarchy

The levels associated with address region hierarchy are:

- ◆ Address Region Hierarchy (All)
- ◆ Region Identifier

- ◆ Region Hierarchy (Address)

The levels associated with region hierarchy (Address) are:

- ◆ Region Hierarchy (Address) (All)
- ◆ Top Level Address Region
- ◆ Second Level Address Region
- ◆ Third Level Address Region
- ◆ Fourth Level Address Region
- ◆ Fifth Level Address Region

Location in the target model: Address Region Hierarchy

### Filter

The filter associated with the address region hierarchy dimension is Region Hierarchy.

### Dimension Role Play

The dimension role plays related to address region hierarchy are:

- ◆ Address Region Hierarchy (Ship To)
- ◆ Address Region Hierarchy (Bill To)

## Document Type

This dimension contains the document type. Document type is a kind of document defined by a specific XML template that has been designed to support a specific transaction set or business process. The document type defines possible processes and system behaviors that occur as the document goes through its life cycle.

### Dimension Hierarchy

The dimension hierarchy related to document type is Document Type.

The levels associated with document type are:

- ◆ Document Type (All)

- ◆ Document Type Code

Location in the target model: Document Type

#### **Filter**

The filter associated with the document type dimension is Document Type Code.

### **User Ranges**

This dimension contains different range types and their possible range values. Customers can extend this dimension to define their own range type and analyze based on the defined range type.

#### **Dimension Hierarchy**

The dimension hierarchy related to user ranges is User Ranges.

The levels associated with user ranges are:

- ◆ User Ranges (All)
- ◆ Range Identifier
- ◆ RangeType Code
- ◆ Range (Days)

Location in the target model: User Ranges

#### **Filters**

There is no filter associated with the user ranges dimension.

### **Calendar**

This dimension contains the default year, quarter, month, and day for the Gregorian calendar. The dimension hierarchy related to calendar and its role play's are:

- ◆ Calendar

The levels associated with Calendar are:

- ◆ Calendar YMD
- ◆ Calendar YQMD

- ◆ Calendar (Charge Date)

The levels associated with Calendar (Charge Date) are:

- ◆ Calendar (Charge Date) YMD
- ◆ Calendar (Charge Date) YQMD

- ◆ Calendar (Modify Date)

The levels associated with Calendar (Modify Date) are:

- ◆ Calendar (Modify Date) YMD
- ◆ Calendar (Modify Date) YQMD

- ◆ Calendar (Hold Date)
  - The levels associated with Calendar (Hold Date) are:
    - ◆ Calendar (Hold Date) YMD
    - ◆ Calendar (Hold Date) YQMD
- ◆ Calendar (Actual Ship Date-SO Line)
  - The levels associated with Calendar (Actual Ship Date-SO Line) are:
    - ◆ Calendar (Actual Ship Date-SO Line) YMD
    - ◆ Calendar (Actual Ship Date-SO Line) YQMD
    - ◆ Calendar (Actual Ship Date-SO Line) YWD
- ◆ Calendar (Released Date)
  - The levels associated with Calendar (Released Date) are:
    - ◆ Calendar (Released Date) YMD
    - ◆ Calendar (Released Date) YQMD
- ◆ Calendar (Confirmed Date)
  - The levels associated with Calendar (Confirmed Date) are:
    - ◆ Calendar (Confirmed Date) YMD
    - ◆ Calendar (Confirmed Date) YQMD
    - ◆ Calendar (Confirmed Date) YWD
    - ◆ Calendar (Confirmed Date) MD
- ◆ Calendar (Actual Ship Date-Shipment Line)
  - The levels associated with Calendar (Actual Ship Date-Shipment Line) are:
    - ◆ Calendar (Actual Ship Date-Shipment Line) YMD
    - ◆ Calendar (Actual Ship Date-Shipment Line) YQMD
    - ◆ Calendar (Actual Ship Date-Shipment Line) MD
    - ◆ Calendar (Actual Ship Date-Shipment Line) YWD
- ◆ Calendar (Expected Shipment Date)
  - The levels associated with Calendar (Expected Shipment Date) are:
    - ◆ Calendar (Expected Shipment Date) YMD
    - ◆ Calendar (Expected Shipment Date) YQMD
- ◆ Calendar (Promised Ship Date)
  - The levels associated with Calendar (Promised Ship Date) are:
    - ◆ Calendar (Promised Ship Date) YMD
    - ◆ Calendar (Promised Ship Date) YQMD

Location in the target model: Calendar



## Filter

There is no filter associated with the calendar dimension.

## Parameter Maps

Parameter maps are objects that store key-value pairs. Use parameters to create conditional query subjects that allow for substitutions when a report is run. Parameter maps are similar to data source look-up tables. Each parameter map has two columns, one for the key and one for the value that the key represents.

Following are the parameter maps for Sterling Business Intelligence:

- ◆ Calendar Quarter Start Date
- ◆ Calendar Quarter End Date
- ◆ Product Category Hierarchy
- ◆ Sales Order Hold
- ◆ SO Hold
- ◆ Payment
- ◆ Dimension Perspective
- ◆ Low Date Map
- ◆ Locale Code Map
- ◆ SO Payment
- ◆ AA\_Prompts\_Map
- ◆ Relative Time Prompt Map
- ◆ Product Category Tree
- ◆ Shipment Line
- ◆ Sales Order Modification
- ◆ CurrentDateMap
- ◆ SO Modification

---

## Troubleshooting

You may encounter the following issues when using Sterling Business Intelligence. Use the information provided here to solve some of the specific problems that you may encounter.

For additional troubleshooting tips, it is recommended that you refer to the *IBM® Cognos® 8 Analytic Applications Adaptive Warehouse User Guide*, the *IBM® Cognos® 8 Analytic Applications Adaptive Analytics User Guide*, and the *IBM® Cognos® 8 Analytic Applications Installation and Configuration Guide*.

### Load Management Process is Stalled

The load management process is stalled when the load process is restarted after the previous run is terminated abnormally.

On analyzing the `aw-etl<timestamp>.log` file, it has been found that, queries such as `SELECT count(*) FROM PWP_ETL_LOCKTABLE WHERE WAREHOUSE_OBJECT_ID = '420f9506-93a3-11de-b501-9a43165eb0f8'` are repeated periodically.

#### Resolving the Problem

The lock rows must exist only for the duration of a build, after the load management process, system will release them. If there is an abnormal termination of this process, the lock rows remain in the table. The application then truncates the `PWP_ETL_LOCKTABLE` table.

If job streams are manually halted, or the application fails for some reason, then the `PWP_ETL_LOCKTABLE` table must be truncated manually. You must ensure that no application activity is taking place when truncating the table manually.

### Report Execution Displays an RSV-BBP-0022 Error

When reports are executed in Microsoft® Internet Explorer, an RSV-BBP-0022 error is displayed stating the following:

```
The absolute affinity request 'asynchWait_Request' fails, the requested session does not exist.
```

#### Resolving the Problem

Ensure that the IBM® Cognos® URL (`http://<hostname>/cognos8/cgi-bin/cognos.cgi`) has not been added to the Restricted sites list in the Internet Explorer Options dialog box (Tools > Internet Options > Security > Restricted sites).

## Schema Not Displayed When Creating an Adaptive Warehouse Project

When creating an Adaptive Warehouse project, after selecting the Sterling Data Source project, no target schema is displayed. You will thus not be able to create the Adaptive Warehouse project.

### Resolving the Problem

Launch IBM Cognos Configuration and click **Save**.

## ETL Script Running in CLI Stalls in an IBM AIX Environment

The `aw-timestamp.log` file contains two entries that are similar to `Running script <script_name>`  
EEM-INFO-0001 Running script: <script\_name>.

When executing a load with IBM Cognos Adaptive Warehouse Runtime on IBM AIX 5.3, the load management is not executed and no error is displayed.

### Resolving the Problem

Download and install the December 2007 IBM XL C/C++ runtime program temporary fix (PTF) from the IBM Web site. Run the ETL script after you have installed the PTF.

## ETL Process Does Not Load All the Data into the Data Mart Database

Orders created in the online transaction processing (OLTP) database after the last ETL process, is not loaded to the Data Mart Database, despite being in the change data capture (CDC) time window.

### Resolving the Problem

Check if the system time of Cognos Server is ahead of OLTP DB Server. If this is the case, you must ensure that all servers are time synched using services like Network Time Protocol (NTP) to a single clock source. Restart the ETL process by resetting the CDC Time Interval. In case the Platform considerations do not allow for time synching, the Cognos server should run one minute slower than the OLTP server.

- B**
- backup 19
- C**
- catalog 8
  - CDC filters 14
  - configure
    - data mart project 8
    - install locale 8
    - purge criteria 9
    - staging project 7
    - Sterling Business Intelligence Application 9
- D**
- data mart database 14
- E**
- extend
    - adaptive analytics reports 35
    - adaptive warehouse content 34
    - dashboards 35
  - extend and customize
    - CSS file 36
    - menu 36
    - user interface 36
- I**
- incremental data load 26
    - data mart project 29
    - staging project 26
  - initial data load 20
    - data mart project 24
- L**
- load interval 15
  - load management 14
- M**
- map address to a region 9
- P**
- purge 13
    - order timestamp tag 10
    - shipment timestamp tag 11
  - purge criteria
    - order timestamp tag 10
    - shipment tag timestamp 12
- S**
- staging database 14
- T**
- Troubleshooting 88