Scenario Wizard
Configuration Guide

Release 6.2.3
May 2014

ORACLE
FINANCIAL SERVICES

ORACLE

Scenario Wizard
Configuration Guide

Release 6.2.3
May 2014

Document Control Number: 9MN12-62310021
Document Number: SWCG-14-FCCM-0021-6.2.3-01

Oracle Financial Services Software, Inc.
1900 Oracle Way
Reston, VA 20190

Document Number: SWCG-14-FCCM-0021-6.2.3-01
First Edition (May 2014)

Copyright 2014, Oracle and/or its affiliates. All rights reserved.
Printed in U.S.A No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise without the prior written permission.

Trademarks
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.
Other names may be trademarks of their respective owners.

Oracle Financial Services Software, Inc.
1900 Oracle Way
Reston, VA 20190
Phone: 703-478-9000
Fax: 703-318-6340
Internet: www.oracle.com/financialservices

Contents

List of Figures ..... ***ix***

List of Tables ..... ***xi***

About this Guide ..... ***xiii***

Who Should Use this Guide xiii
How this Guide is Organized xiii
Where to Find More Information xiv
Conventions Used in this Guide xiv

CHAPTER 1 ***About Scenario Wizard*** 1

Introduction 1
Features 1

CHAPTER 2 ***Template XML Files*** 3

Overview 3
Introduction to XML files 3
Mapping Relationship between XML Files 4
Configuring the Template XML files 5

CHAPTER 3 ***Scenario Focus Mapping (ScenarioFocusMapping.xml)*** 7

Tags 8
<scenarioFocusMap> 8
<scenarioClass> 8
<focus> 9
<focus_name> 10
<focus_code> 10
<associatedRecord> 11
<record_name> 13
<table> 13
<table_name> 13
<display_name> 13
<isPrimary> 14
<isSelected> 14

CHAPTER 4 ***Threshold (threshold.xml)*** 15

Pre-defined Threshold 16

Tags 17
<threshold-data> 17
<threshold_list> 17
<tshld_name> 17
<min_value> 17
<max_value> 17
<default_value> 18
<value_type_code> 18
<value_unit> 18
<current_value> 19
<display_name> 19
<description> 19

CHAPTER 5 *TA Column Map (TAColumnMap.xml)* 21

User-defined Highlight 22
Tags 23
<taScnroGroupMap> 23
<taColumnMap> 24
<taColumnDef> 24
<columnName> 24
<displayName> 24
<type> 24
<precision> 24

CHAPTER 6 *Data Set Rule Map (datasetrulemap.xml)* 25

Primary Rule Dataset 28
Dataset Attributes for Primary Rule Dataset 29
Pre-defined Highlight 30
Pre-defined Threshold 31
Primary Rule Dataset Query 32
Tags 37
<dataSetRuleMap> 37
<datasetsql> 37
<scnro_class_cd> 37
<attribute> 37
<attribute_key_flag> 38
<attribute_name_flag> 38
<focalAttribute> 38
<tableList> 38
<tableName> 39
<primaryRule> 39
<threshold> 39
<tshld_name> 39

<primary_rule_name>....39
<primary_rule_desc>....40
<role>....40
<defaultTestColName>....40
Test Scenario - Default Test Result Columns....41
Sub-Rule Dataset....43

APPENDIX A ***Embedding Queries in XML....45***

APPENDIX B ***Using Functions in Query....47***

APPENDIX C ***Nested Query....51***

List of Figures

Figure 1. Relationship between XML files 4
Figure 2. Scenario Focus Mapping 7
Figure 3. Threshold 15
Figure 4. TA Column Map 22
Figure 5. Data Set Rule Map 26

List of Tables

Table 1. Conventions Used in this Guide xiv
Table 2. List of Scenario Classes 8
Table 3. Scenario class group and the TA Data 21

About this Guide

This guide explains the structure behind the Oracle Financial Services Scenario Wizard and provides comprehensive instructions for configuring modifiable components.

This chapter focuses on the following topics:

- Who Should Use this Guide
- How this Guide is Organized
- Where to Find More Information
- Conventions Used in this Guide

Who Should Use this Guide

The *Scenario Wizard Configuration Guide* is designed for use by application users and client personnel who have a working knowledge of eXtensible Markup Language (XML) and UI software components. Their roles and responsibilities, as they operate within the Oracle Financial Services Enterprise Case Management, include the following:

- **Data Miner:** Configures, maintains, and adjusts the Oracle Financial Services Behavior Detection Framework and Oracle Financial Services Alert Management user interface, and is usually an employee of a specific Oracle Financial Services customer.

How this Guide is Organized

The *Scenario Wizard Configuration Guide* includes the following chapters:

- Chapter 1, *About Scenario Wizard,*, provides a brief introduction about the Oracle Financial Services Scenario Wizard tool.
- Chapter 2, *Template XML Files*,, explains the Template XML files, their structure, content, and inter-relationship. It also describes the configurations that need to be done for scaling out-of-the-box options.
- Chapter 3, *Scenario Focus Mapping (ScenarioFocusMapping.xml),*, explains about the different configurations using `ScenarioFocusMapping.xml` file.
- Chapter 4, *Threshold (threshold.xml),*, describes the tags used by the `threshold.xml` file.
- Chapter 5, *TA Column Map (TAColumnMap.xml),*, describes the tags used by the `TAColumnMap.xml` file.
- Chapter 6, *Data Set Rule Map (datasetrulemap.xml),*, explains about the different configurations using `datasetrulemap.xml` file.
- Appendix A, *Embedding Queries in XML,* shows how to edit XML with simple SQL query.
- Appendix B, *Using Functions in Query,* shows the usage of aggregate functions inside SQL query.
- Appendix C, *Nested Query,* shows the usage of nested query inside the SQL.

Where to Find More Information

For more information about Oracle Financial Services Behavior Detection Framework, refer to the following documents:

- *Administration Guide*
- *Administration Tools User Guide*
- *Scenario Manager User Guide*
- *Installation Guide - Stage 1*
- *Installation Guide - Stage 3*
- *Alert Management User Guide*

To learn more about Oracle Financial Services and our complete product line, refer to our Web site www.oracle.com/financialservices.

Conventions Used in this Guide

Table 1 lists the conventions used in this guide.

Table 1. Conventions Used in this Guide

Convention	Meaning
Italics	• Names of books, chapters, and sections as references • Emphasis
Bold	• Object of an action (menu names, field names, options, button names) in a step-by-step procedure • Commands typed at a prompt • User input
`Monospace`	• Directories and subdirectories • File names and extensions • Process names • Code sample, including keywords and variables within text and as separate paragraphs, and user-defined program elements within text
<Variable>	Substitute input value

CHAPTER 1 About Scenario Wizard

The Oracle Financial Services Scenario Wizard is a tool that allows business users to create scenarios that run within the Oracle Financial Services Behavior Detection Framework.

Introduction

Scenario wizard provides a sequence of screens that leads the user through a series of well-defined steps for creating or editing a scenario. Every screen exposes only that information which can easily be understood by any business user.

At any point the user should enter or select only business related information and is shielded from complex technicalities involved with any scenario. The complete logic of the scenario is built by the system dynamically, based on the user inputs and even without the user realizing it. In effect, any business logic is executed as one or more SQL queries and these queries are constructed by the system for the user.

Features

- Scenarios are created based on pre-defined templates of scenario classes, focuses and combinations of data that may be examined for behaviors of interest.
- Wizard-like interface allows novice users to create fully functional scenarios
- Easy launch option using Java Web Start (from admin tools)
- Architected using web service to ensure UI remains technology agnostic
- SSL communication ensures confidential data is secure over wire
- Ability to execute test runs of the created scenario against business data

CHAPTER 2 Template XML Files

This chapter explains the Template XML files, their structure, content, and inter-relationship. It also describes the configurations that should be done for scaling out-of-the-box options. These template XML files provide the base framework on which the Scenario Wizard runs.

This chapter discusses the following topics:

- Overview
- Introduction to XML files
- Mapping Relationship between XML Files
- Configuring the Template XML files

Overview

All the dynamic data (information and definition) that display in the Scenario Wizard UI is configured in the Template XML files. These XML files also contain the SQL queries for datasets, relationships, and mapping information required for building a valid scenario that can be executed by the algorithm for generating alerts.

Introduction to XML files

The Scenario Wizard uses the following Template XML files as part of the base template framework:

- `ScenarioFocusMapping.xml` - Mapping between scenario class, focus, and associated data. For more information, refer to *Scenario Focus Mapping (ScenarioFocusMapping.xml)*, on page 7.
- `datasetrulemap.xml` - Contains dataset SQL, dataset attributes, mapping for thresholds/highlights for scenarios. For more information, refer to *Data Set Rule Map (datasetrulemap.xml)* on page 25.
- `threshold.xml` - Metadata for pre-defined thresholds. For more information, refer to*Threshold (threshold.xml)* on page 15.
- `TAColumnMap.xml` - Metadata for user-defined highlights. For more information, refer *TA Column Map (TAColumnMap.xml)* on page 21.

Mapping Relationship between XML Files

Every template XML file contains a specific type of information to be used by the system to build a scenario. Some template XML files are independent units whereas some are dependent on others.

The following figure illustrates the relationship among the XML files:

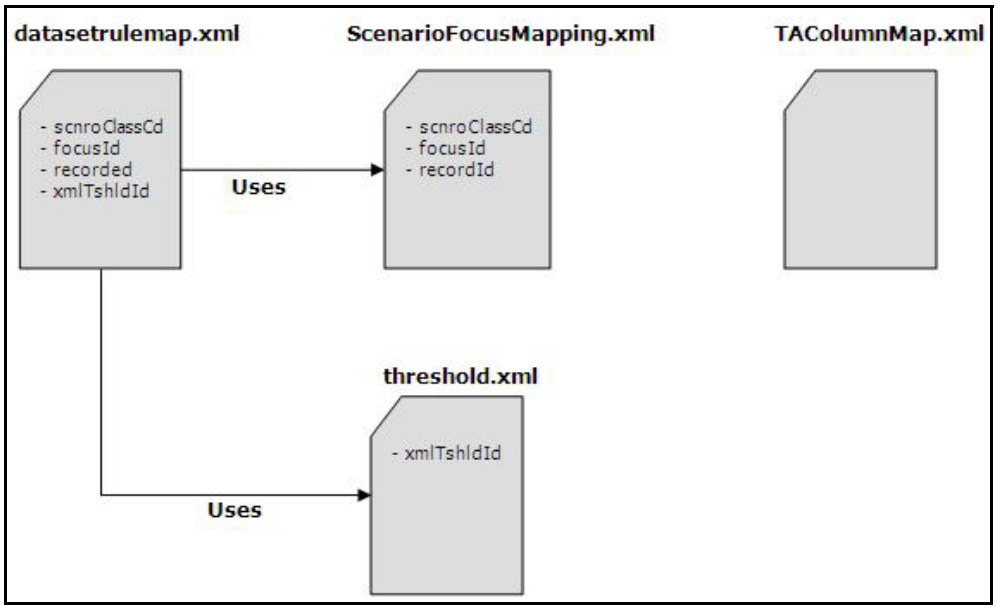

Figure 1. Relationship between XML files

The `TAColumnMap.xml` file contains information regarding user-defined highlights. It is not related to the other three XML files in any way. It neither looks up information nor it is referred from other files.

The `Threshold.xml` file contains definitions for pre-defined thresholds. It is complete in itself and does not look up any other XML file for information. However, this file is not independent. The thresholds defined here are referred in the `datasetrulemap.xml` file using `xmlTshldId`.

The `ScenarioFocusMapping.xml` file specifies various combinations of scenario class, focus, and associated data. It also lists all the tables included with an associated data. This XML file is dependent on the `Datasetrulemap.xml` file for actual business logic for a specified combination of scenario class, focus, and associated data. The system uses the `scnroClassCd, focusId`, and `recordId` fields for mapping between the `ScenarioFocusMapping.xml` and `datasetrulemap.xml`files.

Configuring the Template XML files

The Scenario Wizard application supports deployment on Tomcat, WebLogic and WebSphere servers. While installing Scenario Wizard, you must specify the name *SMLiteWeb* for the context root of the application.

As per the business need, you may modify template XML files to change the out-of-box scenario or configure a new scenario.

In order for the Scenario Wizard UI to reflect the changes made, follow these steps:

1. Login to the Tomcat, WebLogic or WebSphere sever.
2. Stop the server (if it is already running).
3. Change the directory to the following path:
   - For Tomcat Server:
     `$CATALINA_HOME/webapps/<context root specified during installation>/WEB-INF/classes/.`
   - For WebLogic Server:
     `<WebLogic Server Home>/profiles/profile_name/<WebLogic Domain Name>/applications/SMLiteWeb.war/WEB-INF/classes/`
   - For WebSphere Server:
     `<WebSphere Server Home>/profiles/Profile_name/installedApps/Profile_name/SMLiteWeb_war.ear/SMLiteWeb.war/WEB-INF/classes`
4. Verify if the present directory contains a folder named *resources*. If not, create this new folder.
5. Copy the modified template files to the *resources* folder.
   - For Tomcat Server:
     `($CATALINA_HOME/webapps/<context root specified during installation>/WEB-INF/classes/resources/)`
   - For WebLogic Server:
     `(<WebLogic Server Home>/profiles/profile_name/<WebLogic Domain Name>/applications/SMLiteWeb.war/WEB-INF/classes/ resources/)`
   - For WebSphere Server:
     `(<WebSphere Server Home>/profiles/Profile_name/installedApps/Profile_name/SMLiteWeb_war.ear/SMLiteWeb.war/WEB-INF/classes/resources/)`
6. Start the server.
   The Scenario Wizard UI displays the modified or added scenarios and any related entities.

CHAPTER 3 Scenario Focus Mapping (ScenarioFocusMapping.xml)

This chapter focuses on Scenario focus mapping. It explains about the different configurations using `ScenarioFocusMapping.xml` file.

In Scenario Wizard, a scenario is uniquely identified by its scenario class, focus, and the associated data. It also allows configuring new scenario classes, focuses, and associated data, which results in newer combinations defining newer scenarios.

The `ScenarioFocusMapping.xml` file allows the following configurations:

- Available scenario classes
- Allowed focuses for a specific scenario class
- Associated data options for a specific scenario class and focus combination
- Records to match for a particular associated data

The following figure illustrates the structure of `ScenarioFocusMapping.xml` file:

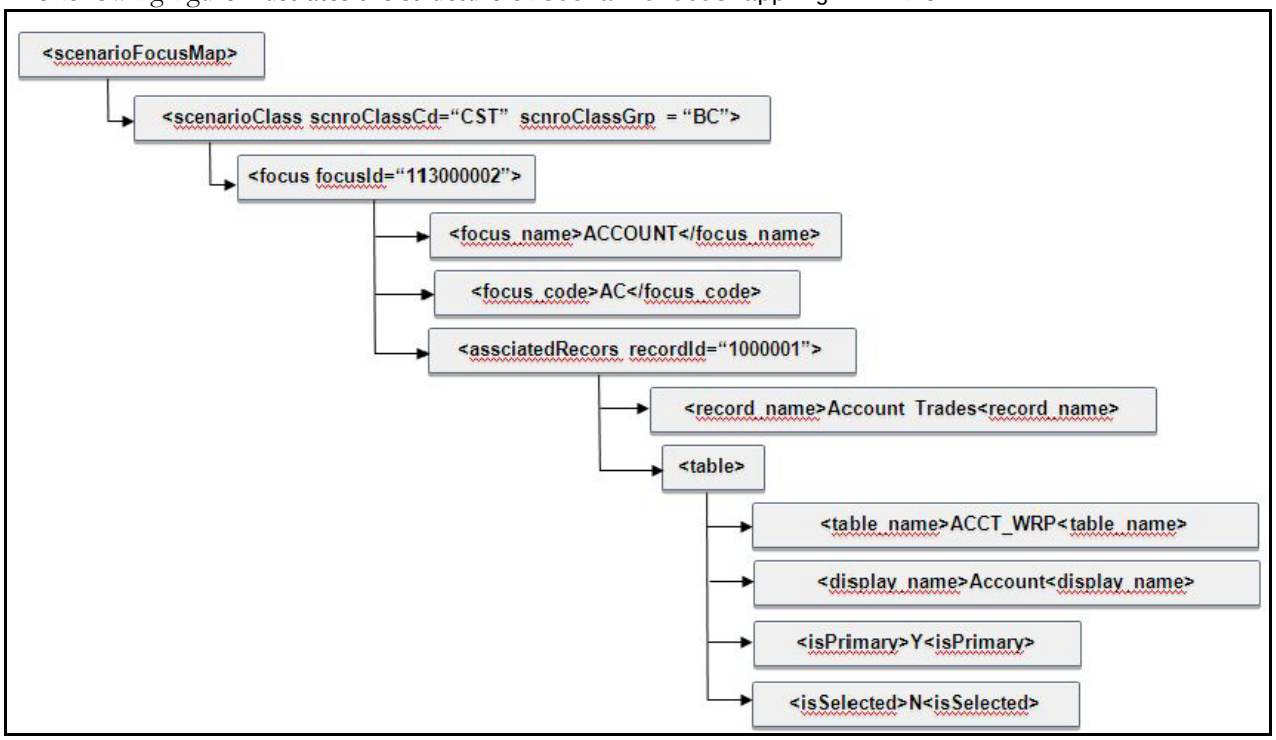

Figure 2. Scenario Focus Mapping

Tags

The following sections describe the tags in the `ScenarioFocusMapping.xml` file:

<scenarioFocusMap>

This is the root tag for the `ScenarioFocusMapping.xml` file.

Attributes - none

Child Tags - `<scenarioClass>`

Occurrences - 1

<scenarioClass>

This tag contains scenario class information and can appear multiple times in the .xml file. Each entry corresponds to a distinct scenario class. The number of times this tag appears in the XML file decides the number of scenario classes available in the Scenario Wizard.

The following table describes the available scenario classes which can be configured for the Scenario Wizard.

Table 2. List of Scenario Classes

SCNRO_CLASS_NM	SCNRO_CLASS_CD	SCNRO_CLASS_GRP
Trading Compliance	TC	TC
Best Execution	BEX	TC
Money Laundering	ML	ML
Institutional Money Laundering	IML	ML
Fraud	FR	ML
Customer Risk and Suitability	CST	BC
Mutual Fund Trading Abuses	MF	BC
Employee Trading	ET	BC
Investment Advisor	IA	BC
Asset Management	AM	BC
Control Room	CR	CR
Trading and Sales Abuses	TRA	TC

For every scenario class, add a `<scenarioClass>` tag.

Example of Scenario Class

```
# Customer Risk and Suitability
<scenarioClass scnroClassCd="CST" scnroClassGrp="BC">
#
# <scenarioClass> is the parent tag for other related
information like focuses for a specific Scenario
# Class
#
</scenarioClass>
```

Attributes -

1. `scnroClassCd` - Every scenario class has a scenario class code and the allowed values are listed in the `KDD_SCNRO_CLASS_CD` table. This attribute contains the scenario class code for a particular scenario class.
2. `scnroClassGrp` - Available scenario classes are categorized under certain scenario class groups. This attribute contains the scenario group for a particular scenario class.

Child Tags - `<focus>`

Occurrences - 1…N

<focus>

This is the child tag for the `<scenarioClass>` tag and there can be multiple occurrences of this tag. The number of times this tag appears inside the `<scenarioClass>` tag determines the number of available focuses for a particular scenario class.

Scenario Wizard displays focuses based on the scenario class. For every available focus of a scenario class, add a `<focus>` tag.

```
# Customer Risk and Suitability
<scenarioClass scnroClassCd="CST" scnroClassGrp="BC">

       # Account focus
        <focus focusId="113000002">
               <focus_name>ACCOUNT</focus_name>
               <focus_code>AC</focus_code>
        #
       # <focus> is the parent tag for other related information like associated data for a
particular
        # Scenario Class and Focus combination
        #
        </focus>

       # Customer focus
        <focus focusId="113000004">
               <focus_name>CUSTOMER</focus_name>
               <focus_code>CU</focus_code>
        #
        # associated data information for CST_CU
        #
        </focus>
</scenarioClass>
```

Attributes -

1. `focusId` - Every allowed focus has an ID associated and which is read from the `KDD_CENTRICITY` table.

Child Tags - `<focus_name>, <focus_code>, <associatedRecord>`

Occurrences - 1…N

<focus_name>

This tag contains the display name for a particular focus ID.

Attributes - none

Child Tags - none

Occurrences - 1

<focus_code>

Every focus has an associated focus code in the `KDD_CENTRICITY` table and the focus code is entered in this tag.

Attributes - none

Child Tags - none

Occurrences – 1

<associatedRecord>

Associated records for particular scenario class and focus combinations are configured using this tag. This can appear multiple times inside the <focus> tag.

Scenario Wizard displays Associated Data options based on the scenario class and focus selected. For every available associated data for a specific scenario class and focus, add an <associatedRecord> tag.

```
# Customer Risk and Suitability
<scenarioClass scnroClassCd="CST" scnroClassGrp="BC">

        # Account focus
        <focus focusId="113000002">
               <focus_name>ACCOUNT</focus_name>
               <focus_code>AC</focus_code>

               # associated data Account Trades for CST-AC
               <associatedRecord recordId="1000001">
                     <record_name>Account Trades</record_name>
                     #
                       # <associatedRecord> tag is the parent tag for other details related
to associated data,
                     # like tables for primary rule dataset and sub rule datasets. In this
case, it lists tables
                     # of CST_AC_AccountTrades
                     #
               </associatedRecord>

               # associated data Account Positions for CST_AC
               <associatedRecord recordId="1000002">
                     <record_name>Account Positions</record_name>
                     #
                     # Lists of tables for CST_AC_AccountPositions
                     #
               </associatedRecord>

        </focus>
</scenarioClass>
```

Every associated data is a collection of database tables that are used in the primary rule dataset and sub-rules datasets.

You cannot unselect primary rule dataset tables from the Scenario Wizard UI. You may select more sub-rule tables for additional building blocks. The Scenario Wizard UI extracts table information from the <associatedRecord> tag. The entire primary rule tables and sub-rule tables are specified within this tag.

```
# Customer Risk and Suitability
<scenarioClass scnroClassCd="CST" scnroClassGrp="BC">
 # Account focus
   <focus focusId="113000002">
      <focus_name>ACCOUNT</focus_name>
      <focus_code>AC</focus_code>
                        # associated data Account Trades for CST_AC
          <associatedRecord recordId="1000001">
          <record_name>Account Trades</record_name>
           #
          # List of tables for CST_AC_AccountTrades
             # Account Table
              <table>
                <table_name>ACCT_WRP</table_name>
                <display_name>Account</display_name>
                <isPrimary>Y</isPrimary>
                <isSelected>N</isSelected>
                </table>
                                  # Trade Table
                <table>
                <table_name>TRADE_EXCTN_DTLS_VW_WRP</table_name>
                       <display_name>Trade</display_name>
                       <isPrimary>Y</isPrimary>
                       <isSelected>N</isSelected>
                     </table>
                        # Account Balance
        <table>
        <table_name>ACCT_BAL_POSN_SMRY_WRP</table_name>
        <display_name>Account Balance</display_name>
                <isPrimary>N</isPrimary>
                <isSelected>N</isSelected>
                 </table>
          # Account Balance
            <table>
             <table_name>ACCT_SMRY_MNTH_WRP</table_name>
<display_name>Account Summary</display_name>         <isPrimary>N</isPrimary>
                 <isSelected>N</isSelected>
                     </table>
          # Investment Advisor          <table>
          <table_name>NVSMT_MGR_WRP</table_name>
          <display_name>Investment Advisor</display_name>
          <isPrimary>N</isPrimary>
          <isSelected>N</isSelected>
                     </table>
               </associatedRecord>
        </focus></scenarioClass
```

Attributes -

- `recordId`- It is unique identifier associated with every entry of the `<associatedRecord>` tag. Its value is entered as per user discretion; the only condition is that the value should be unique across all the `<associatedRecord>` tags in the XML.

Child Tags - `<record_name>, <table>`

Occurrences - 1…N

<record_name>

This tag contains the display name for the associated data.

Attributes - none

Child Tags - none

Occurrences - 1

<table>

This is the child tag for the `<associatedRecord>` tag and it can appear multiple times. Each mention of this tag corresponds to one of the table wrappers available. The number of times this tag is repeated decides the number of tables that are part of the dataset logic.

Note: There can be multiple datasets per associated record but only one of them is the primary dataset.

Attributes - none

Child Tags - `<table_name>, <display_name>, <isPrimary>, <isSelected>`

Occurrences - 1…N

<table_name>

This tag contains the exact name of the table wrapper from the database. You must verify that the name entered is of the table wrappers only. While building the dataset queries, the system replaces the wrapper name with the corresponding `REALIZED_NM` from the `KDD_DATASET` table.

Attributes - none

Child Tags - none

Occurrences - 1

<display_name>

This tag contains the display name for the particular table wrapper entered in the `<table_name>` tag.

Attributes - none

Child Tags - none

Occurrences - 1

<isPrimary>

This tag is the indicator flag, which indicates whether a table is part of the primary dataset. Eligible values are N or Y. For primary dataset tables, the value of this tag is *Y*. For tables exposed for the sub-rule dataset, the value of this tag is *N*.

Attributes - none

Child Tags - none

Occurrences - 1

<isSelected>

This tag is the indicator flag, which indicates whether a sub-rule table is displayed as pre-selected on the Scenario Wizard UI. Eligible values are N or Y. For pre-selected sub-rule tables, the value of this tag is Y.

Note: If the value of the `<isPrimary>` tag is N, then the `<isSelected>` tag is used. If the value of the `<isPrimary>` tag is Y, then the content of this tag is N.

Attributes - none

Child Tags - none

Occurrences - 1

CHAPTER 4 Threshold (threshold.xml)

This chapter describes the tags used by the `threshold.xml` file.

All the pre-defined thresholds are declared and defined in the threshold.xml file. This template XML file contains all the information about a threshold. The mapping of a threshold to a particular combination of scenario class, focus, and associated data is specified in `datasetrulemap.xml` file.

In order to add a new pre-defined threshold to the Scenario Wizard, you must declare and define the new threshold in the `threshold.xml` file.

The following figure illustrates the structure of `Threshold.xml`.

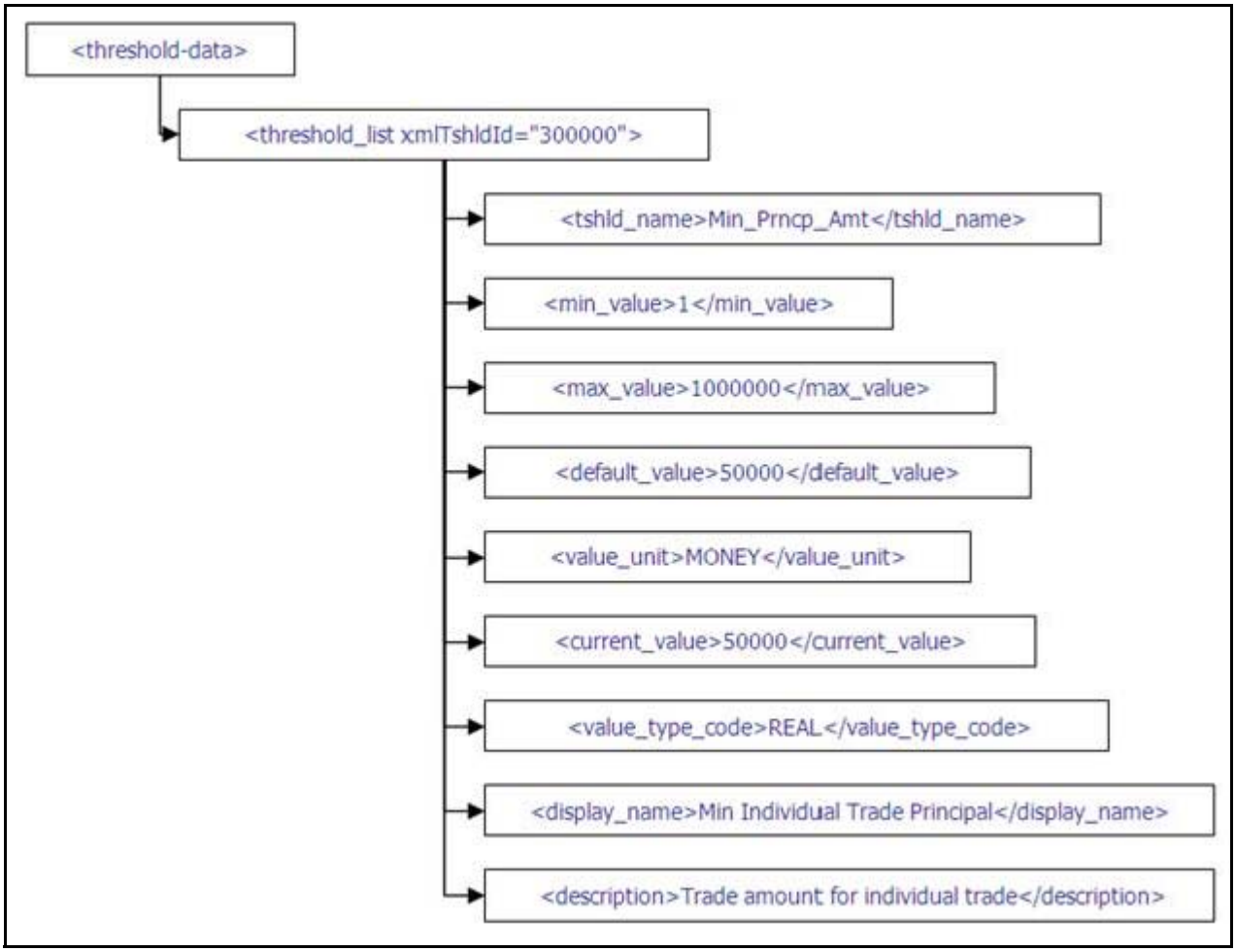

Figure 3. Threshold

Pre-defined Threshold

The Scenario Wizard creates SQL queries for the primary rule dataset and sub-rule dataset internally. These queries contain some pre-defined thresholds. You cannot delete these thresholds from the UI, only their values can be modified.

Note: The thresholds are defined in the `threshold.xml` file and then referred in the `datasetrulemap.xml` file.

Every threshold is defined independently, and each definition must have a `<threshold_list> tag.`

```
# Incl_Jurisdictions_Lst
<threshold_list xmlTshldId="300000">
#
# <threshold_list> is the parent tag that contains all the information related to a thresh-
old.
#
     <tshld_name>Incl_Jurisdictions_Lst</tshld_name>
     <min_value></min_value>
     <max_value></max_value>
     <default_value>'ALL'</default_value>
     <value_unit>NO_UNIT</value_unit>
     <current_value>'ALL'</current_value>
     <value_type_code>LIST</value_type_code>
     <display_name>Included Jurisdictions</display_name>
     <description>The list of jurisdiction codes covered by the scenario.</description>
<threshold_list>

# Min_Prncp_Amt
<threshold_list xmlTshldId="300004">
     <tshld_name>Min_Prncp_Amt</tshld_name>
     <min_value>1</min_value>
     <max_value>1000000</max_value>
     <default_value>50000</default_value>
     <value_unit>MONEY</value_unit>
     <current_value>50000</current_value>
     <value_type_code>REAL</value_type_code>
     <display_name>Min Individual Trade Principal</display_name>
     <description>The trade amount for each individual trade to be considered of
                 interest for generating alert.</description>
<threshold_list>
```

Tags

The following sections describe the tags used by the `threshold.xml` file:

<threshold-data>

This is the root tag for the `threshold.xml`.

Attributes - none

Child Tags - <threshold_list>

Occurrences - 1

<threshold_list>

This tag appears multiple times in the .xml file. Each entry corresponds to a distinct threshold definition.

Attributes -

- `xmlTshldId` - It is unique identifier associated with every entry of the tag `<threshold_list>`. Its value is entered as per user discretion; the only condition is that the value should be unique across all the `<threshold_list>` tags in the XML.

Child Tags - `<tshld_name>, <min_value>, <max_value>, <default_value>, <default_value>, <value_unit>, <current_value>, <value_type_code>, <display_name>, <description>`

Occurrences - 1…N

<tshld_name>

This is the child tag for the `<threshold_list>` tag and it contains the threshold name.

Note: Business requires that the threshold name should not contain any spaces.

Attributes - none

Child Tags - none

Occurrences - 1

<min_value>

For numeric thresholds, this tag contains the minimum value for the threshold. For non-numeric thresholds, this tag should be left empty.

Attributes - none

Child Tags - none

Occurrences - 1

<max_value>

For numeric thresholds, this tag contains the maximum value for the threshold. For non-numeric thresholds this tag should be left empty.

Attributes - none

Child Tags - none
Occurrences - 1

<default_value>

This tag contains the default value for the threshold.
Attributes - none
Child Tags - none
Occurrences - 1

<value_type_code>

This tag contains the format of the defined threshold. Currently the allowed values for the type code are:

- INTEGER
- REAL
- STRING
- LIST
- BOOLEAN

Attributes - none
Child Tags - none
Occurrences - 1

<value_unit>

This tag contains the unit of the defined threshold. Currently the allowed values are:

- SECONDS
- MINUTES
- HOURS
- DAYS
- MONEY
- MONEY NO FRAC (Number value without decimals)
- PERCENTAGE

Attributes - none
Child Tags - none
Occurrences - 1

<current_value>

This tag contains the current value for the threshold.
Attributes - none
Child Tags - none
Occurrences - 1

<display_name>

This tag contains the display name for the threshold.
Attributes - none
Child Tags - none
Occurrences - 1

<description>

This tag contains the description/purpose of the threshold.
Attributes - none
Child Tags - none
Occurrences - 1

CHAPTER 5 TA Column Map (TAColumnMap.xml)

This chapter describes the tags used by the `TAColumnMap.xml` file for Threshold Analyzer Utility.

Threshold Analyzer Utility leverages the decisions made by analysts on past alerts to help tune the system going forward. The past alerts are analyzed, and by some mechanism are categorized to identify the quality of the alert.

Use of Threshold Analyzer helps identifying the correlations between alert attributes and alert quality. The results are represented as graph. The `TAColumnMap.xml` defines the highlights which are mentioned in `DataSetRuleMap.xml` in the name of `<axisFlag>` tag. These column highlights can be used by the Threshold Analyzer Utility.

The Scenario Wizard allows you to create user-defined highlights. The system provides a list of eligible highlight names and a `<Create New>` option in the Add Highlight screen for user-defined highlight. The eligible highlight names are configured in the `TAColumnMap.xml` file.

Note: Business requires that for every user-defined highlight, the highlight name should correspond to one of the eligible column names from the TA data table for that particular scenario group.

These eligible highlight names are the eligible column names from the corresponding TA Data table.

The system also allows you to create a new highlight name. To do this, follow the CR process to add the new highlight name as a column to the corresponding TA Data table.

The following table describes the mapping between scenario class group and the TA Data table.

Table 3. Scenario class group and the TA Data

SCNRO_CLASS_GRP	TA Data Table
TC	KDD_TA_TC_DATA
ML	KDD_TA_ML_DATA
BC	KDD_TA_BC_DATA
CR	Currently not available

The following figure illustrates the structure of TA Column Map:

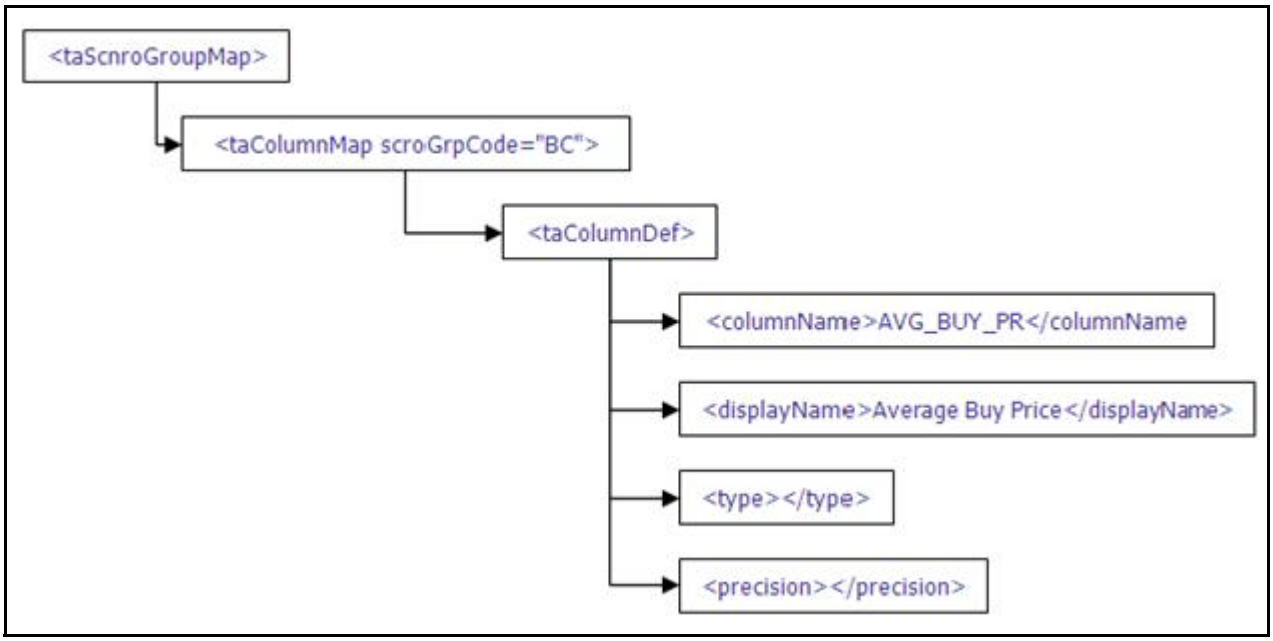

Figure 4. TA Column Map

User-defined Highlight

The Scenario Wizard allows users to add user-defined highlights to a scenario. The system provides a list of eligible highlight names to select from. This list of names is based on the scenario group of the selected scenario class.

Note: Currently, every highlight name in this .XML file corresponds to one of the columns in the TAData table for that scenario group. But there are no constraints while saving a user-defined highlight that validates if the highlight name is one of the columns in the corresponding TA Data table.

Note: Every available scenario group in Scenario Wizard must have a `<taColumnMap>` tag.

```
# BC scenario group
<taColumnMap scroGrpCode="BC">
#
# <taColumnMap> is the parent tag that lists all the eligible highlight names.
#
</taColumnMap>


# BC scenario group
<taColumnMap scroGrpCode="BC">
     # AVG_BUY_PR
     <taColumnDef>
           <columnName>AVG_BUY_PR</columnName>
           <displayName>Average Buy Price</displayName>
           <type></type>
           <precision></precision>
     </taColumnDef>
         # AVG_SELL_PR
     <taColumnDef>
           <columnName>AVG_SELL_PR</columnName>
           <displayName>Average Sell Price</displayName>
           <type></type>
           <precision></precision>
     </taColumnDef>

     # NET_GAIN_LOSS
     <taColumnDef>
           <columnName>NET_GAIN_LOSS</columnName>
           <displayName>Net Gain/Loss</displayName>
           <type></type>
           <precision></precision>
     </taColumnDef>

</taColumnMap>
```

Tags

The following sections describe the tags used by the Scenario Wizard.

<taScnroGroupMap>

This is the root tag for the `TAColumnMap.xml` file.

Attributes - none

Child Tags - `<taColumnMap>`

Occurrences - 1

<taColumnMap>

This tag appears multiple times in the .xml. Each entry corresponds to a distinct scenario class group. This is a parent tag and it contains eligible highlight names for a particular scenario group.

Attributes -

- `scroGrpCode` - This attribute contains the scenario group for a particular scenario class.

Child Tags - `<taColumnDef>`

Occurrences - 1…N

<taColumnDef>

This is the child tag for the `<taColumnMap>` tag. It contains the information for the highlight name. This tag can appear multiple times and each entry corresponds to a highlight name option for user-defined highlight.

Attributes - none

Child Tags - `<columnName>, <displayName>, <type>, <precision>`

Occurrences - 1…N

<columnName>

This tag contains one of the eligible column names from the corresponding TA data table.

Attributes - none

Child Tags - none

Occurrences - 1

<displayName>

This tag contains the display name for the column name.

Attributes - none

Child Tags - none

Occurrences - 1

<type>

Reserved for future implementations.

<precision>

Reserved for future implementations.

CHAPTER 6 Data Set Rule Map (datasetrulemap.xml)

This chapter focuses on Data Set Rule mapping. It explains about the different configurations using `datasetrulemap.xml` file.

The `datasetrulemap.xml` file constitutes the backbone of the Scenario Wizard. It contains the actual logic (that is, the SQL queries) for any scenario. The primary rule and the sub-rule datasets are defined in this XML file. The Scenario Wizard provides flexibility to modify the out-of the-box dataset definition or add new datasets as your business requires.

For a particular scenario class, focus, and associated data combination, this XML file allows the following configurations:

- Primary rule and sub-rule datasets
- Pre-defined highlights
- Pre-defined thresholds
- Test data columns that are not highlights.

The following figure illustrates the structure of `datasetrule map xml`:

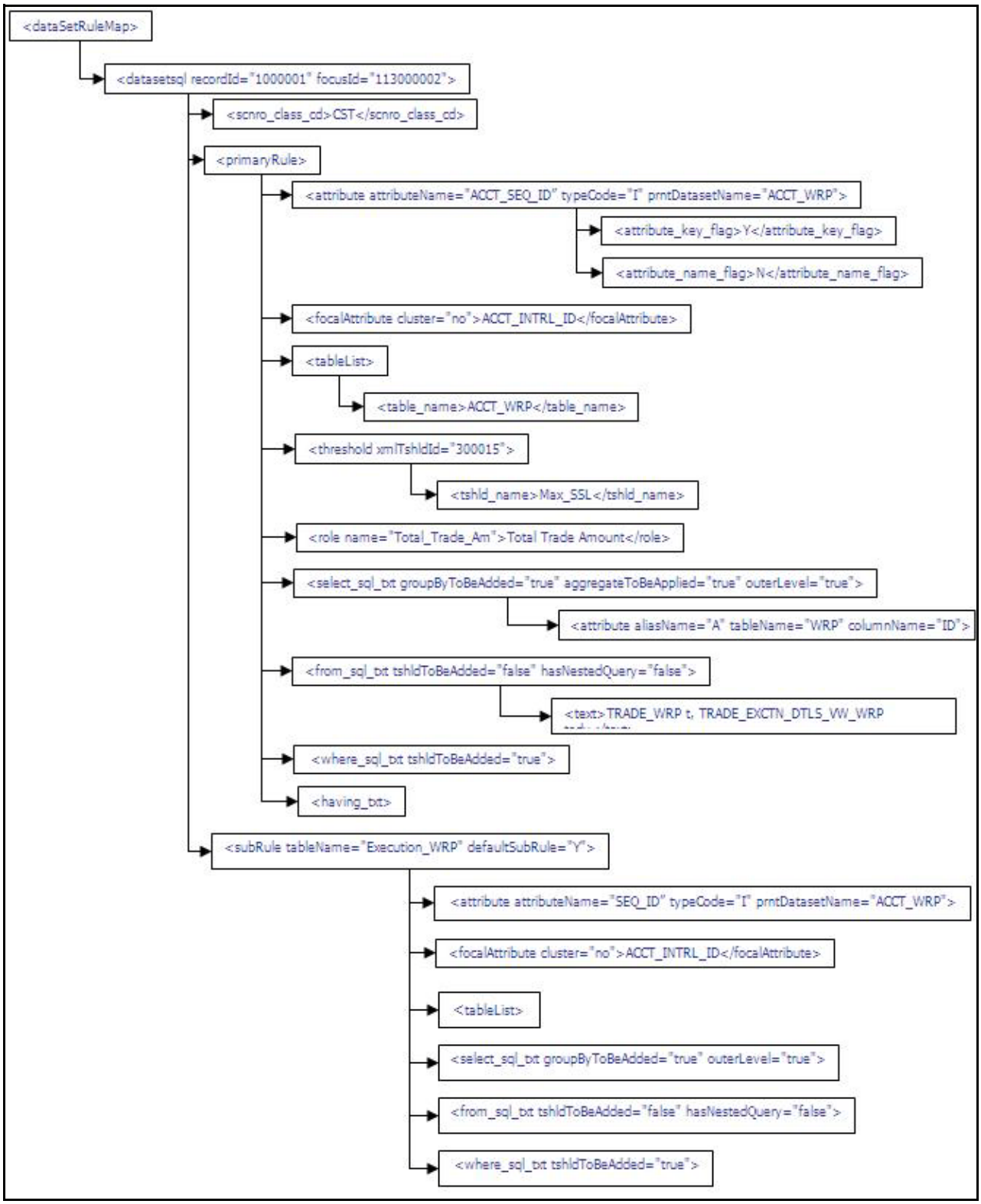

Figure 5. Data Set Rule Map

The datasetrulemap.xml file contains the core business logic and related information. The information is categorized based on a specific combination of scenario class, focus, and associated data. Every unique combination is depicted with a `<datasetsql> tag` and one of the child tags `(<scnro_class_cd>).`

Note: Pre-defined thresholds, pre-defined highlights, and Default columns for test scenario results belong to a primary rule dataset, and are defined as child tags to the `<primaryRule>`tag. This tag also contains the actual SQL query to be executed.

```
# CST_AC_AccountTrades
#
# 1 - focusId contains the CNTRY_ID of account focus from KDD_CENTRICITY table.
# 2 - recordId maps to one of the declared <associatedRecord> in ScenarioFocusMapping.xml
# 3 - Child tag <scnro_class_cd> specifies the SCNRO_CLASS_CD for Customer Risk and Suit-
ability
#      from KDD_SCNRO_CLASS table
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>
     #
     # <datasetsql> tag is the parent tag for primary rule dataset
      and sub rule datasets.
     #
</datasetsql>
```

Primary Rule Dataset

Currently, there can be only one primary rule dataset for a scenario. The primary rule dataset is specified using the `<primaryRule>` tag.

The Primary Rule Dataset contains the alert generation logic for particular scenario. Primary Rule Dataset name is displayed as <Associated Data> in the scenario wizard. The Sub Rule Dataset contains the logic for showing matched record for a particular alert. It is actually building block information for a particular alert.

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
   <scnro_class_cd>CST</scnro_class_cd>

   <primaryRule>
     #name
          <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>

     #description
     <primary_rule_desc>CST_AC_AccountTrades_PR</primary_rule_desc>

           #
     # <primaryRule> tag is the parent tag for all the related entities and the actual SQL
query.
           #
     </primaryRule>

</datasetsql>
```

Dataset Attributes for Primary Rule Dataset

For every dataset attribute, you must specify an <attribute> tag.

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>
     <primaryRule>
           <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
           <primary_rule_desc>CST_AC_AccountTrades_PR</primary_rule_desc>
           # ACCT_SEQ_ID dataset attribute
           <attribute attributeName="ACCT_SEQ_ID" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n"
                 prntDatasetName="ACCT_WRP">
                             # attribute information
                 <attribute_key_flag>Y</attribute_key_flag>
                 <attribute_name_flag>N</attribute_name_flag>
           </attribute>
           # ACCT_INTRL_ID dataset attribute
           <attribute attributeName="ACCT_INTRL_ID" attributePrntId="" attributeId=""
                 typeCode="S" orderFlag="n" discreteFlag="n">
                  # attribute information
                 <attribute_key_flag>N</attribute_key_flag>
                 <attribute_name_flag>N</attribute_name_flag>
           </attribute>
        # SSL_NB dataset attribute
           <attribute attributeName="SSL_NB" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n">
                 # attribute information
                 <attribute_key_flag>N</attribute_key_flag>
                 <attribute_name_flag>N</attribute_name_flag>
           </attribute>
     #
     # <primaryRule> tag contains more entities and the actual SQL query.
     #
     </primaryRule>
</datasetsql>
```

Pre-defined Highlight

Pre-defined highlights can be configured based on the SQL queries for the primary rule dataset. For every highlight, one <role> tag should be specified.

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>

     <primaryRule>
           <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
           <primary_rule_desc>CST_AC_AccountTrades_PR</primary_rule_desc>


           # ACCT_SEQ_ID dataset attribute
           <attribute attributeName="ACCT_SEQ_ID" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n"
                 prntDatasetName="ACCT_WRP">

           # ACCT_INTRL_ID dataset attribute
           <attribute attributeName="ACCT_INTRL_ID" attributePrntId="" attributeId=""
                 typeCode="S" orderFlag="n" discreteFlag="n">

           # SSL_NB dataset attribute
           <attribute attributeName="SSL_NB" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n">

           # Eligible pre-defined highlight Total_Trade_Am
           <role name="Total_Trade_Am" highlight="N" format="REAL5" axisFlag="Y"
                    roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                    roleColumnUsed="last_trade_prnpl_issng_am" taFieldName = "TOT_TRD_AM">

                   # display name
                   Total Trade Amount
           </role>

           # Eligible pre-defined highlight Tot_Trade_Ct
           <role name="Tot_Trade_Ct" highlight="N" format="INTEGER" axisFlag="Y"
                    roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                    roleColumnUsed="trade_intrl_id" taFieldName = "TOT_TRD_CT">

                   # display name
                   Total Trade Ct
           </role>
     #
     # <primaryRule> tag contains more entities and the actual SQL query.
     #
     </primaryRule>

</datasetsql>
```

Pre-defined Threshold

Every threshold that is part of the dataset queries of a scenario must be declared in the `<primaryRule>` tag before it can be used in the query.

In order to include any threshold (that is already defined in the `threshold.xml` file) as a part of any of the dataset queries (primary rule as well as sub-rule), add a <threshold> tag inside the `<primaryRule>` tag.

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>

     <primaryRule>
           <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
           <primary_rule_desc>CST_AC_AccountTrades_PR</primary_rule_desc>

           # ACCT_SEQ_ID dataset attribute
           <attribute attributeName="ACCT_SEQ_ID" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n"
                 prntDatasetName="ACCT_WRP">

           # ACCT_INTRL_ID dataset attribute
           <attribute attributeName="ACCT_INTRL_ID" attributePrntId="" attributeId=""
                 typeCode="S" orderFlag="n" discreteFlag="n">
```

```
          # eligible pre-defined highlight Total_Trade_Am
          <role name="Total_Trade_Am" highlight="N" format="REAL5" axisFlag="Y"
                   roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                  roleColumnUsed="last_trade_prnpl_issng_am" taFieldName = "TOT_TRD_AM">

          # pre-defined threshold Incl_Jurisdictions_Lst
          <threshold xmlTshldId="300000">
                <tshld_name>Incl_Jurisdictions_Lst</tshld_name>
          </threshold>

# pre-defined threshold Min_Prncp_Amt
          <threshold xmlTshldId="300004">
                <tshld_name>Min_Prncp_Amt</tshld_name>
          </threshold>

          # pre-defined threshold Min_Price
          <threshold xmlTshldId="300005">
                <tshld_name>Min_Price</tshld_name>
          </threshold>

     #
     # <primaryRule> tag contains more entities and the actual SQL query.
     #

     </primaryRule>
</datasetsql>
```

Primary Rule Dataset Query

The select query for the primary rule dataset is configured in the xml file using the following tags:

- <select_sql_txt>
- <from_sql_txt>
- <where_sql_txt>
- <group_by_txt>
- <having_txt>

The example of primary rule dataset is shown below.:

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>
     <primaryRule>
           <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
           <primary_rule_desc>CST_AC_AccountTrades_PR</primary_rule_desc>
           # ACCT_SEQ_ID dataset attribute
           <attribute attributeName="ACCT_SEQ_ID" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n"
                 prntDatasetName="ACCT_WRP">
           # ACCT_INTRL_ID dataset attribute
           <attribute attributeName="ACCT_INTRL_ID" attributePrntId="" attributeId=""
                 typeCode="S" orderFlag="n" discreteFlag="n">
# SSL_NB dataset attribute
           <attribute attributeName="SSL_NB" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n">
           # eligible pre-defined highlight Total_Trade_Am
           <role name="Total_Trade_Am" highlight="N" format="REAL5" axisFlag="Y"
                    roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                    roleColumnUsed="last_trade_prnpl_issng_am" taFieldName = "TOT_TRD_AM">
           # eligible pre-defined highlight Tot_Trade_Ct
           <role name="Tot_Trade_Ct" highlight="N" format="INTEGER" axisFlag="Y"
                    roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                    roleColumnUsed="trade_intrl_id" taFieldName = "TOT_TRD_CT">

           # pre-defined threshold Incl_Jurisdictions_Lst
           <threshold xmlTshldId="300000">

           # pre-defined threshold Min_Prncp_Amt
           <threshold xmlTshldId="300004">

           # pre-defined threshold Min_Price
           <threshold xmlTshldId="300005">
#
# primary rule dataset query
#

# primary rule name
<primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
```

```
#focal attribute
<focalAttribute cluster="no">ACCT_INTRL_ID</focalAttribute>

#List of tables which are part of the primary rule dataset query
<tableList>
        <tableName>ACCT_WRP</tableName>
        <tableName>TRADE_EXCTN_DTLS_VW_WRP</tableName>
        <tableName>KDD_CAL_WRP</tableName>
</tableList>

#select clause
<select_sql_txt groupByToBeAdded="true" aggregateToBeApplied="true" outerLevel="false">

     #select clause attributes
     <attribute isAggregate="true" aliasName="ACCT_SEQ_ID" functionName = "max"
                    tableName="ACCT_WRP" columnName="acct_seq_id">
                    t1.acct_seq_id
     </attribute>

     <attribute isAggregate="true" aliasName="SSL_NB" functionName = "max"
                    tableName="ACCT_WRP" columnName="stblty_survl_level_nb">
                    t1.stblty_survl_level_nb
     </attribute>

     <attribute isAggregate="false" aliasName="ACCT_INTRL_ID" tableName="ACCT_WRP"
                    columnName="acct_intrl_id">
                    t1.acct_intrl_id
     </attribute>

     #alias name
     <alias_name>t1</alias_name>

</select_sql_txt>
```

```
#from clause that has a nested query
<from_sql_txt tshldToBeAdded="false" hasNestedQuery="true">

     #select clause of the nested query
     <select_sql_txt>

           <attribute isAggregate="false" tableName="ACCT_WRP"
                         columnName="acct_seq_id">ac.acct_seq_id</attribute>

           <attribute isAggregate="false" tableName="ACCT_WRP"
                         columnName="acct_intrl_id">ac.acct_intrl_id</attribute>

           <attribute isAggregate="false" tableName="TRADE_EXCTN_DTLS_VW_WRP"
                         columnName="trade_intrl_id">tedv.trade_intrl_id</attribute>

     </select_sql_txt>

     #from clause of the nested query; this from clause contains tables, no nested query
<from_sql_txt tshldToBeAdded="true" hasNestedQuery="false">

           #<text> tag contains the table names
           <text>ACCT_WRP ac, TRADE_EXCTN_DTLS_VW_WRP tedv</text>

     </from_sql_txt>
```

```
 #where clause of the nested query
     <where_sql_txt tshldToBeAdded="true">
           <![CDATA[ac.acct_intrl_id = tedv.LAST_CUST_ACCT_INTRL_ID
              --Filtering on Included Jurisdictions threshold
              AND (('ALL' IN (@Incl_Jurisdictions_Lst)) or
              (ac.JRSDCN_CD in (@Incl_Jurisdictions_Lst)))
              --Filtering on Min Individual Trade Principal threshold
              AND tedv.last_trade_prnpl_issng_am > @Min_Prncp_Amt
              --Filtering on Min Price threshold
              AND tedv.last_trade_issng_pr >= @Min_Price]]>
     </where_sql_txt>

     <group_by_txt></group_by_txt>
     <having_txt></having_txt>
</from_sql_txt>

#where clause for the main query
<where_sql_txt tshldToBeAdded="false"></where_sql_txt>

#having clause for the main query
<having_txt>
     <![CDATA[MAX(t1.LAST_DATA_DUMP_DT) > (select add_days(b.CLNDR_DT, -
         @Frequency_Period) from KDD_CAL_WRP b where b.CLNDR_NM = 'SYSCAL'      and
b.CLNDR_DAY_AGE = 0 )]]>
</having_txt>

</primaryRule>

</datasetsql>
```

Tags

<dataSetRuleMap>

This tag is the root tag for the `datasetrulemap.xml` file.

Attributes - none

Child Tags - `<datasetsql>`

Occurrences - 1

<datasetsql>

This tag contains all the datasets. It can appear multiple times in the .xml file. Each entry corresponds to a unique combination of scenario class, focus, and associated data.

Attributes -

- `recordId` - The unique identifier for a particular associated data.
- `focusId` - The focus ID.

Child Tags - `<scnro_class_cd>, <primaryRule>, <subRule>`

Occurrences - 1…N

<scnro_class_cd>

This tag is the child tag for `<datasetsql>` and contains the scenario class code.

Attributes - none

Child Tags - none

Occurrences - 1

<attribute>

This tag defines a dataset attribute which can be used in the select clause of primary rule or sub rule dataset query. This tag can appear multiple times for defining all the dataset attributes.

Attributes -

- `attributeName` - This specifies the name of the attribute.
- `attributePrntId` - This is a reserved attribute for future implementation.
- `attributeId` - This is a reserved attribute for future implementation.
- `typeCode` - In this attribute, user provides the format type of the attribute.
- `orderFlag` - This attribute corresponds to the ORDER_FL column of the KDD_ATTR table. Its value can either be N or Y.
- `discreteFlag` - This attribute corresponds to the DISCRETE_FL column of the KDD_ATTR table. Its value can either be N or Y.
- `prntDatasetName` - This attribute specifies name of the parent dataset for the attribute.

Note: If <attribute_key_flag> tag is Y, then only user should specify some value for this attribute, otherwise it should be empty.

Child Tags - none

Occurrences - 1…N

<attribute_key_flag>

This tag corresponds to the KEY_FL column of the KDD_DATASET_ATTR table. Allowed values for this tag are Y and N.

Attributes - none

Child Tags - none

Occurrences - 1

<attribute_name_flag>

This tag corresponds to the NAME_FL column of the KDD_DATASET_ATTR table. Allowed values for this tag are Y and N.

Attributes - none

Child Tags - none

Occurrences - 1

<focalAttribute>

This tag contains the focal attribute to be used with the dataset query. The algorithm uses this attribute while executing the scenario. For Example Account ID, Customer ID, Order ID, Execution ID has to be used by the dataset query when selecting and aggregating data from tables that are part of the dataset.

Note: This tag is mandatory for both primary rule and sub-rule datasets. For a specific scenario, all datasets should have the same value for the focal attribute. The focal attribute should correspond to one of the attributes defined for the dataset using the <attribute> tag.

Attributes -

- cluster - Currently, the only allowed value for this attribute is *no*.

Child Tags - none

Occurrences - 1

<tableList>

This tag lists all the tables that are used in a dataset query.

Attributes - none

Child Tags - <tableName>

Occurrences - 1

<tableName>

This is the child tag for the <tableList> tag and contains the table name. It can appear multiple times and the number of times it appears equals the number of tables used in the dataset query.

Attributes - none

Child Tags - none

Occurrences - 1…N

<primaryRule>

This tag contains pre-defined thresholds and highlights for the scenario and meta-data related to the primary rule. It also contains the primary rule dataset query.

Attributes - none

- Child Tags - <threshold>, <select_sql_txt>, <from_sql_txt>, <where_sql_txt>, <having_txt>, <attribute>, <prnt_dataset_id>, <primary_rule_name>,<focalAttribute>, <tableList>, <role>, <primary_rule_desc>

Occurrences - 1

<threshold>

Any scenario can have a set of pre-defined thresholds. The definition of every pre-defined threshold used by Scenario Wizard is present in the threshold.xml file.

This tag declares a pre-defined threshold for a particular scenario definition. It can occur multiple times and the number of times it appears equals the number of pre-defined thresholds for that particular scenario.

Attributes -

- xmlTshldId - It is a unique identifier for a threshold. It maps the included threshold to its definition in the threshold.xml file.

Child Tags - <tshld_name>

Occurrences - 1…N

<tshld_name>

This is the child tag for the <threshold> tag. The threshold name value comes from the threshold xml file. If this value is not same, then an error occurs while saving the scenario.

Attributes - none

Child Tags - none

Occurrences - 1

<primary_rule_name>

This tag contains a name for the primary rule dataset.

Attributes - none

Child Tags - none

Occurrences - 1

<primary_rule_desc>

This tag contains a description for the primary rule dataset.

Attributes - none

Child Tags - none

Occurrences - 1

<role>

This tag provides the definition for an eligible pre-defined highlight for the scenario. It can appear multiple times. You can add one or more of these eligible roles as pre-defined highlights for the scenario.

Note: Every highlight (user-defined as well as pre-defined) in Scenario Wizard is, by default, a TA Role. Whenever a highlight is added, one row is entered in the `KDD_PTTRN_ROLE` table and corresponding entry is made in the `KDD_TA_ROLE` table. For this reason, the attribute of the `<role>` tag are mapped to columns of both these tables.

Attributes -

- `name` - This contains the highlight name. Business requires that the highlight name should not contain any spaces. This name should correspond to the name of one of the attributes defined for the dataset using the `<attribute>` tag.
- `highlight` - This attribute indicates whether this role is a highlight or not. The only allowed value is N.
- `format` - This configures the default format for the highlight, such as REAL5, STRING, and so on. You can change the default format in the Add/Edit Highlight pop-up.
- `axisFlag` - Indicates whether the corresponding TA Role can be used on X/Y axis (of threshold analyzer). Allowed values are Y and N.
- `roundDigitCount` - This sets the precision for the data.
- `roleTableUsed` - This indicates the table name of the highlight attribute.
- `roleColumnUsed` - This contains the column name.
- `taFieldName` - This attribute has the name of the column from the corresponding TA Data table to which this highlight maps.

Child Tags - none

Occurrences - 1…N

<defaultTestColName>

This tag lists the default test result columns for a scenario. There can be zero or more such columns.

Note: While testing a scenario, results are displayed under the highlights defined. Apart from highlights, you can configure some columns from primary rule dataset tables for which data is always displayed on test screen.

Test Scenario - Default Test Result Columns

Default test result columns are given as child tag for the <primaryRule> tag. For every test result column required, add a <defaultTestColName> tag.

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>

     <primaryRule>
           <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
           <primary_rule_desc>CST_AC_AccountTrades_PR</primary_rule_desc>

           # ACCT_SEQ_ID dataset attribute
           <attribute attributeName="ACCT_SEQ_ID" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n"
                 prntDatasetName="ACCT_WRP">

           # ACCT_INTRL_ID dataset attribute
           <attribute attributeName="ACCT_INTRL_ID" attributePrntId="" attributeId=""
                 typeCode="S" orderFlag="n" discreteFlag="n">

           # SSL_NB dataset attribute
           <attribute attributeName="SSL_NB" attributePrntId="" attributeId=""
                 typeCode="I" orderFlag="n" discreteFlag="n">

           # eligible pre-defined highlight Total_Trade_Am
           <role name="Total_Trade_Am" highlight="N" format="REAL5" axisFlag="Y"
                    roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                   roleColumnUsed="last_trade_prnpl_issng_am" taFieldName = "TOT_TRD_AM">
```

```
# eligible pre-defined highlight Tot_Trade_Ct
          <role name="Tot_Trade_Ct" highlight="N" format="INTEGER" axisFlag="Y"
                   roundDigitCount="2" roleTableUsed="TRADE_EXCTN_DTLS_VW_WRP"
                   roleColumnUsed="trade_intrl_id" taFieldName = "TOT_TRD_CT">

          # pre-defined threshold Incl_Jurisdictions_Lst
          <threshold xmlTshldId="300000">
          # pre-defined threshold Min_Prncp_Amt
          <threshold xmlTshldId="300004">
          # pre-defined threshold Min_Price
          <threshold xmlTshldId="300005">
          # primary rule name
          <primary_rule_name>CST_AC_AccountTrades_PR</primary_rule_name>
          #focal attribute
          <focalAttribute cluster="no">ACCT_INTRL_ID</focalAttribute>
          #List of tables which are part of the primary rule dataset query
<tableList>
                          #select clause
          <select_sql_txt groupByToBeAdded="true" aggregateToBeApplied="true"
                                    outerLevel="false">
          #from clause for the main query
          <from_sql_txt tshldToBeAdded="false" hasNestedQuery="true">
          #where clause for the main query
          <where_sql_txt tshldToBeAdded="false">
           #having clause for the main query
          <having_txt>
     #specifying ACCT_SEQ_ID is specified as a default test column name
          <defaultTestColName name="ACCT_SEQ_ID"
                                          displayName="Account Sequence ID">
          </defaultTestColName>
          #specifying ACCT_INTRL_ID is specified as a default test column name
          <defaultTestColName name="ACCT_INTRL_ID"
                                          displayName="Account Internal ID">
          </defaultTestColName>
     </primaryRule>
</datasetsql>
```

Attributes -

- `name` - This attribute has the column name from primary rule dataset tables.
- `displayName` - This attribute contains the display name for the test column.

Child Tags - none

Occurrences - 0…N

Sub-Rule Dataset

For every scenario there can be one or more sub-rule datasets. For every sub-rule dataset, specify a `<subRule>` tag.

Note: Both the `<primaryRule>` and `<subRule>` tags follow the same convention for configuring the queries. A `<subRule>` tag primarily differs from `<primaryRule>` in that it does not contain tags for pre-defined highlight, pre-defined threshold and default test columns.

```
#
# CST_AC_AccountTrades
#
<datasetsql recordId="1000001" focusId="113000002">
     <scnro_class_cd>CST</scnro_class_cd>

     # primary rule query
     <primaryRule>
     # sub rule dataset query
     <subRule tableName="TRADE_WRP" defaultSubRule="Y">
           # name and description
           <sub_rule_name>CST_AC_AccountTrades_Trades_BB_SR</sub_rule_name>
           <sub_rule_desc>CST_AC_AccountTrades_Trades_BB_SR</sub_rule_desc>
               # dataset attributes
           <attribute attributeName="TRADE_SEQ_ID" attributePrntId="" attributeId=""
                            typeCode="I" orderFlag="n" discreteFlag="n"
                            prntDatasetName="TRADE_WRP">
               <attribute attributeName="ACCT_INTRL_ID" attributePrntId="" attributeId=""
                            typeCode="S" orderFlag="n" discreteFlag="n">
                       # focal attributes
           <focalAttribute cluster="no">ACCT_INTRL_ID</focalAttribute>
                    # table list
           <tableList>
  # select, from, where and having clause
           <select_sql_txt groupByToBeAdded="false"
                                  aggregateToBeApplied="true" outerLevel="true">

           <from_sql_txt tshldToBeAdded="true" hasNestedQuery="false">
                <where_sql_txt tshldToBeAdded="true">
                 <having_txt></having_txt>
     </subRule>
</datasetsql>
```

APPENDIX A Embedding Queries in XML

The below example shows how to edit XML with simple SQL query.

SQL Query

```
SELECT e.EXCTN_SEQ_ID, e.last_cust_acct_intrl_id
FROM EXECUTION_WRP e, TRADE_EXCTN_DTLS_VW_WRP tedv
WHERE e.exctn_intrl_id = tedv.TRADE_INTRL_ID
          AND (('ALL' IN (@Product_Type_List)) or (tedv.PRDCT_TYPE_CD in
(@Product_Type_List)))
          AND (('ALL' IN (@Incl_Prod_Sub_Tys_Lst)) or
                     (tedv.PRDCT_SUB_TYPE_CD IN (@Incl_Prod_Sub_Tys_Lst)))
          AND tedv.TRADE_PURP_CD in (@Incl_Trade_Purp_Codes_Lst)
          AND tedv.LAST_TRADE_PRNPL_BASE_AM > @Min_Prncp_Amt
          AND tedv.LAST_TRADE_BASE_PR >= @Min_Price
          AND tedv.LAST_TRADE_BASE_PR <= @Max_Price
          AND tedv.TRDR_BUY_SELL_CD = @trd_dir
          AND tedv.TRADE_DT > ( select add_days(kl.CLNDR_DT, - @Look_Back_Period)
                                                                    from KDD_CAL_WRP kl
where kl.CLNDR_NM = 'SYSCAL'
and kl.CLNDR_DAY_AGE = 0)
AND tedv.TRADE_DT <= ( select kl.CLNDR_DT
from KDD_CAL_WRP kl
where kl.CLNDR_NM = 'SYSCAL'
 and kl.CLNDR_DAY_AGE = 0)
```

XML Equivalent

```
<select_sql_txt groupByToBeAdded="false" aggregateToBeApplied="true"
                outerLevel="true">
  <attribute aliasName="EXCTN_SEQ_ID" tableName="EXECUTION_WRP"
             columnName="EXCTN_SEQ_ID">e.EXCTN_SEQ_ID</attribute>
  <attribute aliasName="ACCT_INTRL_ID" tableName="TRADE_EXCTN_DTLS_VW_WRP"
             columnName="last_cust_acct_intrl_id">e.last_cust_acct_intrl_id</attribute>
</select_sql_txt>
<from_sql_txt tshldToBeAdded="true" hasNestedQuery="false">
      <text>EXECUTION_WRP e, TRADE_EXCTN_DTLS_VW_WRP tedv</text>
</from_sql_txt>
<where_sql_txt tshldToBeAdded="true">
  <![CDATA[
       e.exctn_intrl_id = tedv.TRADE_INTRL_ID
       AND (('ALL' IN (@Product_Type_List)) or
       (tedv.PRDCT_TYPE_CD in (@Product_Type_List)))
       AND (('ALL' IN (@Incl_Prod_Sub_Tys_Lst)) or
           (tedv.PRDCT_SUB_TYPE_CD IN (@Incl_Prod_Sub_Tys_Lst)))
       AND tedv.TRADE_PURP_CD in (@Incl_Trade_Purp_Codes_Lst)
       AND tedv.LAST_TRADE_PRNPL_BASE_AM > @Min_Prncp_Amt
       AND tedv.LAST_TRADE_BASE_PR >= @Min_Price
       AND tedv.LAST_TRADE_BASE_PR <= @Max_Price
       AND tedv.TRDR_BUY_SELL_CD = @trd_dir
       AND tedv.TRADE_DT >  (select add_days(kl.CLNDR_DT, - @Look_Back_Period)
from KDD_CAL_WRP kl
where kl.CLNDR_NM = 'SYSCAL'
and kl.CLNDR_DAY_AGE = 0)
AND tedv.TRADE_DT <= (select kl.CLNDR_DT
from KDD_CAL_WRP kl
where kl.CLNDR_NM = 'SYSCAL'
and kl.CLNDR_DAY_AGE = 0)]]>
</where_sql_txt>
<having_txt></having_txt>
```

APPENDIX B Using Functions in Query

The below example shows the usage of aggregate functions inside SQL query.

SQL Query

```
SELECT ac.acct_seq_id ACCT_SEQ_ID ,
            ac.acct_intrl_id ACCT_INTRL_ID ,
            s.scrty_intrl_id Security_ID ,
            s.scrty_shrt_nm Security_Short_Name ,
            sum(ap.unit_qt) Unit_Quantity ,
            max(ac.stblty_survl_level_nb) SSL_NB ,
            s.prod_type_cd Product_Type ,
            max(
                    CASE WHEN coalesce(abps.net_wrth_am, 1) > 0
                    THEN ((ap.mktval_am / coalesce(abps.net_wrth_am, 1)) * 100)
                    ELSE 0 END ) Net_Worth_Percentage
FROM ACCT_WRP ac,
         ACCT_POSN_WRP ap,
         SCRTY_WRP s,
         ACCT_BAL_POSN_SMRY_WRP abps
WHERE s.scrty_intrl_id = ap.scrty_intrl_id
           AND ac.acct_intrl_id = ap.acct_intrl_id
           AND ac.acct_intrl_id = abps.acct_intrl_id
           --Filtering on Included Jurisdictions threshold
           AND (('ALL' IN (@Incl_Jurisdictions_Lst)) or (ac.JRSDCN_CD in
(@Incl_Jurisdictions_Lst)))
           --Filtering on Incluided Mantas Business Account Types
           AND ac.MANTAS_ACCT_BUS_TYPE_CD in (@Mantas_Bus_Acct_Type_Lst)
           -- Filtering on Minimum SSL
           AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) >= @Min_SSL
           -- Filtering on maximum SSL
           AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) <= @Max_SSL
           -- Filtering on Included Product Types threshold
           AND (('ALL' IN (@Product_Type_List)) or (s.prod_type_cd in
(@Product_Type_List)))
           -- Filtering on Included Product Subtypes threshold
           AND (('ALL' IN (@Incl_Prod_Sub_Tys_Lst))
                     or (s.prod_sub_type_cd IN (@Incl_Prod_Sub_Tys_Lst)))
           -- FIltering on Min Position Unit Quantity
           AND ap.unit_qt >= @Min_posn_Unit_qt
           -- Filtering on Min Position Value
           AND ap.mktval_am >= @Min_Posn_Value
           --Filtering on Min Position as Pct Net Worth
           AND (CASE WHEN coalesce(abps.net_wrth_am, 1) > 0
```

```
       THEN((ap.mktval_am / coalesce(abps.net_wrth_am, 1)) * 100)
                   ELSE 0 END) >= @Min_Posn_As_Net_Wrth_Pctg
              --Filtering on Included Manners Held
              AND (('ALL' IN (@Incl_Manners_Type)) or (ap.acctg_rule_cd in
(@Incl_Manners_Type)))
GROUP BY ac.acct_seq_id ,
                  ac.acct_intrl_id ,
                  s.scrty_intrl_id ,
                  s.scrty_shrt_nm ,                s.prod_type_cd
```

XML Equivalent

```
    <select_sql_txt groupByToBeAdded="true" aggregateToBeApplied="true" outerLevel="true">
    <attribute isAggregate="false" aliasName="ACCT_SEQ_ID" tableName="ACCT_WRP"
                columnName="acct_seq_id">ac.acct_seq_id</attribute>
    <attribute isAggregate="false" aliasName="ACCT_INTRL_ID" tableName="ACCT_WRP"
                columnName="acct_intrl_id">ac.acct_intrl_id</attribute>
    <attribute isAggregate="false" aliasName="Security_ID" tableName="SCRTY_WRP"
                columnName="scrty_intrl_id">s.scrty_intrl_id</attribute>
    <attribute isAggregate="false" aliasName="Security_Short_Name" tableName="SCRTY_WRP"
                columnName="scrty_shrt_nm">s.scrty_shrt_nm</attribute>
    <attribute isAggregate="true" aliasName="Unit_Quantity" functionName = "sum"
               tableName="ACCT_POSN_WRP" columnName="unit_qt">ap.unit_qt</attribute>
    <attribute isAggregate="true" aliasName="SSL_NB" functionName = "max"
               tableName="ACCT_WRP" ZolumnName="stblty_survl_level_nb">
               ac.stblty_survl_level_nb</attribute>
    <attribute isAggregate="false" aliasName="Product_Type" tableName="SCRTY_WRP"
               columnName="prod_type_cd">s.prod_type_cd</attribute>
    <attribute isAggregate="true" aliasName="Net_Worth_Percentage" functionName="max"
                tableName="ACCT_BAL_POSN_SMRY_WRP, ACCT_POSN_WRP"
                columnName="net_wrth_am, mktval_am">
                <![CDATA[CASE WHEN coalesce(abps.net_wrth_am, 1) > 0
                              THEN ((ap.mktval_am / coalesce(abps.net_wrth_am, 1)) *
100)
                              ELSE 0 END ]]>
</attribute>
</select_sql_txt>
<from_sql_txt tshldToBeAdded="true" hasNestedQuery="false">
    <text>ACCT_WRP ac, ACCT_POSN_WRP ap, SCRTY_WRP s, ACCT_BAL_POSN_SMRY_WRP abps
    </text>
</from_sql_txt>
<where_sql_txt tshldToBeAdded="true">
         <![CDATA[s.scrty_intrl_id = ap.scrty_intrl_id
              AND ac.acct_intrl_id = ap.acct_intrl_id
              AND ac.acct_intrl_id = abps.acct_intrl_id
              --Filtering on Included Jurisdictions threshold
```

```
                  AND (('ALL' IN (@Incl_Jurisdictions_Lst)) or (ac.JRSDCN_CD in
(@Incl_Jurisdictions_Lst)))
                -Filtering on Incluided Mantas Business Account Types
                AND ac.MANTAS_ACCT_BUS_TYPE_CD in (@Mantas_Bus_Acct_Type_Lst)
                 -- Filtering on Minimum SSL
                AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) >= @Min_SSL
               -- Filtering on maximum SSL
               AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) <= @Max_SSL
               -- Filtering on Included Product Types threshold
               AND (('ALL' IN (@Product_Type_List)) or  (s.prod_type_cd in
(@Product_Type_List)))
               -- Filtering on Included Product Subtypes threshold
               AND (('ALL' IN (@Incl_Prod_Sub_Tys_Lst)) or (s.prod_sub_type_cd IN
(@Incl_Prod_Sub_Tys_Lst)))
              -- FIltering on Min Position Unit Quantity
              AND ap.unit_qt >= @Min_posn_Unit_qt
              -- Filtering on Min Position Value
              AND ap.mktval_am >= @Min_Posn_Value
              --Filtering on Min Position as Pct Net Worth
              AND (CASE WHEN coalesce(abps.net_wrth_am, 1) > 0
              THEN((ap.mktval_am / coalesce(abps.net_wrth_am, 1)) * 100) ELSE 0 END)
>=@Min_Posn_As_Net_Wrth_Pctg
              --Filtering on Included Manners Held
              AND (('ALL' IN (@Incl_Manners_Type)) or (ap.acctg_rule_cd in
@Incl_Manners_Type)))
]]>
</where_sql_txt>
<having_txt></having_txt>
```

APPENDIX C Nested Query

The below example shows the usage of nested query inside the SQL.

SQL Query

```
SELECT max(t1.acct_seq_id) ACCT_SEQ_ID ,
            sum(t1.last_trade_prnpl_issng_am) Total_Trade_Am ,
            count(t1.trade_intrl_id) Tot_Trade_Ct ,
            max(t1.stblty_survl_level_nb) SSL_NB ,
            t1.acct_intrl_id ACCT_INTRL_ID ,
            MIN(t1.ABS_PRICE_CHG) ABS_PRICE_CHG
FROM (
            SELECT ac.acct_seq_id ,
                        ac.acct_intrl_id ,
                        tedv.trade_intrl_id ,
                        tedv.last_trade_prnpl_issng_am ,
                        ac.stblty_survl_level_nb ,
                        tedv.LAST_DATA_DUMP_DT ,
                        tedv.BUS_DMN_LIST_TX ABS_PRICE_CHG
            FROM ACCT_WRP ac, TRADE_EXCTN_DTLS_VW_WRP tedv
            WHERE ac.acct_intrl_id = tedv.LAST_CUST_ACCT_INTRL_ID
                        -- Filtering on Included Business Account Types threshold
                       AND ac.MANTAS_ACCT_BUS_TYPE_CD in (@Mantas_Bus_Acct_Type_Lst)
                       -- Filtering on Included Product Types threshold
                       AND (('ALL' IN (@Product_Type_List))
                                   or (Tedv.PRDCT_TYPE_CD in (@Product_Type_List)))
   --Filtering on Included Jurisdictions threshold
                       AND (('ALL' IN (@Incl_Jurisdictions_Lst))
                                   or (ac.JRSDCN_CD in (@Incl_Jurisdictions_Lst)))
                       -- Filtering on Included Product Subtypes threshold
                       AND (('ALL' IN (@Incl_Prod_Sub_Tys_Lst))
                                   or (tedv.PRDCT_SUB_TYPE_CD IN
(@Incl_Prod_Sub_Tys_Lst)))
                       --Filtering on Included Trade Purpose Types threshold
                       AND tedv.TRADE_PURP_CD in (@Incl_Trade_Purp_Codes_Lst)
                       --Filtering on Min Individual Trade Principal threshold
                       AND tedv.last_trade_prnpl_issng_am > @Min_Prncp_Amt
                       -- Filtering on Min Price threshold
                       AND tedv.last_trade_issng_pr >= @Min_Price
```

```
                         -- Filtering on Max Price threshold
                        AND tedv.last_trade_issng_pr <= @Max_Price
                        -- Filtering on Trade Direction (Buy==1/Sell==2) threshold
                        AND tedv.TRDR_BUY_SELL_CD = @trd_dir
                        --Filtering on Lookback period threshold
                        AND tedv.TRADE_DT >   (select add_days(kl.CLNDR_DT, -
@Look_Back_Period)
                                                              from KDD_CAL_WRP kl
                                                              where kl.CLNDR_NM = 'SYSCAL'
                                                              and kl.CLNDR_DAY_AGE = 0)
                        AND tedv.TRADE_DT <= (select kl.CLNDR_DT
                                                               from KDD_CAL_WRP kl
                                                               where kl.CLNDR_NM = 'SYSCAL'
                                                               and kl.CLNDR_DAY_AGE = 0)
                        --Filtering on Suitability Level threshold
                        AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) >= @Min_SSL
                        AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) <= @Max_SSL
) t1
GROUP BY t1.acct_intrl_id
HAVING MAX(t1.LAST_DATA_DUMP_DT) > (select add_days(b.CLNDR_DT, - @Frequency_Period)
                                                                 from KDD_CAL_WRP b
                                                                where b.CLNDR_NM = 'SYSCAL'
                                                                 and b.CLNDR_DAY_AGE = 0 )
   AND max(t1.LAST_DATA_DUMP_DT) <= (select KC.CLNDR_DT
                                                                 from KDD_CAL_WRP KC
                                                                where KC.CLNDR_NM = 'SYSCAL'
                                            and KC.CLNDR_DAY_AGE = 0)

  --Filtering on Min Total Trade Principal threshold

   AND sum(t1.last_trade_prnpl_issng_am) > @Min_Tot_Prncp_Amt
   --Filtering on Min Trade Ct threshold
   AND count(t1.trade_intrl_id) >= @Min_Trd_Ct
```

XML Equivalent

```
<select_sql_txt groupByToBeAdded="true" aggregateToBeApplied="true" outerLevel="false">
     <attribute isAggregate="true" aliasName="ACCT_SEQ_ID" functionName = "max"
                    tableName="ACCT_WRP"
columnName="acct_seq_id">t1.acct_seq_id</attribute>
     <attribute isAggregate="true" aliasName="Total_Trade_Am" functionName = "sum"
                    tableName="TRADE_EXCTN_DTLS_VW_WRP"
                 columnName="last_trade_prnpl_issng_am">t1.last_trade_prnpl_issng_am

     </attribute>
     <attribute isAggregate="true" aliasName="Tot_Trade_Ct" functionName = "count"
                    tableName="TRADE_EXCTN_DTLS_VW_WRP"
                    columnName="trade_intrl_id">t1.trade_intrl_id
     </attribute>
     <attribute isAggregate="true" aliasName="SSL_NB" functionName = "max"
                    tableName="ACCT_WRP" columnName="stblty_survl_level_nb">
                    t1.stblty_survl_level_nb
     </attribute>
     <attribute isAggregate="false" aliasName="ACCT_INTRL_ID" tableName="ACCT_WRP"
                    columnName="acct_intrl_id">t1.acct_intrl_id
     </attribute>

     <alias_name>t1</alias_name>
</select_sql_txt>
```

```
<from_sql_txt tshldToBeAdded="false" hasNestedQuery="true">
     <select_sql_txt>
           <attribute isAggregate="false" tableName="ACCT_WRP"
                          columnName="acct_seq_id">ac.acct_seq_id
           </attribute>
           <attribute isAggregate="false" tableName="ACCT_WRP"
                          columnName="acct_intrl_id">ac.acct_intrl_id
           </attribute>
           <attribute isAggregate="false" tableName="TRADE_EXCTN_DTLS_VW_WRP"
                          columnName="trade_intrl_id">tedv.trade_intrl_id
           </attribute>
           <attribute isAggregate="false" tableName="TRADE_EXCTN_DTLS_VW_WRP"

columnName="last_trade_prnpl_issng_am">tedv.last_trade_prnpl_issng_am
           </attribute>
           <attribute isAggregate="false" tableName="ACCT_WRP"
                          columnName="stblty_survl_level_nb">ac.stblty_survl_level_nb
           </attribute>
           <attribute isAggregate="false" tableName="TRADE_EXCTN_DTLS_VW_WRP"
                          columnName="LAST_DATA_DUMP_DT">tedv.LAST_DATA_DUMP_DT
           </attribute>
     </select_sql_txt>
```

```
<from_sql_txt tshldToBeAdded="true" hasNestedQuery="false">
          <text>ACCT_WRP ac, TRADE_EXCTN_DTLS_VW_WRP tedv</text>
    </from_sql_txt>

    <where_sql_txt tshldToBeAdded="true"><![CDATA[
      ac.acct_intrl_id = tedv.LAST_CUST_ACCT_INTRL_ID
          -- Filtering on Included Business Account Types threshold
          AND ac.MANTAS_ACCT_BUS_TYPE_CD in (@Mantas_Bus_Acct_Type_Lst)
          -- Filtering on Included Product Types threshold
          AND (('ALL' IN (@Product_Type_List)) or
              (Tedv.PRDCT_TYPE_CD in (@Product_Type_List)))
          --Filtering on Included Jurisdictions threshold
          AND (('ALL' IN (@Incl_Jurisdictions_Lst)) or
              (ac.JRSDCN_CD in (@Incl_Jurisdictions_Lst)))
          -- Filtering on Included Product Subtypes threshold
          AND (('ALL' IN (@Incl_Prod_Sub_Tys_Lst)) or
              (tedv.PRDCT_SUB_TYPE_CD IN (@Incl_Prod_Sub_Tys_Lst)))
          --Filtering on Included Trade Purpose Types threshold
          AND tedv.TRADE_PURP_CD in (@Incl_Trade_Purp_Codes_Lst)
          --Filtering on Min Individual Trade Principal threshold
          AND tedv.last_trade_prnpl_issng_am > @Min_Prncp_Amt
          -- Filtering on Min Price threshold
          AND tedv.last_trade_issng_pr >= @Min_Price
          -- Filtering on Max Price threshold
          AND tedv.last_trade_issng_pr <= @Max_Price
          -- Filtering on Trade Direction (Buy==1/Sell==2) threshold
          AND tedv.TRDR_BUY_SELL_CD = @trd_dir
          --Filtering on Lookback period threshold
```

```
        AND tedv.TRADE_DT >
               (select add_days(kl.CLNDR_DT, - @Look_Back_Period)
                  from KDD_CAL_WRP kl
                 where kl.CLNDR_NM = 'SYSCAL'
                   and kl.CLNDR_DAY_AGE = 0)
           AND tedv.TRADE_DT <=
               (select kl.CLNDR_DT
                  from KDD_CAL_WRP kl
                 where kl.CLNDR_NM = 'SYSCAL'
                   and kl.CLNDR_DAY_AGE = 0)
           --Filtering on Suitability Level threshold
           AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) >= @Min_SSL
           AND coalesce(ac.STBLTY_SURVL_LEVEL_NB, 99) <= @Max_SSL
       ]]>
     </where_sql_txt>
     <group_by_txt></group_by_txt>
     <having_txt></having_txt>
</from_sql_txt>

<where_sql_txt tshldToBeAdded="false"></where_sql_txt>
<having_txt><![CDATA[
   MAX(t1.LAST_DATA_DUMP_DT) > (select add_days(b.CLNDR_DT, - @Frequency_Period)
                                                    from KDD_CAL_WRP b
                                                    where b.CLNDR_NM = 'SYSCAL'
                                                    and b.CLNDR_DAY_AGE = 0 )
   AND max(t1.LAST_DATA_DUMP_DT) <= (select KC.CLNDR_DT
                                                                from KDD_CAL_WRP KC
                                                               where KC.CLNDR_NM = 'SYSCAL'
                                          AND KC.CLNDR_DAY_AGE = 0)
```

```
   --Filtering on Min Total Trade Principal threshold

   AND sum(t1.last_trade_prnpl_issng_am) > @Min_Tot_Prncp_Amt
   --Filtering on Min Trade Ct threshold
   AND count(t1.trade_intrl_id) >= @Min_Trd_Ct]]>
  </having_txt>
```