Retek® Predictive Application Server™ 11.0.4

Solution Extension Configuration Guide

Retek®

Retek Predictive Application Server

The software described in this documentation is furnished under a license agreement, is the confidential information of Retek Inc., and may be used only in accordance with the terms of the agreement.

No part of this documentation may be reproduced or transmitted in any form or by any means without the express written permission of Retek Inc., Retek on the Mall, 950 Nicollet Mall, Minneapolis, MN 55403, and the copyright notice may not be removed without the consent of Retek Inc.

Information in this documentation is subject to change without notice.

Retek provides product documentation in a read-only-format to ensure content integrity. Retek Customer Support cannot support documentation that has been changed without Retek authorization.

Retek® Predictive Application Server™ is a trademark of Retek Inc.

Retek and the Retek logo are registered trademarks of Retek Inc.

This unpublished work is protected by confidentiality agreement, and by trade secret, copyright, and other laws. In the event of publication, the following notice shall apply:

©2004 Retek Inc. All rights reserved.

All other product names mentioned are trademarks or registered trademarks of their respective owners and should be treated as such.

Printed in the United States of America.

Corporate Headquarters:

Retek Inc.
Retek on the Mall
950 Nicollet Mall
Minneapolis, MN 55403

888.61.RETEK (toll free US)
+1 612 587 5000

Fax: +1 612.587.5100

European Headquarters:

Retek
110 Wigmore Street
London
W1U 3RW
United Kingdom

Switchboard:
+44 (0)20 7563 4600

Sales Enquiries:
+44 (0)20 7563 46 46

Fax: +44 (0)20 7563 46 10

Customer Support

Customer Support hours

Customer Support is available 7x24x365 via e-mail, phone, and Web access.

Depending on the Support option chosen by a particular client (Standard, Plus, or Premium), the times that certain services are delivered may be restricted. Severity 1 (Critical) issues are addressed on a 7x24 basis and receive continuous attention until resolved, for all clients on active maintenance. Retek customers on active maintenance agreements may contact a global Customer Support representative in accordance with contract terms in one of the following ways.

Contact Method	**Contact Information**
E-mail	support@retek.com
Internet (ROCS)	rocs.retek.com Retek's secure client Web site to update and view issues
Phone	1 612 587 5800

Toll free alternatives are also available in various regions of the world:

Australia	1 800 555 923 (AU-Telstra) or 1 800 000 562 (AU-Optus)
France	0800 90 91 66
United Kingdom	0800 917 2863
United States	1 800 61 RETEK or 800 617 3835

Mail	Retek Customer Support Retek on the Mall 950 Nicollet Mall Minneapolis, MN 55403

When contacting Customer Support, please provide:

- Product version and program/module name.
- Functional and technical description of the problem (include business impact).
- Detailed step by step instructions to recreate.
- Exact error message received.
- Screen shots of each step you take.

Contents

Chapter 1 – Introduction.. 1

Overview 1

Chapter 2 – Configuring the Curve solution extension 3

Overview 3

Creating a Curve solution extension 4

Renaming an solution extension 5

Curve hierarchy configuration requirements...................................... 5

Configuring Curve differentiator dimensions (optional) and merchandise hierarchy requirements to support RMS... 6

Creating a final profile 8

Creating a source profile 8

Deleting a profile level... 8

Editing Curve profile properties... 9

Autogenerating hierarchies, measures, and rules 14

Configuration restrictions on autogenerated hierarchies, measures, rules, workbooks, and wizards... 15

Curve example... 16

Chapter 3 – Configuring the Retek Demand Forecasting solution extension 17

Overview 17

Creating a RDF solution extension 18

Renaming an solution extension 18

Forecasting Merchandise and Location Hierarchy Requirements 18

Creating a final forecast level.. 19

Creating a source forecast level 19

Deleting a forecast level 19

Editing RDF level properties 20

Autogenerating hierarchies, measures, and rules 25

Configuration restrictions on autogenerated hierarchies, measures, rules, workbooks, and wizards 25

Common Solution Measures Required 26

RDF example 28

Chapter 4 – Configuring the Promote solution extension. 29

Overview 29

Creating a Promote solution extension 29

Renaming an solution extension 29

Promotion/Causal Forecast Levels 30

Creating a promotion 30

Deleting a promotion 30

Editing promotion properties 30

Autogenerating hierarchies, measures, and rules 31

Configuration restrictions on autogenerated hierarchies, measures, rules, workbooks, and wizards 32

Promotion example 33

Chapter 5 – Configuring the Forecast procedure 35

Overview 35

Syntax Legend 36

Syntax 37

Examples 38

Forecast Parameter Table 38

Forecast Method/Model List 43
Parameter/Model Dependencies 43
Forecast Procedure Configuration Notes 44
Forecast Method Configuration Notes 45
Chapter 6 – Configuring the LostSale Function 47
Overview 47
Syntax Legend 47
Syntax 48
Parameter Table Legend 48
LostSale Parameter Table 49
LostSale Parameter Description 51
LostSale Filtering Method List 52
LostSale Filtering Method Descriptions 54
Standard Median 54
Retek Median 55
Standard Exponential Smoothing 57
Lost Sales – Standard Exponential Smoothing 59
Forecast Sigma 60
Forecast Sigma Event 62
Override 64
Increment 66
Clear 68
LostSale Function Configuration Notes 69
Configuration Restriction 70

Chapter 7 – Configuring the Markdown Optimizer Function 71

Syntax Legend 71

Syntax 72

Example Usage 72

Configuration Parameters and Rules 73

Input Parameters 73

Input Measures to support optimization rules 77

Calc Rules to support the configuration of input parameters: 78

Output Parameters 79

Configuration Restriction 80

Chapter 1 – Introduction

Overview

A solution extension is functionality that is included in RPAS, but can be leveraged to configure and build a solution with specific functionality (for example, Promote which supports causal/promotional forecasting). The solution extensions that have been developed within RPAS 11.0.4 are as follows:

- Retek Demand Forecasting (RDF)
- Retek Promote
- Retek Curve
- Forecasting
- LostSale (formerly Preprocessing)
- Retek Markdown Optimization

The configuration tools allow you to create special solutions for these extensions. For some solution extensions (RDF, Promote, Curve) the configuration tools provide a user interface to define levels, parameters and other special properties unique to the extension. Based on the values set by the user, an autogeneration process will create measures and rules that are required to support essential solution extension functionality. A separate process is outlined in the RDF Installation Guide to generate the RDF, Promote and Curve Workbook Templates.

Other solutions extensions such as Forecasting, LostSale and Retek Markdown Optimization, are developed as functions or procedures. A separate user interface is not required since the Configuration Tools functionality may be leveraged to flexibly configure these solution extensions through measures, rules and workbook templates specific to the customer's needs.

Configuration of the solution extensions may be limited based on your specific licensing agreement with Retek. Check with your system administrator for more information.

Chapter 2 – Configuring the Curve solution extension

Overview

Curve is an RPAS utility that can generate ratio arrays from historical data at user-specified intersections. The profiles generated by Curve can be used for various purposes: to convert the organization-level assortment plans into base level weekly sales forecasts, for generating seasonal forecasts, daily forecasts, or new product forecasting using lifecycle profiles.

RDF requires profiles (created by Curve) to determine how a source level forecast (ex. Item/Chain/Week) is spread down to the execution or final level (ex. Item/Store/Day). Profiles are generated using historical data and phase definitions based on the system configuration. Typical profiles include but are not limited to store participation, size distribution, and time profiles. Since a profile is used to spread the source level plan into the destination level, a naive way of calculating a profile is to obtain the ratio of sales at the execution level to the sales at the source level. However, in most situations the sales at the execution level are sparse, noisy, or do not exist (new product or new store). Instead of generating the profiles at the execution level, the sales can be aggregated, and profiles can be obtained at higher levels. The aggregated level at which a profile is generated is referred to as the *profile source level*. Profiles can be calculated at multiple source levels.

A separate profile is calculated for each hierarchy. Several profiles are used to spread the assortment plan into the final level, the user can generate daily profiles to be used for daily forecasting, seasonal profiles to be used in generating a seasonal forecast, or new product forecasting can be done using lifecycle profiles. In order to improve the usability of the system, Curve has the capability of processing several profiles at the same time. Therefore, instead of working on one profile at a time, the user can focus on the *processing plan*, which defines the configuration of one or multiple profiles.

Using the configuration tools, you must define the final and source profiles for your curve solution.

Creating a Curve solution extension

To create a Curve solution extension:

1. From the File Menu or Configuration Manager right-click menu, select New, and then select: Curve Solution – will create Curve solution
2. Type the name of the new solution extension in the text box.
3. Select the project in which to create the new solution extension.
4. Click Finish to create the new solution extension in the specified project.

Renaming an solution extension

To rename an solution extension:

1 Select the solution extension that you want to rename.

2 From the Configuration Manager right-click menu, select Rename.

3 Type the new name for the solution extension in the text box.

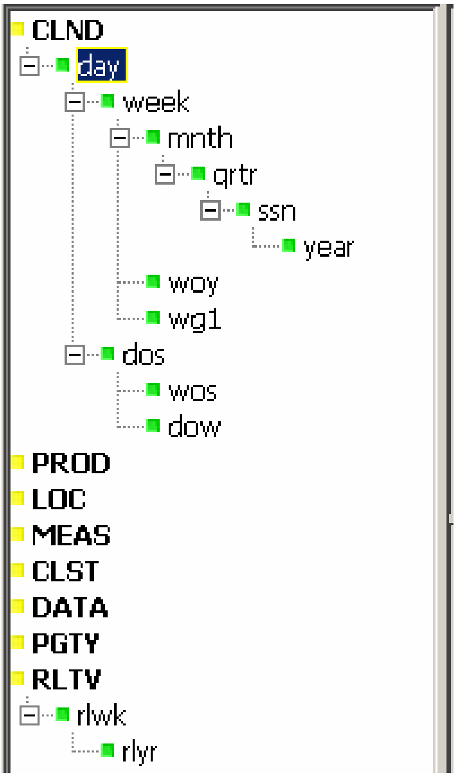
> **Note:** You cannot use ‘CurveExtension’ as a name. It is reserved by the system. It is the combination, not the capitalization of these names that are not allowed.

4 Click **Finish** to save the solution extension under the new name.

Curve hierarchy configuration requirements

- Optional: To support the generation of Lifecycle and Daily Seasonal Profiles, the calendar hierarchy must include the following:
  - Create a ‘day’ dimension at the base level of the calendar hierarchy.
  - Off the ‘day’ dimension, add ‘dos’ (Day of Season) dimensions.
  - Off the ‘dos’ (Day of Season) dimension, add ‘wos’ (Week of Season) dimension and ‘dow’ (Day of Week)
  - Off the ‘week’ (Week) dimension, add ‘woy’ (Week of Year). As a configuration option, you may also add ‘wg1’ (Week Grouping 1) or additional week groupings off ‘week’ to support user-defined week groupings once the application is being used.
- Optional: If Lifecycle profiles will be utilized, Curve also requires an additional hierarchy to support relative time periods. Configure the hierarchy as follows:
  - Add a New Hierarchy named RLTV
  - Add the ‘rlwk’ (Relative Week) dimension.

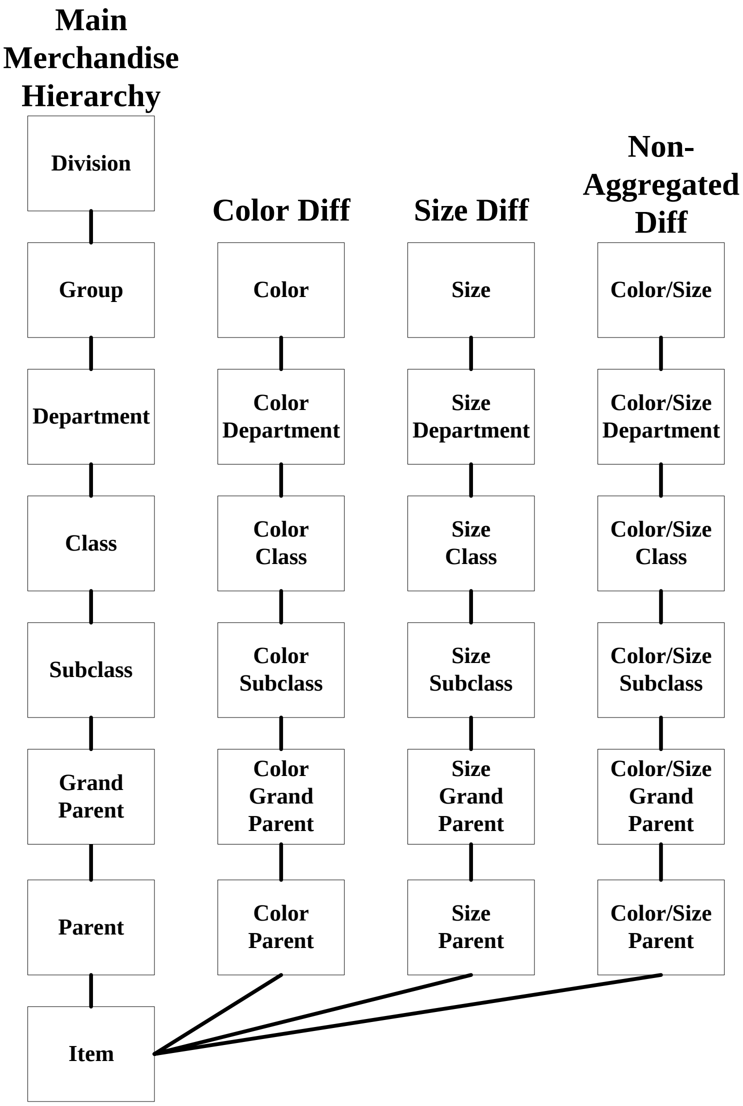
The following illustrates these additional hierarchy requirements that must be configured by the user:



Configuring Curve differentiator dimensions (optional) and merchandise hierarchy requirements to support RMS

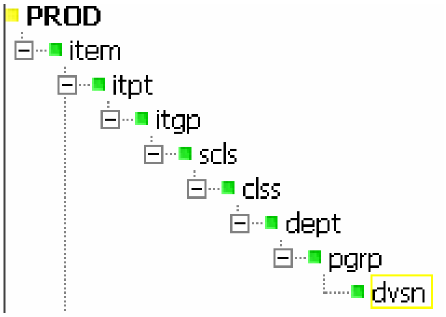
Configuring Differentiator dimensions (also referred to as Diff dimensions) within the merchandise hierarchy is optional. Differentiator dimensions allows the merchandise dimensions to be distinguished based upon an alternative attribute property such as Color, Size, Flavor (or other attributes properties required to support your merchandising needs).

Differentiator dimensions are the combination of each Differentiator and a dimension created off of the lowest dimension in the merchandise hierarchy (item) and only goes up as high as Department. The GA release of Curve is configured with 3 Differentiators however depending on the client’s needs, they may configure up to 10. Below is an example of how Color and Size Differentiator dimensions may be configured:



If Curve is to be integrated with RMS Allocation, the Diff dimensions configured in the RPAS Configuration Tools must map to the same Diff dimensions that are or will be configured in the RMS Allocation hierarchies. As well, Allocation also requires Non-Aggregated Differentiator dimensions. These dimensions allow for Diff Dimensions to be combined (as shown in the diagram above) and allow for Curve to generate profiles to support Allocation. Within Allocation, these Non-Aggregated Differentiators are represented by Diffs with Aggregation Indicators set to ‘No’.

Within the merchandise hierarchy, also required to support RMS is the ‘itpt’ (Item Parent) dimension off the ‘item’ (Item) dimension. And off of the ‘itpt’ dimension, add ‘itgp’ (Item Grandparent) dimension. The other aggregate dimensions above ‘item’, should be dimensions beginning off of ‘itgp’. The following illustrates the GA configuration of the merchandise hierarchies configured using Item Parent and Item Grandparent:



PROD
item
itpt
itgp
scls
clss
dept
pgrp
dvsn

Creating a final profile

To create a final profile:

1. On the Curve Manager page, click the F button. A new final profile is added and is assigned the next available number.
2. Specify the properties for the profile in the Curve Final Profiles pane.
   See Editing curve profiles for details.

Creating a source profile

To create a source profile:

1. On the Curve Manager page, check the box next to the final profile under which you want to create the source level.
2. Click the S button. A new source profile is added and is assigned the next available number.
3. Specify the properties for the source profile in the Curve Source Profiles pane.
   See Editing curve profiles for details.

Deleting a profile level

Deletion of a profile level will cause the system-assigned enumerated values in the Profile Name to renumber such that levels are in consecutive order starting with profile level 01. Deleting a profile level may impact any solution configuration that utilizes a specific profile level. For example, the following parameters within an RDF Solution configuration may be affected if profile levels are deleted or renumbered:

- Seasonal Profile
- Spreading Profile

As well, if the domain using the configuration has previously been installed, there is potential to lose data associated to a level that has been deleted or renumbered.

To delete a level:

1. On the Curve Manager page, select the number of the profile you want to delete.
2. Click the X button. The profile is deleted. If you delete a final profile, any source profiles that are associated with it will also be deleted..

Editing Curve profile properties

To edit a curve profile:

1. On the Curve Manager page, select the number of the level whose profile you want to edit.
2. For a curve final profile, enter the appropriate information for each of the following fields:
   - Profile Name – The system-assigned level number when a Final Profile is created.
   - Profile Label – The profile description that will be viewed by the user once the domain is created. The system will initially assign a Profile Label, but the user may overwrite this value.
     - Level Labels may not exceed forty characters.
     - It is recommended that Profile Labels include the Profile Name (which is the system-assigned profile number). The Profile Name is referenced in the RDF Solution to specify Spreading Profiles. Curve requires that profiles 1 through 9, be referenced as 01, 02, …, 09 when being specified as a Spreading Profile in the RDF Solution.
     - It is recommended that Profile Labels for Final Profiles include ‘Final’ or another designator that distinguishes the profile as a Final Profile.
     - RPAS will automatically put () around Profile Labels. The configuration specialist should not include these in their level label configuration or the installer will fail. An example of a Profile Label that would violate this requirement is (01 – chn->str-Final). It is acceptable as 01 – chn->str-Final.

    - '-' should not be used before or after the Profile Label. An example of a Profile Label that would violate this requirement is –01-chn->str-Final-. It is acceptable as 01 – chn->str-Final.

    - ':' should not be used at all in the Profile Label. An example of a Profile Label that would violate this requirement is 01: chn->str Final.

- Profile Type – A pick list of profile types used to determine the profile algorithm and validation required by the profile level. Profile Types are represented with pre-defined configuration information related to the Phase Flag and Reshape Boolean parameters. The Phase Flag is used to define whether or not the profile phase dates are required. When Reshape is set to TRUE, it will take the data and automatically normalize it along its entire calendar dimension.

    - Store Contribution Profile - The Store Contribution Profile is used to determine the data relationship between stores to aggregate dimensions in the location hierarchy.

      Phase Flag: false

      Reshape: false

    - Hourly Profile - The Hourly Profile is used to determine the data relationship between hour to an aggregate dimension in the calendar hierarchy (for example, hour to week).

      Phase Flag: false

      Reshape: false

    - Daily Profile - The Daily Profile is used to determine the data relationship between a given day to the week in which it belongs.

      Phase Flag: false

      Reshape: false

    - Daily Seasonal Profile - The Daily Seasonal Profile is used to determine the data relationship between a given day of the week to aggregate dimensions in the calendar hierarchy.

      Phase Flag: true

      Reshape: false

    - Life Cycle Profile - The Life Cycle Profile uses data along a user-defined training window, then stretches or shrinks data to fit a user-defined phase window.

    Phase Flag: true

    Reshape: true

    - Product Profile - The Product Profile is used to determine the data relationship between any two dimensions along the product hierarchy.

      Phase Flag: false

      Reshape: false

    - Size Profile - The Size Profile is used to determine the data relationship between any dimension in the size hierarchy and any dimension in the product hierarchy. A size hierarchy must be defined to use this profile type.

      Phase Flag: false

      Reshape: false

    - User Defined Profile - The User Defined Profile may be used to support any profile configuration.

      Phase Flag: false

      Reshape: false

- Profile Intersection – The initial intersection to which the profile will spread down.
- Aggregation Intersection – The intersection at which the profile will sum to one (or 100%).
- Approval Intersection – The intersection at which the profile is approved. Approval Intersection should be above or equal to the Aggregation Intersection.
- Maintenance Intersection – The intersection at which the profile is edited in the Profile Approval Workbook. The Maintenance Intersection should be equal to the Stored Intersection.
- Stored Intersection – The intersection at which the profile is stored. The Stored Intersection should be equal to the Profile Intersection.
- Default Source Level – The Profile Level that will be used in the calculation of the Final Profile. The desired Source Level must be created before it is an option in this pick list.

3 For a curve source profile, enter the appropriate information for each of the following fields:

- Profile Name – The system-assigned level number when a Source Profile is created.
- Profile Label – The profile description that will be viewed by the user once the domain is created. The system will initially assign a Profile Label, but the user may overwrite this value.
  - It is recommend that Profile Labels include the Profile Name (which is the system-assigned profile number). The Profile Name is referenced in the RDF Solution to specify Spreading Profiles. Curve requires that profiles 1 through 9, be referenced as 01, 02, …, 09 when being specified as a Spreading Profile in the RDF Solution.
  - RPAS will automatically put () around Profile Labels. The configuration specialist should not include these in their level label configuration or the installer will fail. An example of a Profile Label that would violate this requirement is (01 – chn->str-Final). It is acceptable as 01 – chn->str-Final.
  - '-' should not be used before or after the Profile Label. An example of a Profile Label that would violate this requirement is –01-chn->str-Final-. It is acceptable as 01 – chn->str-Final.
  - ':' should not be used at all in the Profile Label. An example of a Profile Label that would violate this requirement is 01: chn->str Final.
- Profile Type – A pick list of profile types used to determine the profile algorithm and validation required by the profile level. Profile Types are represented with pre-defined configuration information related to the Phase Flag and Reshape Boolean parameters. The Phase Flag is used to define whether or not the profile phase dates are required. When Reshape is set to TRUE, it will take the data and automatically normalize it along its entire calendar dimension.
  - Store Contribution Profile - The Store Contribution Profile is used to determine the data relationship between stores to aggregate dimensions in the location hierarchy.

    Phase Flag: false

    Reshape: false
  - Hourly Profile - The Hourly Profile is used to determine the data relationship between hour to an aggregate dimension in the calendar hierarchy (for example, hour to week).

    Phase Flag: false

    Reshape: false

  - Daily Profile - The Daily Profile is used to determine the data relationship between a given day to the week in which it belongs.

    Phase Flag: false

    Reshape: false

  - Daily Seasonal Profile - The Daily Seasonal Profile is used to determine the data relationship between a given day of the week to aggregate dimensions in the calendar hierarchy.

    Phase Flag: true

    Reshape: false

  - Life Cycle Profile - The Life Cycle Profile uses data along a user-defined training window, then stretches or shrinks data to fit a user-defined phase window.

    Phase Flag: true

    Reshape: true

  - Product Profile - The Product Profile is used to determine the data relationship between any two dimensions along the product hierarchy.

    Phase Flag: false

    Reshape: false

  - Size Profile - The Size Profile is used to determine the data relationship between any dimension in the size hierarchy and any dimension in the product hierarchy. A size hierarchy must be defined to use this profile type.

    Phase Flag: false

    Reshape: false

  - User Defined Profile - The User Defined Profile may be used to support any profile configuration.

    Phase Flag: false

    Reshape: false

- Profile Intersection – The initial intersection to which the profile will spread down.

- Aggregation Intersection – The intersection at which the profile will sum to one (or 100%).
- Maintenance Intersection – The intersection at which the profile is edited in the Profile Approval Workbook. The Maintenance Intersection should be equal to the Stored Intersection.
- Approval Intersection – The intersection at which the profile is approved. Approval Intersection should be above or equal to the Aggregation Intersection.
- Stored Intersection – The intersection at which the profile is stored. The Stored Intersection should be equal to the Profile Intersection.
- Source Data -– This is the measure name from which the profile is generated. No Source Data should be specified for any profiles that are dynamically generated to support Source Level Forecasting in RDF. See the RDF Users Guide for more information on Source Level Forecasting

Autogenerating hierarchies, measures, and rules

The following is the process to autogenerate the required hierarchies (not location, merchandise or calendar), measures, and rules:

1 On the Curve Manager page, click the G button.

The system automatically generates the required hierarchies, measures,and rules associated with a curve solution. Additional measures, rules, workbooks and wizards may be defined using the configuration tools.

You may continue to make changes to the Curve Solution configuration and the autogeneration process may be repeated as often as needed prior to the installation.

Note: Each time autogenerate is initiated, any user changes since the last autogenerate are wiped out.

Configuration restrictions on autogenerated hierarchies, measures, rules, workbooks, and wizards

The autogeneration process creates hierarchies, measures, and rules required to support the essential Curve functionality. The following outlines the strict guidelines you must adhere to if making changes to any by-product of autogeneration:

- Measures: Only the Label associated with a measure that is a by-product of autogeneration may be modified in the RPAS Configuration Tools.
- Rules: No modification or deletion of rules that are a by-product of autogeneration is allowed in the RPAS Configuration Tools.

Note: The configuration tools will not prevent these changes from being made; however, if these restrictions are not adhered to by the user, there will be problems in the domain build.

- After the auto generation of a Curve configuration, the Curve Solution will display exclamation icons next to the Measures and Rules Designers. Once you save and re-open the configuration, these icons will not appear.
- If customizations to measures and rules will be required, '_' (underscore) may not be used in any custom measure names and rules that ARE NOT to be expanded using the solution's classification scheme. The classifications in question are as follows:

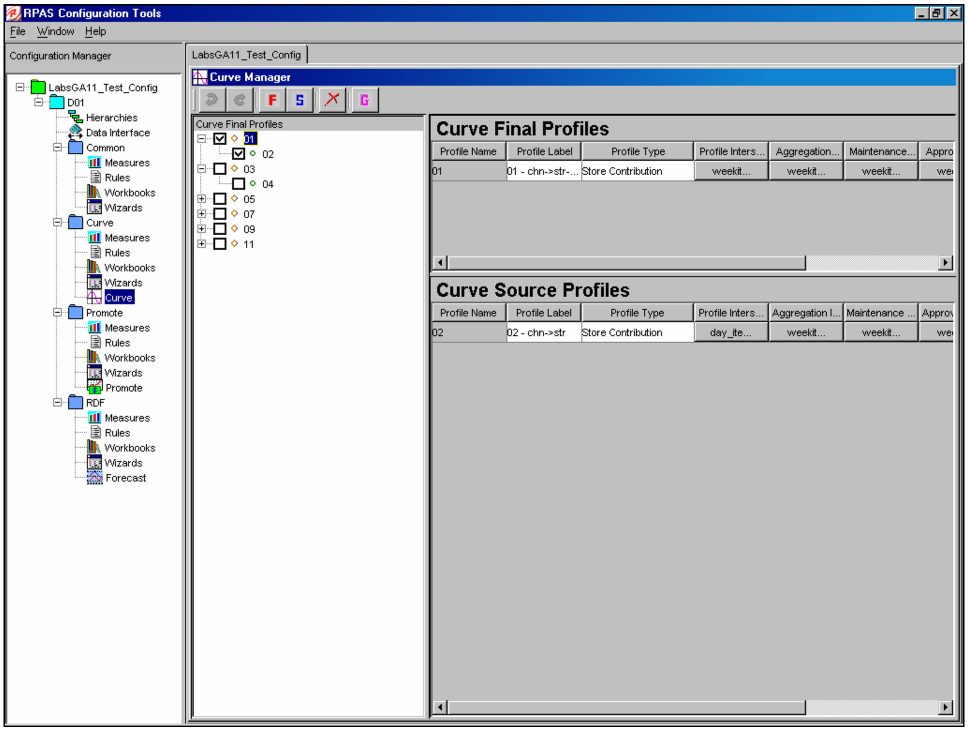
  _F: Expand measures and rules across final levels

  _S: Expand measures and rules across source levels

  _B: Expand measures and rules across birth dates

Curve example

The following example shows the Curve Manager window.



Curve Final Profiles

Profile Name	Profile Label	Profile Type	Profile Inters...	Aggregation...	Maintenance...	Appro
01	01 - chn->str-...	Store Contribution	weekit...	weekit...	weekit...	we

Curve Source Profiles

Profile Name	Profile Label	Profile Type	Profile Inters...	Aggregation I...	Maintenance ...	Approv
02	02 - chn->str	Store Contribution	day_ite...	weekit...	weekit...	we

Chapter 3 – Configuring the Retek Demand Forecasting solution extension

Overview

Retek Demand Forecasting™ is a Windows-based statistical forecasting solution. It uses state-of-the-art modeling techniques to produce high quality forecasts – with minimal human intervention. Forecasts produced by the Demand Forecasting system enhance the retailer's supply-chain planning, allocation, and replenishment processes, enabling a profitable and customer-oriented approach to predicting and meeting product demand.

Often, forecast information is required for items at the lowest levels in a hierarchy. Problems can arise when historic sales data for these items is too sparse and noisy to identify clear selling patterns. In such cases, generating a reliable forecast requires aggregating sales data from a low level up to a higher level in the hierarchy. After a forecast is generated at the higher level, the resulting data can be allocated (spread) back down to the lower level, based on the lower level's relationship to the total. Before you can spread forecast data back down to a lower level, you should have an understanding of the relationship between the lower level and the higher level dimensions.

Frequently, an additional forecast will be generated at the low level, to help determine this relationship. This low level is called the final forecast level. Forecast data at this level might be sufficient to generate reliable percentage-to-whole information, but the actual forecast numbers will be more robust when they are generated at an aggregate level. This aggregate level from which forecast data is spread is referred to as the source forecast level.

Some high-volume items may possess sufficient sales data for robust forecast calculations directly at the final forecast level. In these cases, forecast data generated at an aggregate level and then spread down to lower levels can be compared to forecasts run directly at the low level. Comparing the two forecasts, each generated at a different hierarchy level, can be an invaluable forecast performance evaluation tool.

Your Retek Demand Forecasting system may include multiple final forecast levels. Forecast data must appear at some final level for the data to be approved and exported to some other system.

Using the configuration tools, you must define the final levels and source levels for your RDF forecast solution.

Note: Ability to configure the RDF Forecast Solution may be limited based on your licensing agreement.

Creating a RDF solution extension

To create a RDF solution extension:

1 From the File Menu or Configuration Manager right-click menu, select New, and then select: RDF Solution – will create RDF solution

2 Type the name of the new solution extension in the text box.

3 Select the project in which to create the new solution extension.

4 Click Finish to create the new solution extension in the specified project.

Renaming an solution extension

To rename a solution extension:

1 Select the solution extension that you want to rename.

2 From the Configuration Manager right-click menu, select Rename.

3 Type the new name for the solution extension in the text box.

> **Note:** You cannot use ‘ForecastExtension’ as a name. It is reserved by the system. It is the combination, not the capitalization of these names that are not allowed.

4 Click **Finish** to save the solution extension under the new name.

Forecasting Merchandise and Location Hierarchy Requirements

With any Forecast Solution, configuration of the merchandise and location hierarchies must always include ‘item’ or ‘store’ at the lowest dimension level name. For example, if your lowest dimension in your merchandise hierarchy is ‘subclass’, the hierarchy dimension name must be ’item’, but the dimension label may be ‘subclass’. It is the dimension label that is viewed by the user within the domain. As well, the lowest dimension in your location hierarchy must be named ‘str’.

Creating a final forecast level

To create a final level:

1 On the Forecast Manager page, click the F button.

A new final level is added and is assigned the next available number.

2 Specify the properties for the level in the Forecast Final Levels pane.

See Editing forecast properties for details.

Creating a source forecast level

To create a source level:

1 On the Forecast Manager page, check the box next to the level under which you want to create the source level.

2 Click the S button.

A new source level is added and is assigned the next available number.

3 Specify the properties for the source level in the Forecast Source Levels pane.

See Editing forecast properties for details.

Deleting a forecast level

Deletion of a forecast level will cause the system-assigned enumerated values in the Level Name to renumber such that levels are in consecutive order starting with forecast level 01. Deleting a forecast level may impact any solution configuration that utilizes a specific level.

As well, if the domain using the configuration has previously been installed, there is potential to lose data associated to a level that has been deleted or renumbered.

To delete a level:

1 On the Forecast Manager page, select the number of the level you want to delete.

2 Click the X button.

The level is deleted. If you delete a final profile, any source profiles that are associated with it will also be deleted.

Editing RDF level properties

To edit forecast properties:

1. On the Forecast Manager page, select the number of the level whose properties you want to edit.
2. For a Forecast Final Level, enter the appropriate information for each of the following fields:
   - Level Name – The system-assigned level number when a Final Profile is created.
   - Level Label – The description that will be viewed by the user once the domain is created. The system will initially assign a Level Label, but the user may overwrite this value.
     - Level Labels may not exceed forty characters.
     - It is recommended that Level Labels include the Level Name (which is the system-assigned level number). Within the RDF Solution, Source Levels are now referenced as Level Names, rather than Level Labels as in previous releases. Unlike Curve, RDF does not require levels 1 through 9 to be specified as 01, 02, …, 09. For example, in the LabsGA configuration, level 03 (Level Name) has a Level Label of ‘3-itm/chn/week’. Within RDF, the Source Level for Level 01 is specified as ‘3’.
     - The calendar dimension specified in your Forecast Level Labels should match the case specified in the Calendar Dimension Labels in the Calendar Hierarchy configuration. For example, if you have configured a ‘day’ dimension in your calendar hierarchy and the label is specified as ‘day’ and you have a forecast level that is defined at day, the Forecast Level Label should be Item/Store/day, where ‘day’ matches the case of the Calendar Dimension Label.
       - The position of the calendar dimension in the level label string does not matter
       - Separators do not matter
     - It is recommended that Forecast Level Labels for Final Levels include ‘Final’ or another designator that distinguishes the level as Final.
     - RPAS will automatically put () around Forecast Level Labels. The configuration specialist should not include these in their level label configuration or the installer will fail. An example of a Forecast Level Label that would violate this requirement is (1:itm/str/week - Final). It is acceptable as:1- item/str/week - Final.

    - '-' should not be used before or after the Forecast Level Label. An example of a Forecast Level Label that would violate this requirement is: -1:itm/str/week – Final-

      It is acceptable as: 1-itm/str/week – Final

    - ':' should not be used at all in the Level Label. An example of a Level Label that would violate this requirement is 1: itm/str/week-Final Intersection – The hierarchy dimensions that define the forecasting level.

- Intersection – The hierarchy dimensions that define the forecasting level.
- Default Source Level – The level at which the aggregate, more robust forecast is run. The desired Source Level must first be created within the RDF Solution for it to be a selection in the pick list. No value is required if Source Level Forecasting is not being used at the Final Level. For more information on Source Level Forecasting, see the RDF 10.5 User's Guide.
- Source Data – The measure to be used as the input data (for example, POS) for the generation of forecasts. The value in this parameter is a measure name.
- Periodicity – The number of periods within the Calendar dimension defined in the forecast level Intersection. For example, if Intersection is defined at Week/Item/Store the Periodicity value will be 52 (52 weeks within a year).
- Forecast Method – A multi-select of forecast generation methods. The Default Forecast Method is also determined here. For a final level and its source levels, the list of forecast methods that you select should be the same. This will become the full list of available methods for a final level and its sources within the RDF solution. However, the 'Causal' method should be selected as a valid method for ONLY levels in which causal forecasting will be utilized. The following is a list of Forecast Methods that may be selected (see RDF 10.5 User's Guide for more information on each method):
    - No Forecast
    - Average
    - Simple
    - Intermittent
    - Simple/Intermittent
    - Trend
    - Additive Seasonal
    - Multiplicative Seasonal
    - Seasonal
    - AutoES
    - Causal

- This method is only available upon the creation of a Promote Solution. However, if Causal is selected as an available method for a level and you have not licensed Promote, the RDF batch forecast will not generate. If Promote is licensed you must include the Causal forecast method for any current causal forecast level. It is recommended that Causal is included for any levels that you may want to be active for causal forecasting in the future. Adding Causal at a later time after the domain is build will require the re-auto generation of measures and rules within the Configuration Tools and additional patching to support it.

- Bayesian
- Profile-based
- LoadPlan
- Copy

- Plan Data – Plan Data (sales plans) provide details of the anticipated shape and scale of an item's selling pattern. This information is required when Bayesian forecasting is used as the Forecast Method. The value in this parameter is a measure name.

- Seasonal Profile – A seasonal profile provides details of the anticipated seasonality of an item's selling pattern. The seasonal profile is required in conjunction with the Profile-based Forecasting Method. The seasonal profile can be generated or loaded, depending on your configuration. The value in this parameter is a measure name.

3 For a Forecast Source Level, enter the appropriate information for each of the following fields:

- Level Name – The system-assigned level number when a Source Profile is created.

- Level Label – The description that will be viewed by the user once the domain is created. The system will initially assign a Level Label, but the user may overwrite this value.

  - Level Labels may not exceed forty characters.

  - It is recommended that Level Labels include the Level Name (which is the system-assigned level number). Within the RDF Solution, Source Levels are now referenced as Level Names, rather than Level Labels as in previous releases. Unlike Curve, RDF does not require levels 1 through 9 to be specified as 01, 02, …, 09. For example, in the LabsGA configuration, level 03 (Level Name) has a Level Label of '3-itm/chn/week'. Within RDF, the Source Level for Level 01 is specified as '3'.

  - The calendar dimension specified in your Forecast Level Labels should match the case specified in the Calendar Dimension Labels in the Calendar Hierarchy configuration. For example, if you have configured a ‘day’ dimension in your calendar hierarchy and the label is specified as ‘day’ and you have a forecast level that is defined at day, the Forecast Level Label should be Item/Store/day, where ‘day’ matches the case of the Calendar Dimension Label.

    - The position of the calendar dimension in the level label string does not matter

    - Separators do not matter

  - RPAS will automatically put () around Forecast Level Labels. The configuration specialist should not include these in their level label configuration or the installer will fail. An example of a Forecast Level Label that would violate this requirement is (1:itm/str/week - Final). It is acceptable as:1- item/str/week - Final.

  - '-' should not be used before or after the Forecast Level Label. An example of a Forecast Level Label that would violate this requirement is: -1:itm/str/week – Final-

  - It is acceptable as: 1-itm/str/week – Final

  - ‘:’ should not be used at all in the Level Label. An example of a Level Label that would violate this requirement is 1: itm/str/week-Final Intersection – The hierarchy dimensions that define the forecasting level.

- Intersection – The hierarchy dimensions that define the forecasting level.

- Periodicity – The number of periods within the Calendar dimension defined in the forecast level Intersection. For example, if Intersection is defined at Week/Item/Store the Periodicity value will be 52 (52 weeks within a year).

- Forecast Method – Select one or more of the available forecast methods. You can also specify a default forecast method. For a final level and its source levels, the list of forecast methods that you select should be the same. This will become the full list of available methods for a final level and its sources within the RDF solution. However, the ‘Causal’ method should be selected as a valid method for ONLY levels in which causal forecasting will be utilized. The following is a list of Forecast Methods that may be selected (see RDF 10.5 User’s Guide for more information on each method):
  - No Forecast
  - Average
  - Simple
  - Intermittent

- Simple/Intermittent
- Trend
- Additive Seasonal
- Multiplicative Seasonal
- Seasonal
- AutoES
- Causal
  - This method is only available upon the creation of a Promote Solution. However, if Causal is selected as an available method for a level and you have not licensed Promote, the RDF batch forecast will not generate. If Promote is licensed you must include the Causal forecast method for any current causal forecast level. It is recommended that Causal is included for any levels that you may want to be active for causal forecasting in the future. Adding Causal at a later time after the domain is build will require the re-auto generation of measures and rules within the Configuration Tools and additional patching to support it.
- Bayesian
- Profile-based
- LoadPlan
- Copy

- Spreading Profile – Used in conjunction with Source Level Forecasting, this is the Profile Level(s) (configured within the Curve Solution) used to spread source level forecasts down to the final forecast level. . The value in this parameter may be a measure name, a profile level name, or any combination of these separated by commas. For example: 07,apvp11 (this is profile level 07 combined with the Approved Profile for level 11). For more information on Source Level Forecasting, see the RDF 10.5 User's Guide.

- Seasonal Profile – A seasonal profile provides details of the anticipated seasonality of an item's selling pattern. The seasonal profile is required in conjunction with the Profile based Forecasting Method. The seasonal profile can be generated by Curve or loaded, depending on your configuration. The value in this parameter is a measure name.

Autogenerating hierarchies, measures, and rules

The following process autogenerates hierarchies, measures, and rules:

1 On the Forecast Manager page, click the G button.

The system automatically generates the default hierarchies, measures, and rules associated with a forecast solution. Additional measures, rules, workbooks and wizards may be defined using the configuration tools.

You may continue to make changes to the Forecasting Solution configuration and the auto generation process may repeated as often as needed prior to the installation.

Note: Each time autogenerate is initiated, any user changes that were made are wiped out.

Configuration restrictions on autogenerated hierarchies, measures, rules, workbooks, and wizards

The autogeneration process creates hierarchies, measures, and rules required to support the essential RDF functionality. The following outlines the strict guidelines you must adhere to if making changes to any by-product of autogeneration:

- Measures: Only the Label associated with a measure that is a by-product of auto generation may be modified. The implication is that if autogeneration is done after a change has been made to a Label, this change will be overwritten.

- Rules: No modification or deletion of rules that are a by-product of autogeneration is allowed.

Note: The configuration tools will not prevent these changes from being made; however, if these restrictions are not adhered to by the user, there will be problems in the domain build.

- After the auto generation of an RDF configuration, the Forecast Solution will display exclamation icons next to the Measures, Rules, Workbooks and Wizard Designers. Once you save and re-open the configuration, these icons will not appear.

- After the auto generation of an RDF configuration, the following rule groups will be flagged (shown in red) as invalid, but will be installed properly in the domain:

  FAP_load_It

  FEV_load_It

  FINT_load

  FINT_calc_L01 (and any other levels in your configuration: L02, L03…)

- '_' (underscore) may not be used in any custom measure names and rules The exception to this are measures and rules that are to be expanded using the Curve solution's classification scheme. The classifications in question are as follows:

  _F: Expand measures and rules across final levels

  _S: Expand measures and rules across source levels

  _B: Expand measures and rules across birth dates

Common Solution Measures Required

The following measures must be added **as specified below** and realized in the Common solution for any RDF configuration:

Measure 1

Identifier: it_L_B
Label: Interim Forecast Level Birth
Data Type: Boolean
Base Intersection: week
Agg Method: ambig
Spread Method: repl
Base State: read
Agg State: read
Database: data/it

Measure 2

Identifier: it_LWORKBOOK
Label: Interim Forecast Level Wkbk
Data Type: Boolean
Base Intersection: week
Agg Method: ambig
Spread Method: repl
Base State: read
Agg State: read
Database: data/it

Measure 3

Identifier: it_F_B
Label: Interim Forecast Final Birth
Data Type: Boolean
Base Intersection: week
Agg Method: ambig
Spread Method: repl
Base State: read
Agg State: read
Database: data/it

Measure 4

Identifier: it_FWORKBOOK
Label: Interim Forecast Final Wkbk
Data Type: Boolean
Base Intersection: week
Agg Method: ambig
Spread Method: repl
Base State: read
Agg State: read
Database: data/it

Measure 5

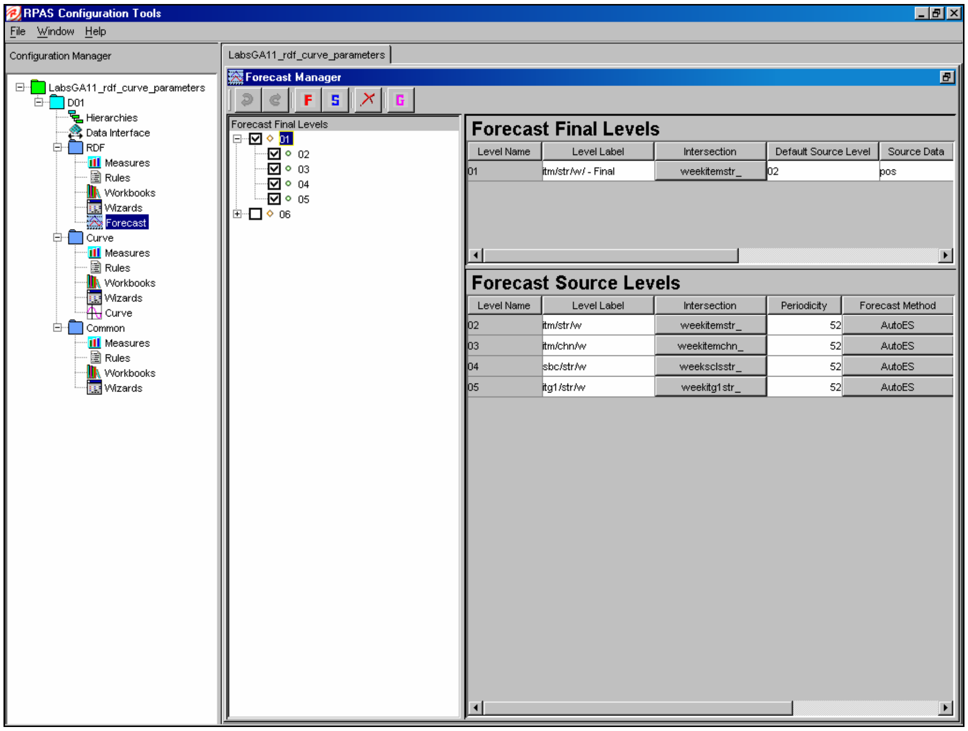
Identifier: tmpscalarbool
Label: Temp Scalar Bool for wkbk
Data Type: Boolean
Base Intersection: scalar
Agg Method: none
Spread Method: none
Database: data/temp

Measure 6

Identifier: tmpscalarstr
Label: Temp Scalar String for wkbk
Data Type: String
Base Intersection: scalar
Agg Method: none
Spread Method: none
Database: data/temp

RDF example

The following example shows the Forecast Manager window.



Forecast Final Levels

Level Name	Level Label	Intersection	Default Source Level	Source Data
01	itm/str/w/ - Final	weekitemstr_	02	pos

Forecast Source Levels

Level Name	Level Label	Intersection	Periodicity	Forecast Method
02	itm/str/w	weekitemstr_	52	AutoES
03	itm/chn/w	weekitemchn_	52	AutoES
04	sbc/str/w	weeksclsstr_	52	AutoES
05	itg1/str/w	weekitg1str_	52	AutoES

Chapter 4 – Configuring the Promote solution extension

Overview

Promote is an optional add-on solution to RDF that allows you to incorporate the effects of promotional and causal events, such as radio advertisements and holiday occurrences, into your time series forecasts. The promotional forecasting process uses both past sales data and promotional information to forecast future demand.

Using the configuration tools, you must define the promotions that will be utilized within the Promote Solution.

Creating a Promote solution extension

To create a Promote solution extension:

1 From the File Menu or Configuration Manager right-click menu, select New, and then select: Promote Solution – will create Promote solution

2 Type the name of the new solution extension in the text box.

3 Select the project in which to create the new solution extension.

4 Click Finish to create the new solution extension in the specified project.

Renaming an solution extension

To rename an solution extension:

1 Select the solution extension that you want to rename.

2 From the Configuration Manager right-click menu, select Rename.

3 Type the new name for the solution extension in the text box.

Note: You cannot use ‘PromoteExtension’ as a name. It is reserved by the system. It is the combination, not the capitalization of these names that are not allowed.

4 Click **Finish** to save the solution extension under the new name.

Promotion/Causal Forecast Levels

Promotion/causal forecasting levels are determined within the RDF Solution by selecting 'Causal' as a valid Forecasting Method for any source or final forecasting level.

Creating a promotion

To create a promotion:

1 On the Promote Manager page, click the P button. A new promotion is added and is assigned the default name "Prom".

2 Specify the properties for the promotion in the Promotions pane.
See Editing promotion properties for details.

Deleting a promotion

To delete a promotion:

1 On the Promote Manager page, select the promotion that you want to delete.

2 Click the X button. The promotion is deleted.

Editing promotion properties

To edit promotion properties:

1 On the Promotion Manager page, select the promotion whose properties you want to edit.

2 Type the appropriate information for each of the following fields:

- Promotion Name – The internal system identifier of the promotion. The system will initially assign a generic Promotion Name (Prom), but you must overwrite this value. The Promotion Name may not be greater than 4 characters. The following characters may not precede or follow the label that is entered in this field:
  '( )' Example: (xmas)
  '-' Example: -xmas-

  The following may not be use at all in the Promotion Name:

  ':' Example: xmas:

- Promotion Label – The label of the promotion that will be viewed by the user once the domain is created.
  - Promotion Labels may not exceed forty characters.
  - The following characters may not precede or follow the label that is entered in this field:
    '( )' Example: (xmas)
    '-' Example: -xmas-

    The following may not be use at all in the Promotion Name:

    ':' Example: xmas:
- Promotion Intersection – Independent of the causal forecasting levels, the promotion intersection is the hierarchy dimensions that define the promotion.

Autogenerating hierarchies, measures, and rules

To autogenerate hierarchies, measures, rules, workbooks, and wizards:

1 On the Promotion Manager page, click the G button.

The system automatically generates the default hierarchies, measures, rules associated with a Promote solution. Additional measures, rules, workbooks and wizards may be defined using the configuration tools.

You may continue to make changes to the Promote Solution configuration and the auto generation process may be repeated as often as needed prior to the installation.

Note: Each time autogenerate is initiated, any user changes that were made are wiped out.

Configuration restrictions on autogenerated hierarchies, measures, rules, workbooks, and wizards

The autogeneration process creates hierarchies, measures, rules required to support the essential Promote functionality. The following outlines the strict guidelines you must adhere to if making changes to any by-product of autogeneration:

- Measures: Only the Label associated with a measure that is a by-product of auto generation may be modified. The implication is that if auto generation is done after a change has been made to a Label, this change will be overwritten.

- Rules: No modification or deletion of rules that are a by-product of auto generation is allowed.

Note: The configuration tools will not prevent these changes from being made; however, if these restrictions are not adhered to by the user, there will be problems in the domain build.

- After the auto generation of a Promote configuration, the Promote Solution will display exclamation icons next to the Measures, Rules, Workbooks and Wizard Designers. Once you save and re-open the configuration, these icons will not appear.

- After the auto generation of a Promote configuration, the Rule Definition tool will not include any rules since all Promote-specific rules are dynamically created in the domain based on the configured causal forecast levels and promotions.

- '_' (underscore) may not be used in any custom measure names and rules The exception to this are measures and rules that are to be expanded using the RDF solution's classification scheme.   The classifications in question are as follows:

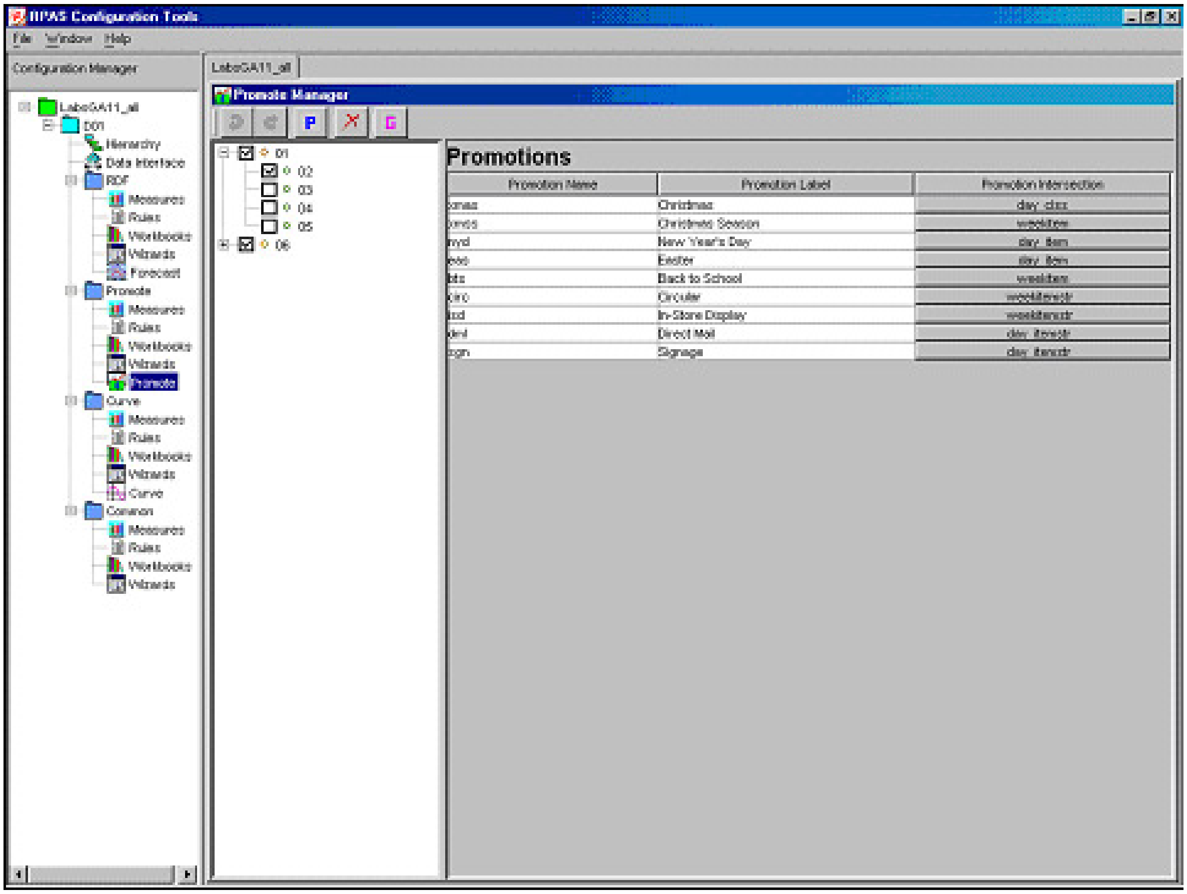
  _F: Expand measures and rules across final levels

  _S: Expand measures and rules across source levels

  _B: Expand measures and rules across birth dates

Promotion example

The following example shows the Promotion Manager window.



Promotions

Promotion Name	Promotion Label	Promotion Intersection
xmas	Christmas	day_clss
xmds	Christmas Season	weekitem
nyd	New Year's Day	day_item
eas	Easter	day_item
bts	Back to School	weekitem
circ	Circular	weekitemstr
isd	In-Store Display	weekitemstr
dml	Direct Mail	day_itemstr
sgn	Signage	day_itemstr

Chapter 5 – Configuring the Forecast procedure

Overview

RPAS may provide a time-series demand forecast to any solution configured via the RPAS Configuration Tools, also allowing for this forecast to be exported to other systems outside RPAS. The Forecast procedure provides only a small subset of the functionality available through RDF. The following highlights the differences between these solution extensions:

- The forecast produced by the Forecast procedure is a single-level forecast
- RDF allows for forecasts to be generated at aggregate levels in the data (to remove sparsity) then this forecast is spread down to the execution level using a profile.
- The Forecast procedure allows for a single forecasting method to be used for a forecast produced only on aggregated data in a domain.
- RDF allows for forecasting methods and forecasting parameters to be modified as needed down to the lowest level in your data.
- Using the Forecast procedure, all forecasted time-series are automatically approved.
- RDF allows for forecast adjustments and approvals to be made at the lowest level necessary in your data.

The following section contains the specifications and syntax for configuring the Forecast procedure.   Note that the syntax is slightly different than the standard RPAS functions and procedures described in the RPAS 11.0.4 Rule Functions Reference Guide.

Syntax Legend

[…] all options listed in brackets are optional

{…|…} options listed in “{}” with “|” separators are mutually exclusive (either/or)

{…,…} options listed in “{}” with “,” separators way are a complete set.

Bold – labels

Italics – indicates a temporary placeholder for a constant or a measure (will end with ‘meas’)

Italics/meas – indicates that the placeholder can be either a constant or a measure.

BoldItailics – indicates a numeric placeholder for the dynamic portion of a label. Usually a number from 1 to N

Normal – normal text that has to be there.

<u>Underlined And Large font</u> – Used to identify the function name.

Syntax

FORECAST: *FORMEAS* [, **INT**: *INTMEAS*, **CUMINT**:*CUMINTMEAS*, **PEAKS**:*PEAKSMEAS*, **CHMETHOD**:*METHMEAS*, **CHLEVEL**:*LVLMEAS*, **CHTREND**:*TRENDMEAS*, **ALERTS**:*ALERTSMEAS*] <- FORECAST(**MASK**:*MEASKMEAS*, {**STARTDATE**:*STARTDATE* | **STARTDATEMEAS**:*STARTDATEMEAS*}, **FORECASTLENGTH**:*FORECASTLENGTH*, **PERIOD**:*PERIOD* [

[,{**PROMO_*0***:PROMO0, **PROMOEFF_*0***:*PROMOEFF0*, **PROMOOVER_*0***:*PROMOOVER0,* **PROMOTYPE_*0***:*PROMOTYPE0*} … {,**PROMO_*N***:PROMON, **PROMOEFF_*N***:*PROMOEFFN*,**PROMOOVER_*N***:*PROMOOVERN*, **PROMOTYPE_*N***:*PROMOTYPEN*}],

HISTSTART: *HISTSTARTMEAS*, {**FRCSTSTARTMEAS**:*FRCSTSTARTMEAS* | **FRCSTSTART**:*FRCSTSTART*}, **MINWINTERS**:*MINWINTERSMEAS*, **MINHOLT**: *MINHOLTMEAS*, **MINCROSTON**:*MINCROSTON*, **MAXALPHA**:*MAXALPHA*, **MAXWINTERSALPHA**:*MAXWINALPHA*, **MAXPROFILEALPHA**:*MAXPROFILEALPHA*, **BAYESALPHA**:*BAYESALPHA*, **TRENDDAMP**:*TRENDDAMP*, **VALID_DD**:*VALID_DD*, **PROMO_IN_BASELINE**:*PROMO_IN_BASELINE*, **PLAN**:*PLAN*, **PROFILE**:*PROFILE*, **DDPROFILE**:*DDPROFILE*, **VERBOSE**:*VERBOSE*, **CUSTOMERID**:*CUSTOMERID*, **AGGPROF**:*AGGPROF*, **SPREADPROF**:*SPREADPROF*, **READMODE**:*READMODE*, **BAYESIAN_HORIZ**,*BAYESIAN_HORIZ*, **MINB**:*MINB*, **MAXB**:*MAXB*, **KEEPCLAMPEDMAXB**:*KEEPCLAMPEDMAXB*, **SMOOTHBASELINE**:*SMOOTHBASELINE*, **CAUSALMERGE**:*CAUSALMERGE*, **CAPS**:*CAPSMEAS*, **CAPRATIOS**:*CAPRATIOSMEAS*, **USECAPPING**:*USECAPPING*, **MINCAPHIST**:*MINCAPHIST*, **PLANINT**:*PLANINTMEAS*, **PLANCUMINT**:*PLANCUMINTMEAS*]

Examples

- Example 1: startdate as string

```
FORECASTOUT <-FORECAST(BAYESALPHA:0.15,
FORECASTLENGTH:12, HISTORY:RSAL, MASK:METHMASK1,
MAXALPHA:0.99, MAXB:50, MAXWINTERSALPHA:0.99,
MINCROSTON:5, MINHOLT:13, MINWINTERS:104, PERIOD:52,
PLAN:BAYES_PLAN1, SMOOTHBASELINE:TRUE,
STARTDATE:"D19980505", TRENDDAMP:0.5,
HISTSTART:H_STARTDATE1, FRCSTSTART:F_STARTDATE1)
```

- Example 2: startdate as parameter measure

```
FORECAST:FORECASTOUT <-FORECAST(BAYESALPHA:0.15,
FORECASTLENGTH:12, HISTORY:RSAL, MASK:METHMASK1,
MAXALPHA:0.99, MAXB:50, MAXWINTERSALPHA:0.99,
MINCROSTON:5, MINHOLT:13, MINWINTERS:104, PERIOD:52,
PLAN:BAYES_PLAN1, SMOOTHBASELINE:TRUE,
STARTDATEMEAS:TODAY*, TRENDDAMP:0.5,
HISTSTART:H_STARTDATE1, FRCSTSTART:F_STARTDATE1)
```

- Example 3: getting multiple results

```
FORECAST: FORECASTMEAS,  INT:INTMEAS,
CUMINT:CUMINTMEAS,  PEAKS: PEAKSMEAS <-
FORECAST(BAYESALPHA:0.15, FORECASTLENGTH:12,
HISTORY:RSAL, MASK:METHMASK1, MAXALPHA:0.99, MAXB:50,
MAXWINTERSALPHA:0.99, MINCROSTON:5, MINHOLT:13,
MINWINTERS:104, PERIOD:52, PLAN:BAYES_PLAN1,
SMOOTHBASELINE:TRUE, STARTDATEMEAS:TODAY*,
TRENDDAMP:0.5, HISTSTART:H_STARTDATE1,
FRCSTSTART:F_STARTDATE1)
```

Forecast Parameter Table

Left Hand Side – Label identifies a Left Hand Side(LHS) Parameter. In other words, these are the labels that identify output parameters that will appear on the left hand side of the equal sign. Otherwise the parameter is Right Hand Side(RHS) or input parameter.

Required – Label identifies required parameters.

Measure – Indicates whether the parameter is expected to be a measure or not.

Allowed to be Multiple – Allowed to be multiple measures.

Label	Parameter/ Result	Required	Measure	Allowed to be multiple	Type	Description
ALERTS	RESULT	N	Y	N	bool	A high-level forecast alert generated by the forecast engine
CHALPHA	RESULT	N	Y	N	double	ES alpha
CHLEVEL	RESULT	N	Y	N	int	ES level
CHMETHOD	RESULT	N	Y	N	int	Chosen method-Refer to Forecast Model/Model List table.
CHTREND	RESULT	N	Y	N	double	ES trend
CUMINT	RESULT	N	Y	N	double	Cumulative interval forecast
FORECAST	RESULT	Y	Y	N	double	forecast output
INT	RESULT	N	Y	N	double	Interval forecast
PEAKS	RESULT	N	Y	N	double	Peaks… for calculating baseline of forecast
AGGPROF	PARAMETER	N	Y	N	double	Profile to aggregate promotions to source level
BAYESALPHA	PARAMETER	N	N	N	double	Max Bayesian alpha
BAYESIAN_HORIZ	PARAMETER	N	Y	N	int	Horizon to which the Bayesian adjustment is applied
CAPRATIOS	PARAMETER	N	Y	N	double	Cap ratio for each time series
CAPS	PARAMETER	N	Y	N	double	Caps for each time series
CUSTOMERID	PARAMETER	N	N	N	string	A string ID assigned to a customer. Used only customers that require special processing
DDPROFILE	PARAMETER	N	Y	N	double	De-seasonalized demand measure for profile-based forecasting
FORECASTLENGTH	PARAMETER	Y	N	N	int	Length of forecast
FRCSTSTART	PARAMETER	N	N	N	datetime	Forecast start date

Label	Parameter/ Result	Required	Measure	Allowed to be multiple	Type	Description
FRCSTSTARTMEAS	PARAMETER	N	Y	N	datetime	Measure of forecast start dates
HISTORY	PARAMETER	Y	Y	N	double	Input measure forecast is based on
HISTSTART	PARAMETER	N	Y	N	double	Historical startdate
KEEPCLAMPEDMAXB	PARAMETER	N	N	N	double	Whether variables exceeding maxb are clamped or values are dropped and regression is re-run
KEEPCLAMPEDMINB	PARAMETER	N	N	N	double	Whether variables exceeding minb are clampled or values are dropped and regression is re-run
MASK	PARAMETER	Y	Y	N	bool	Array identifying what forecast method is used for each time series. Please see the Method list. Refer to Forecast Model/Model List table.
MAXALPHA	PARAMETER	N	N	N	double	Max alpha parameter
MAXB	PARAMETER	N	N	N	double	Maximum ratio between beta and baseline
MAXPROFILEALPHA	PARAMETER	N	N	N	double	Max alpha for profile method
MAXWINTERSALPHA	PARAMETER	N	N	N	double	Max alpha in Winters method
MINB	PARAMETER	N	N	N	double	Minimum ratio between beta and baseline
MINCAPHIST	PARAMETER	N	N	N	double	Minimum number of weeks before capping can be used.
MINCROSTON	PARAMETER	N	N	N	int	Min Croston history
MINHOLT	PARAMETER	N	N	N	int	Min Holt history
MINWINTERS	PARAMETER	N	N	N	int	Min winters history

Label	Parameter/ Result	Required	Measure	Allowed to be multiple	Type	Description
PERIOD	PARAMETER	Y	N	N	int	Forecasting period for calculating seasonal coefficients
PLAN	PARAMETER	N	Y	N	double	Plan measure
PLANCUMINT	PARAMETER	N	Y	N	double	Cumulative Interval of the plan associated with the plan(PARAMETER forecast). Bayesian only.
PLANINT	PARAMETER	N	Y	N	double	Interval of the plan associated with the plan(PARAMETER forecast). Bayesian only.
PROFILE	PARAMETER	N	Y	N	double	Seasonal profile measure
PROMO	PARAMETER	N	Y	Y	int	Promo variable measure(one for each promotion)
PROMO_IN_BASELINE	PARAMETER	N	Y	N	bool	Indicator that says if the promotion is incorporated in the baseline
PROMOEFF	PARAMETER	N	Y	Y	double	Calculated promotional effects (one per promotion)
PROMOOVER	PARAMETER	N	Y	Y	bool	Promo effect override measure (one for each promotion)
PROMOTYPE	PARAMETER	N	Y	Y	int	Promo type measure (one for each promotion)
READMODE	PARAMETER	N	N	N	int/enum	Whether mode is Random or Sequential.
SMOOTHBASELINE	PARAMETER	N	N	N	bool	Default to true in the causal method. When it is true, historical baseline is smoothed prior to future baseline forecast

Label	Parameter/ Result	Required	Measure	Allowed to be multiple	Type	Description
SPREADPROF	PARAMETER	N	Y	N	double	Profile to spread forecast to final level
STARTDATE/ STARTDATEMEAS:	PARAMETER	Y	N	N	date as string/ date as measure	Forecast start date . This or startdatemeas is required
TRENDDAMP	PARAMETER	N	N	N	double	Trend damping parameter
USECAPPING	PARAMETER	N	N	N	bool	Boolean measure that indicates whether capping is used or not.
VALID_DD	PARAMETER	N	N	N	int	Max non-zero history to use deseasonalized demand value for seasonal profile-based forecasting
VERBOSE	PARAMETER	N	N	N	bool	When verbose is set to true, there is much more information print out in the log file. It is useful for debugging

Forecast Method/Model List

Model	Numeric Value
FORECAST	1
SIMPLE	2
HOLT	3
WINTERS	4
CAUSAL	5
AVERAGE	6
NO FORECAST	7
COPY	8
CROSTON	9
M. WINTERS	10
A.WINTERS	11
SIMPLE CROSTON	12
BAYESIAN	13
LOADPLAN	14
PROFILE	15
SPREAD	16
NN	17

Parameter/Model Dependencies

The following is a list of models and the labels of the parameters that are required for the model to work.

- Bayesian – plan measure
- Profile – profile measure

Forecast Procedure Configuration Notes

The following notes are intended to serve as a guide for configuring the forecast procedure within the RPAS Configuration Tools.

- The following syntax must be used within the rule:

  ```
  forecast <- Forecast(…)
  ```

- Please refer to the Forecast documentation for the appropriate input parameters and output measures.

- The resultant measure (that is, frcstout) should be at the same intersection as your history measure (that is, pos). This will be the baseintersection of the final level.

- The Forecast procedure is a multi-result procedure, meaning that it can return multiple results with one procedure call within a rule. In order to get multiple results, the resultant measures must be configured in the Measure Tool and the specific measure label must be used on the left-hand-side (LHS) of the procedure call. The resultant measure parameters must be comma-separated in the procedural call as in the example below:

  ```
  forecast:frcstout,cumint:cumintout,int:intout<-
  Forecast(forecastlength:12,history:pos,mask:frcstmask,pe
  riod:26,startdatemeas:todaymeas)
  ```

- For standard deviations, specify the int measure on the LHS of the Forecast procedure call in addition to the forecast measure.

- Startdate (scalar, date type)

- Mask (picklist)

  - range value – see Forecast Method list

  - range syntax: 0(No Forecast), 1(Simple), 2(Holt), 3(Croston)

  - naval – for Planning, set to 8 for Bayesian for all products/locations

- History start date (by prod/loc)

- Forecast start date (by prod/loc)

- Forecast mask (by prod/loc)

- Need to determine the rule that will automatically update the startdatemeas each week with the current week’s date.

Forecast Method Configuration Notes

The following notes are intended to serve as a guide for configuring forecast methods within the RPAS Configuration Tools.

- The syntax for the range is # (value), # (value). For Example:

  0(No Frcst), 1(Simple), 2(Holt), 3(Croston)

- Spaces are allowed within the label inside the parentheses.

- Define your picklist measure with basestate and aggstate = write. If you define it as read, only the first value in the picklist range will show up.

- Set the measure type to int.

- For the Forecast Method measure, set the baseint to item/str (no week).

- You can specify the naval to a numeric integer within the range, such as 2 in the example above. Then, within the client, it automatically defaults to that value, such as Holt. If you specify the naval outside the range, then the picklist is displayed with NA. Given this ability, the planning teams can then specify 8 as the naval such that Bayesian is used for all item/stores. In this case, it may be desirable to configure the picklist measure as a scalar such that it cannot be changed on an item/store basis.

Chapter 6 – Configuring the LostSale Function

Overview

The purpose of Retek's LostSale module, which may also be referred to as "Preprocessing," is to correct past data points that represent unusual sales values not representative of a general demand pattern. Such correction may be necessary either when an item is out of stock and cannot be sold, resulting in usually low sales. Conversely, correction of data may also be necessary in a period when demand is unusually high. The LostSale module allows you to automatically make adjustments to the raw POS (Point of Sales) data, so that subsequent demand forecasts do not replicate undesired patterns caused by lost sales or unusually high demand.

The following section contains the specifications and syntax for configuring the LostSale function in the RPAS Configuration Tools. Note that the syntax is slightly different than the standard RPAS functions and procedures described in the RPAS 11.0.4 Rule Functions Reference Guide.

Syntax Legend

[…] all options listed in brackets are optional
{…|…} options listed in "{}" with "|" seperaters are mutually exclusive (either/or)
{…,…} options listed in "{}" with "," seperators way are a complete set.
Bold – labels
Italics – indicates a temporary placeholder for a constant or a measure (will end with 'meas')
Italics/meas – indicates that the placeholder can be either a constant or a measure.
BoldItailics – indicates a numeric placeholder for the dynamic portion of a label. Usually a number from 1 to N
Normal – normal text that has to be there.

Underlined And Large font – Used to identify the function name.

Syntax

```
LSOVER: @LSOVERMEAS, LS: @LSMEAS, [, TSALERT: @TSALERTMEAS,
SERVICE_LEVEL: @SERVICELEVELMEAS, STOCK_LEVEL:
@STOCKLEVELMEAS, FLP_FIRST: @FLPFIRSTMEAS, FLP_LAST:
@FLPLASTMEAS] = preprocess(SRC: @SRCMEAS, LSTODAY:
@LSTODAYMEAS, NPTS: @NPTSMEAS [, MIN_TSALERT:
@MINTSALERTMEAS, OUTAGE: @OUTAGEMEAS, TSMASK_DENSE:
@TSMASKMEAS, UP_ADJ_RATIO: @UPADJMEAS, DOWN_ADJ_RATIO:
@DOWNADJMEAS, REFERENCE: @REFMEAS, DEVIATION: @DEVMEAS {,
WINDOW: @WINDOWMEAS | , WINDOW1: @WINDOW1MEAS, WINDOW2:
@WINDOW2MEAS, WINDOW3: @WINDOW3MEAS, WINDOW4: @WINDOW4MEAS,
WINDOW5: @WINDOW5MEAS} {, ALPHA: @ALPHAMEAS, NPAS:
@NPASMEAS, NFUT: @NFUTMEAS} {, NSIGMA_MIN: @NSIGMA_MINMEAS,
NSIGMA_MAX: @NSIGMA_MAXMEAS | , NSIGMAOUT_MIN:
@NSIGMAOUT_MINMEAS, NSIGMAOUT_MAX: @NSIGMAOUT_MAXMEAS,
NSIGMAADJ_MIN: @NSIGMAADJ_MINMEAS, NSIGMAADJ_MAX:
@NSIGMAADJ_MAXMEAS} {, FRCST_MIN: @FRCST_MINMEAS,
HIST_MIN_FS: @HIST_MIN_FSMEAS} {, PRICE: @PRICEMEAS,
INVENTORY: @INVENTORYMEAS, HIST_MIN_MD: @HISTMINMDMEAS} ,
DELTA: @DELTAMEAS, LSOVER_REF: @LSOVERREFMEAS]
```

Example 1

LSOVER:@LSOVER1, LS:@LS1, TSALERT:@TSALERT1 = preprocess(SRC:@POS, METHODID:@MTHID, LSTODAY:@TODAY1, NPTS:@NPTS, WINDOW:@WIN)

Parameter Table Legend

Group – Whether the label is general enough to be used with all the filtering methods, or can only be used for specific filtering methods.

Input or Output – Whether the label is used as input or output label. In other words, the Label should appear on the left hand side or the right hand side of the function.

Required by All Methods – Whether the label is required for all filtering methods.

Measure – Whether the parameter is expected to be a measure or not.

Has Calendar Dimension – Whether the parameter measure has the calendar dimension or not.

Type – input/output parameter’s data type.

LostSale Parameter Table

Group	Label	Input Or Output	Required By All Methods	Measure	Has Calendar Dimension	Type	Default Value
General Parameters	LSOVER	Output	Y	Y	Y	Double	N/A
	LS	Output	Y	Y	Y	Double	N/A
	TSALERT	Output	N	Y	N	Boolean	N/A
	SERVICE_LEVEL	Output	N	Y	Y	Double	N/A
	STOCK_LEVEL	Output	N	Y	N	Double	N/A
	FLP_FIRST	Output	N	Y	N	Double	N/A
	FLP_LAST	Output	N	Y	N	Double	N/A
	SRC	Input	Y	Y	Y	Double	N/A
	METHODID	Input	Y	N	N	Double	N/A
	LSTODAY	Input	Y	Y	N	Double	N/A
	NPTS	Input	Y	N	N	Double	N/A
	MIN_TSALERT	Input	N	N	N	Double	N/A
	OUTAGE	Input	N	Y	Y	Boolean	N/A
	TSMASK_DENSE	Input	N	Y	N	Boolean	N/A
	UP_ADJ_RATIO	Input	N	Y	N	Double	1.0*
	DOWN_ADJ_RATIO	Input	N	Y	N	Double	1.0*
Method Specific Parameters	REFERENCE	Input	N	Y	Y	Double	N/A
	DEVIATION	Input	N	Y	Y	Double	N/A
	WINDOW	Input	N	N	N	Double	13
	WINDOW1	Input	N	N	N	Double	13
	WINDOW2	Input	N	N	N	Double	19
	WINDOW3	Input	N	N	N	Double	7
	WINDOW4	Input	N	N	N	Double	5
	WINDOW5	Input	N	N	N	Double	11
	ALPHA	Input	N	N	N	Double	0.2
	NPAST	Input	N	N	N	Double	5
	NFUT	Input	N	N	N	Double	5
	NSIGMA_MIN	Input	N	N	N	Double	3.0
	NSIGMA_MAX	Input	N	N	N	Double	3.0

Group	Label	Input Or Output	Required By All Methods	Measure	Has Calendar Dimension	Type	Default Value
Method Specific Parameters	FRCST_MIN	Input	N	N	N	Double	0.1
	HIST_MIN_FS	Input	N	N	N	Double	5
	NSIGMAOUT_MIN	Input	N	N	N	Double	3.0
	NSIGMAOUT_MAX	Input	N	N	N	Double	3.0
	NSIGMAADJ_MIN	Input	N	N	N	Double	1.5
	NSIGMAADJ_MAX	Input	N	N	N	Double	1.5
	PRICE	Input	N	Y	Y	Double	N/A
	INVENTORY	Input	N	Y	Y	Double	N/A
	HIST_MIN_MD	Input	N	N	N	Double	5
	DELTA	Input	N	Y	N	Double	1.0*
	LSOVER_REF	Input	N	Y	Y	Double	N/A

*: If the measure is not specified, the default value will be applied to each of the time series to be processed.

LostSale Parameter Description

Group	Label	Description
General Parameters	LSOVER	Adjusted source data. It is the Primary Result. LSOVER = SRC + LS
	LS	Adjustment on the source data.
	TSALERT	Boolean flag set to true when more than MIN_TSALERT number of data points have been modified
	SERVICE_LEVEL	SERVICE_LEVEL = SRC / LSOVER
	STOCK_LEVEL	Used by Mark Down filter only
	FLP_FIRST	First populated position. Used by FLP filter only
	FLP_LAST	Last populated position. Used by FLP filter only
	SRC	Source data
	METHODID	Filtering method ID
	LSTODAY	End date for filter processing
	NPTS	Number of points into history will be filtered
	MIN_TSALERT	Threshold value used to set off TSALERT
	OUTAGE	Outage indicator
	TSMASK_DENSE	Boolean to specify which time series will be processed
	UP_ADJ_RATIO	Upward adjustment ratio will be applied on LS
	DOWN_ADJ_RATIO	Downward adjustment ratio will be applied on LS
Method Specific Parameters	REFERENCE	Reference will be used for source data substitution
	DEVIATION	Standard deviation for confidence interval calculation by Forecast Sigma filters
	WINDOW	Filter window length for Standard Median filter
	WINDOW1	First round filter window length for Retek Median filter
	WINDOW2	Second round filter window length for Retek Median filter
	WINDOW3	Third round filter window length for Retek Median filter
	WINDOW4	Forth round filter window length for Retek Median filter
	WINDOW5	Fifth round filter window length for Retek Median filter
	ALPHA	Exponential coefficient used to evaluate past and future velocities
	NPAST	Maximum number of historical points to calculate past velocity
	NFUT	Maximum number of historical points to calculate future velocity

Group	Label	Description
Method Specific Parameters	NSIGMA_MIN	Number of std. Deviations for lower bound calculation
	NSIGMA_MAX	Number of std. Deviations for upper bound calculation
	FRCST_MIN	Forecast lower bound for Forecast Sigma filters
	HIST_MIN_FS	Minimum number of historical points required for Forecast Sigma filters
	NSIGMAOUT_MIN	Number of std. Deviations for lower outlier calculation
	NSIGMAOUT_MAX	Number of std. Deviations for upper outlier calculation
	NSIGMAADJ_MIN	Number of std. Deviations for lower bound calculation
	NSIGMAADJ_MAX	Number of std. Deviations for upper bound calculation
	PRICE	Historical price data. Used by Mark Down filter only
	INVENTORY	Historical inventory data. Used by Mark Down filter only
	HIST_MIN_MD	Minimum number of historical points. Used by Mark Down filter only
	DELTA	Ratio of reference will be used to copy or increase for OVERRIDE and INCREMENT filters
	LSOVER_REF	Data will be used to override SRC. Used by CLEAR filter only

LostSale Filtering Method List

Model	Comment	Numeric Value	Implemented	Required method-specific input parameters	Optional method-specific input parameters
MEDIAN5	Retek Median	0	X		WINDOW1, 2, 3, 4, 5
MEDIAN1	Std Median	1	X		WINDOW
OVERRIDE	Override	2	X	REFERENCE	DELTA
INCREMENT	Increment	3	X	REFERENCE	DELTA
ES_LT	Std ES	4	X	OUTAGE	ALPHA, NPAS, NFU
ES_LTS_LO	ES with daily seasonality	5			

Model	Comment	Numeric Value	Implemented	Required method-specific input parameters	Optional method-specific input parameters
ES_LTS_HI	ES with yearly seasonality	6			
ES_LTS_LOHI	ES with daily and yearly seasonality	7			
ES_CR	Causal Regression	8			
LS_ES_LT	Lost Sales -- Std ES	9	X	OUTAGE	ALPHA, NPAS, NFU
LS_ES_LTS_LO	Lost Sales -- ES with daily seasonality	10			
LS_ES_LTS_HI	Lost Sales -- ES with yearly seasonality	11			
LS_ES_LTS_LOHI	Lost Sales -- ES with daily and yearly seasonality	12			
LS_ES_CR	Lost Sales -- Causal Regression	13			
FRCST_SIGMA	Forecast and stddev algo	14	X	REFERENCE, DEVIATION	NSIGMA_MAX, NSIGMA_MIN, FRCST_MIN, HIST_MIN_FS
FRCST_SIGMA_EVENT	Forecast and stddev algo with event	15	X	OUTAGE, REFERENCE, DEVIATION	NSIGMAOUT_MAX, NSIGMAOUT_MIN, NSIGMAADJ_MAX, NSIGMAADJ_MIN, FRCST_MIN, HIST_MIN_FS

Model	Comment	Numeric Value	Implemented	Required method-specific input parameters	Optional method-specific input parameters
MARK_DOWN	Markdown removal -- interpolation on Mdarea	16			PRICE, INVENTORY, HIST_MIN_MD ,
CLEAR	Clear (clear specified result measures)	17	X		TSMASK_DENSE, LSOVER_REF
CLEAR_ALERT	Clear Alert measure	18			
NO_FILT	No Filtering (does not do anything)	19	X		
FLP_CALC	First and last populated locations calculation	20	X		

LostSale Filtering Method Descriptions

Standard Median

Recommended for getting data baselines on long time ranges when promo indicators are not available.

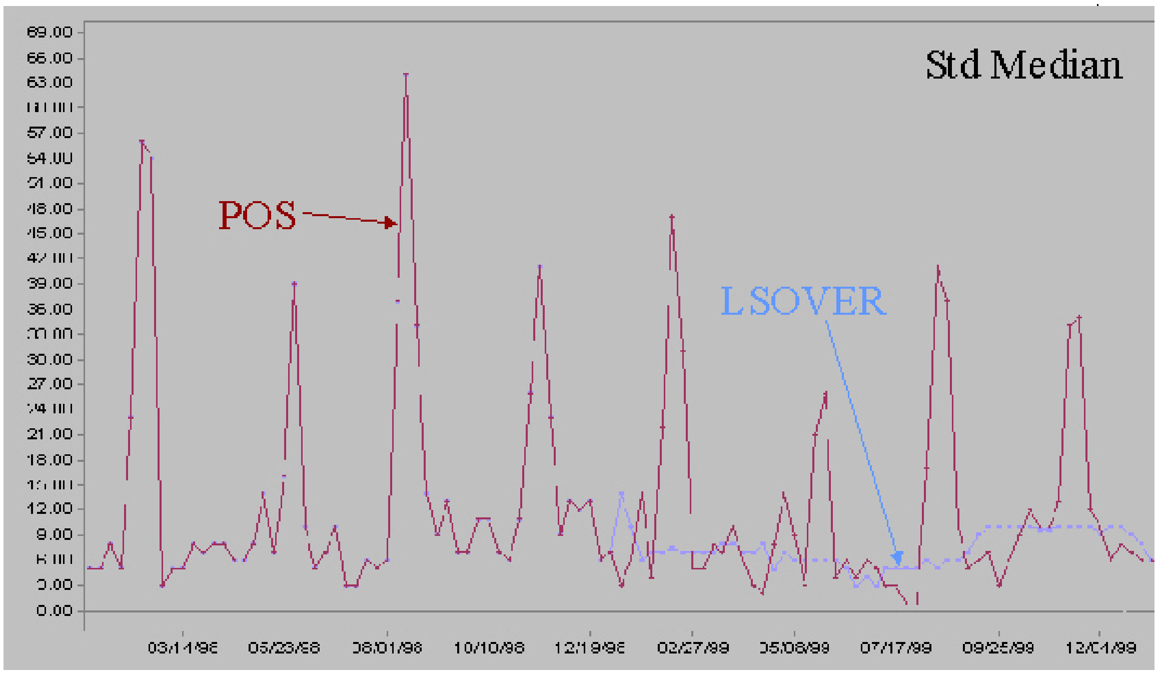
- A standard median filter implementation.
- Does not take outage info as an input.
- One optional parameter: window length.

Mathematical Formulation

LSOVER(t) = median value of SRC over [t-window/2, t+window/2],

Where: “window” is the parameter window length of the filter.

Example Chart



Std Median with "window" = 13 points

Example Usage

```
LSOVER:@lsover1, LS:@ls1, TSALERT:@tsalert1 =
preprocess(SRC:@pos, METHODID:@mthid, LSTODAY:@today1,
NPTS:@npts, WINDOW:@win)
```

Retek Median

Recommended for getting data baselines on long time ranges when promo indicators are not available.

- A sophisticated median filter that takes into consideration trends and improves side effects over the standard median filter. It makes 5 standard median filter passes.
- Does not take outage info as an input.
- Five optional parameters: window length for each pass.

Mathematical Formulation

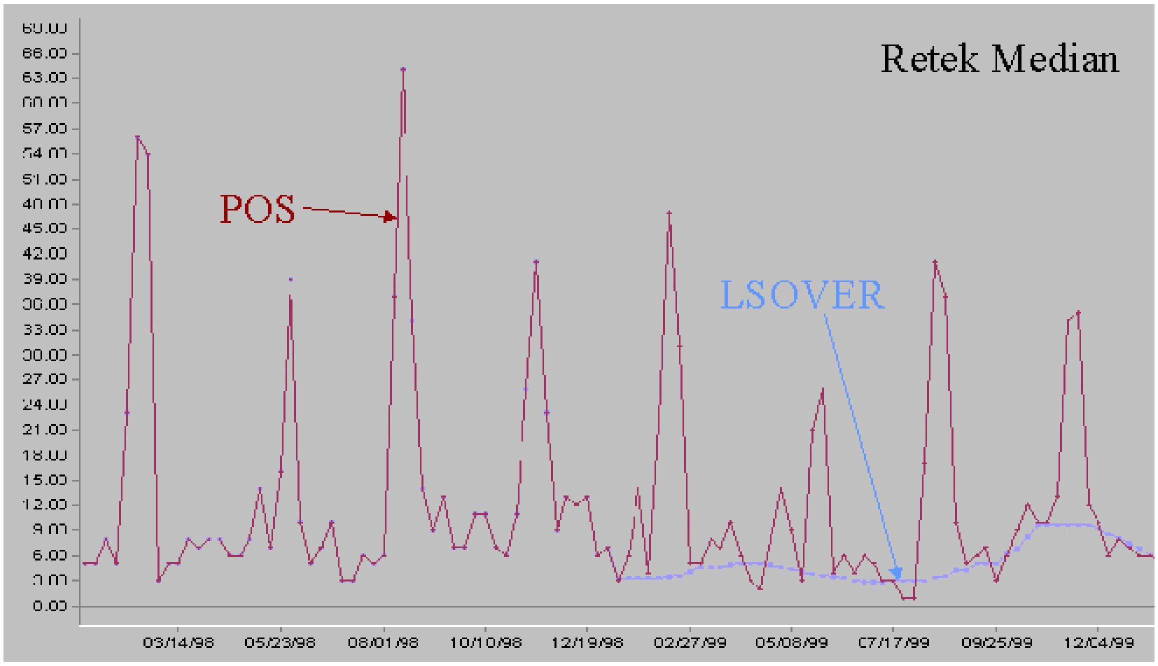
1 The first two passes recursively apply the standard median filter. The result is denoted by MEDIAN_2(t). The one-step difference of MEDIAN_2(t) is calculated, that is, DIFF_1(t) = MEDIAN_2(t) – MEDIAN_2(t-1). Then, the standard median filter is applied to DIFF_1(t). The result is denoted by MEDIAN_DIFF_1(t).

2 Using MEDIAN_DIFF_1(t), a first smoothed version (that is, baseline) of the source data is calculated at the third step: SMOOTH_1(t) = SMOOTH_1(t-1) + MEDIAN_DIFF_1(t) on points where the absolute deviation of SRC(t) over its mean is larger than half of the global absolute standard deviation, SMOOTH_1(t) = SRC(t) otherwise.

3 To prepare for the fourth pass, the one-step difference of SMOOTH_1(t) is calculated, that is, DIFF_2(t) = SMOOTH_1(t) - SMOOTH_1(t-1). An average version of DIFF_2(t) is calculated using the standard median filter. The result is denoted by AVG_DIFF_2(t). The result of the fourth pass is SMOOTH_2(t) = SMOOTH_2(t-1) + AVG_DIFF_2(t).

4 Finally, LSOVER(t) is the result of applying the standard median filter to SMOOTH_2(t).

Example Chart



Retek Median with default parameters

Example Usage

```
LSOVER:@lsover1, LS:@ls1, TSALERT:@tsalert1 =
preprocess(SRC:@pos, METHODID:@mthid, LSTODAY:@today1,
NPTS:@npts, WINDOW1:@win, WINDOW2:@win2,
WINDOW3:@win3, WINDOW4:@win4, WINDOW5:@win5)
```

Standard Exponential Smoothing

Recommended for removing specific spikes of low or non-seasonal data when spike indicators are available.

- Based on standard Exponential Smoothing calculations of past and future sales velocities.
- Must have “unusual event” measure info as input (that is, also referred to as outage).
- Does not take into consideration seasonal components.
- 3 optional parameters:
    - Alpha (ES parameter used to evaluate past and future velocities).
    - Maximum number of historical points to calc past velocity.
    - Maximum number of future points to calc future velocity.

Mathematical Formulation

Std ES is the standard Exponential Smoothing filter. It preprocesses a subset of points as predetermined by an input measure. For every contiguous sequence of points to adjust, say between t_f and t_l, a past velocity and a future velocity are calculated using an exponentially weighted average. For the points between t_f and t_l, the adjustment is calculated as a linear interpolation of the past and future velocities.

$$Past\_Velocity = \frac{\sum_{i=1}^{np}(1-\alpha)^{i-1} * SRC(t_f - i)}{\sum_{i=1}^{np}(1-\alpha)^{i-1}}$$

$$Future\_Velocity = \frac{\sum_{i=1}^{nf}(1-\alpha)^{i-1} * SRC(t_l + i)}{\sum_{i=1}^{nf}(1-\alpha)^{i-1}}$$

$$LSOVER(t) = Past\_Velocity + \frac{Future\_Velocity - Past\_Velocity}{t_l - t_f + 2} * (t - t_f + 1), \forall t \in [t_f, t_l]$$

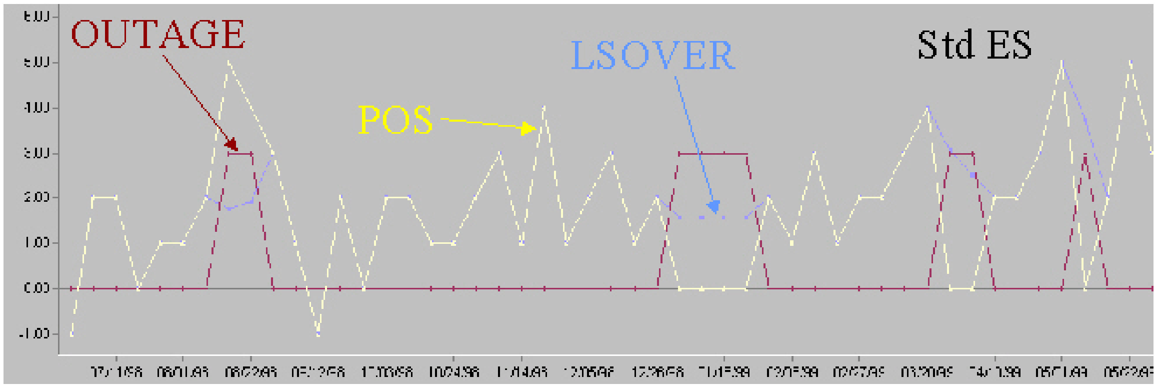
Where:

"α" is the exponential coefficient used to evaluate past and future velocities.

"np" is the maximum number of historical points to calc past velocity.

"nf" is the maximum number of future points to calc future velocity.

Example Chart



Std ES with "α" = 0.2, "np" = 2 weeks, and "nf" = 2 weeks

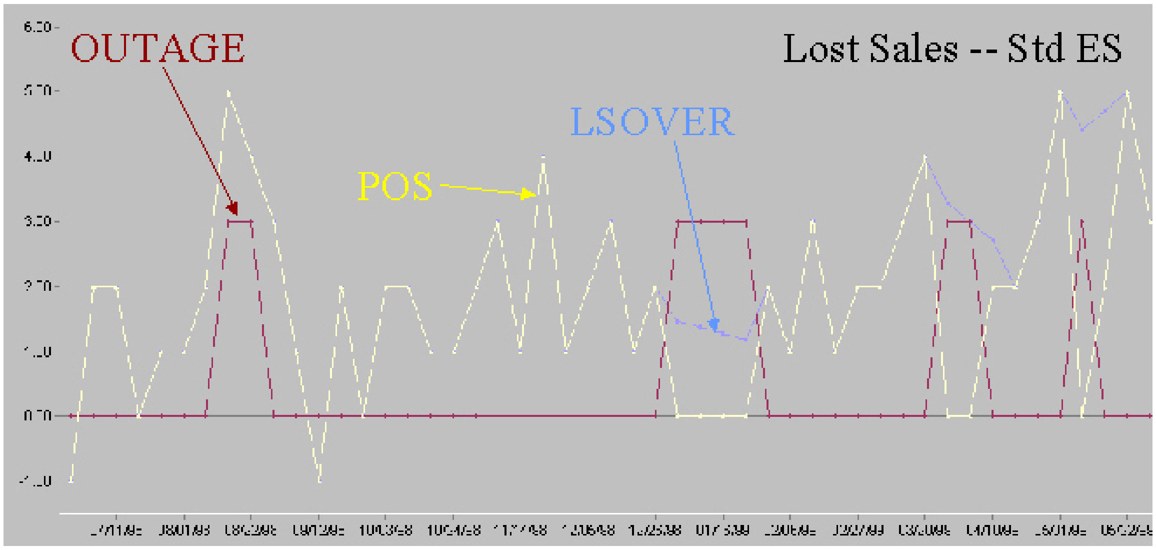
Example Usage

```
LSOVER:@lsover1, LS:@ls1, TSALERT:@tsalert1 =
preprocess(SRC:@pos, METHODID:@mthid, LSTODAY:@today1,
NPTS:@npts, OUTAGE:@outage1, ALPHA:@alpha,
NPAST:@npast, NFUT:@nfut)
```

Lost Sales – Standard Exponential Smoothing

Functions the same as Std ES except that it only adjusts lost sales (that is, negative spikes).

Example Chart



Lost Sales -- Std ES with “α ” = 0.2, “np ” = 2 weeks, and “nf ”= 2 weeks

Example Usage

LSOVER:lsover1, LS:ls1, TSALERT:tsalert1 = preprocess(SRC:pos, METHODID:9, LSTODAY:today1, NPTS:30, OUTAGE:outage1, ALPHA:0.2, NPAST:5, NFUT:5)

Forecast Sigma

Recommended for removing recent spiky data points when approved forecasts and approved confidence intervals are available on the filtering window but spike indicators are not available. This method is based on the principle that if a data point significantly deviates from an approved forecast, then this data point is likely to be an unusual event that should be overridden in the source measure (POSOVER) used by the forecasting engine. It is adjusted by bringing the override value within some bounds of the approved forecast as defined by a proportional coefficient scalar of the forecasts' standard deviation.

- Does not take outage info as an input.
- Two required parameters:
  - Approved forecast array.
  - Approved standard deviation array of forecast.
- Four optional parameters:
  - Number of std. deviations for upper bound.
  - Number of std. deviations for lower bound.
  - Forecast lower bound.
  - Minimum item history (# points) required for filtering.

Mathematical Formulation

This method relies on already approved forecasts with their corresponding confidence intervals. It adjusts the points that are far (as defined by a multiple of the forecast standard deviation) from their corresponding previously approved forecasts by bringing the override values to their closest confidence interval bounds.

```
IF # historical points < MinHist THEN
    LSOVER(t) = SRC(t)
ELSE IF forecast(t) < MinFrcst THEN
    forecast(t) = MinFrcst AND σ = MinFrcst
ELSE IF σ = 0 THEN
    IF forecast(t) < 1.0 THEN
```

```
$\sigma$ = forecast(t)

ELSE $\sigma = \sqrt{\text{forecast(t)}}$

IF SRC(t) > forecast(t) + nsu*$\sigma$ THEN

LSOVER(t) = forecast(t) + nsu*$\sigma$

        ELSE IF SRC(t) < forecast(t) – nsl*$\sigma$ THEN

LSOVER(t) = forecast(t) - nsl*$\sigma$

ELSE LSOVER(t) = SRC(t)
```

Where:

“nsu” is the number of std. deviations for upper bound.

“nsl” is the number of std. deviations for lower bound.

“MinFrcst” is the forecast lower bound.

“MinHist” is the minimum item history (# points) required for filtering.

Example Chart

Error! Objects cannot be created from editing field codes.

Lost Sales – Forecast Sigma with nsu = 3, nsl = 3, minFrcst = 0.1 and minHist = 5 weeks

Example Usage

```
LSOVER:@LSOVER1, LS:@LS1, TSALERT:@TSALERT1 =
preprocess(SRC:@POS, METHODID:@mthid, LSTODAY:@TODAY1,
NPTS:@npts, REFERENCE:@forecast1, DEVIATION:@dev1,
NSIGMA_MIN:@nsigma_min, NSIGMA_MAX:@nsigma_max,
FRCST_MIN:0.1, HIST_MIN_FS:@hist_min_fs)
```

Forecast Sigma Event

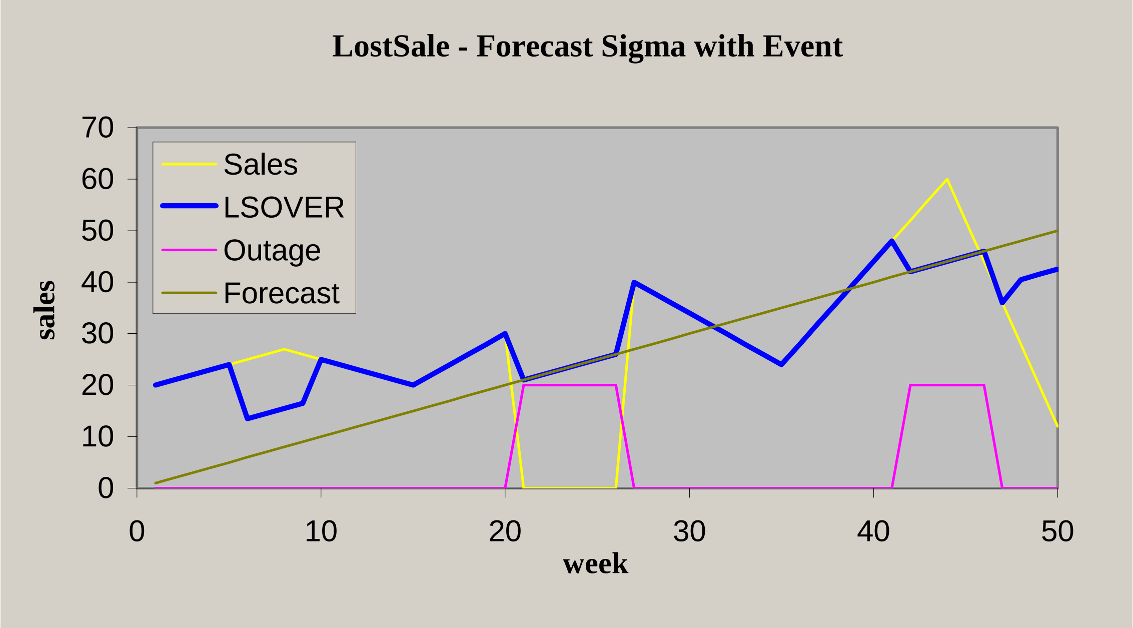
Similar to ‘Frcst Sigma’ and it takes an outage (i.e., event) indicator to further process.

Mathematical Formulation

When the outage/event mask is ON: LSOVER(t) = forecast(t)

When the outage/event mask is OFF: if the data points that are outside the outliers calculated through NSIGMAOUT_MIN and NSIGMAOUT_MAX, they will be brought into the confidence interval bounds defined through NSIGMAADJ_MIN and NSIGMAADJ_MAX.

Example Chart



Lost Sales – Forecast Sigma Event with nsigmaout_min = 3, nsigmaout_max = 3, nsigmaadj_min = 1.5, nsigmaadj_max = 1.5,

minFrcst = 0.1 and minHist = 5 weeks

Example Usage

```
LSOVER:@LSOVER1, LS:@LS1, TSALERT:@TSALERT1 =
preprocess(SRC:@POS, METHODID:@mthid, LSTODAY:@TODAY1,
NPTS:@npts, OUTAGE:@outage1, REFERENCE:@forecast1,
DEVIATION:@dev1, NSIGMAOUT_MIN:@nsigmaout_min,
NSIGMAOUT_MAX:@nsigmaout_max,
NSIGMAADJ_MIN:@nsigmaadj_min,
NSIGMAADJ_MAX:@nsigmaadj_max, FRCST_MIN:@frcst_min,
HIST_MIN_FS:@hist_min_fs)
```

Override

This method overrides the destination measure with the source measure adjusted by the adjustment percentage, according to the mask. It is recommended for filling data gaps when an existing reference measure exists as a default value.

- It is a simple data copy of a given percentage of the reference data to copy from.
- May or may not take outage (i.e., event) info as an input to mask the operation.
- Two required parameters:
  - Reference measure to copy data from.
  - Source measure for the original data
- One optional parameter:
  - Ratio of reference to actually copy.

Mathematical Formulation

This method uses the following parameters:

- A source measure that can be any measure in the system as long as it has the same intersection as the destination measure.
- A reference measure that can be any measure in the system as long as it has the same intersection as the destination measure.
- A destination measure that can be any measure in the system as long as it has the same intersection as the source measure.
- A mask that is a Boolean measure that has the same intersection as the source and destination measures.
- An adjustment percentage

This method overrides the destination measure with the source measure adjusted by the adjustment percentage, according to the mask.

Let:

S(i) is the value in cell (i) of the source measure

R(i) is the value in cell (i) of the reference measure

D(i) is the value in cell (i) of the destination measure

M(i) is the value of cell (i) of the mask

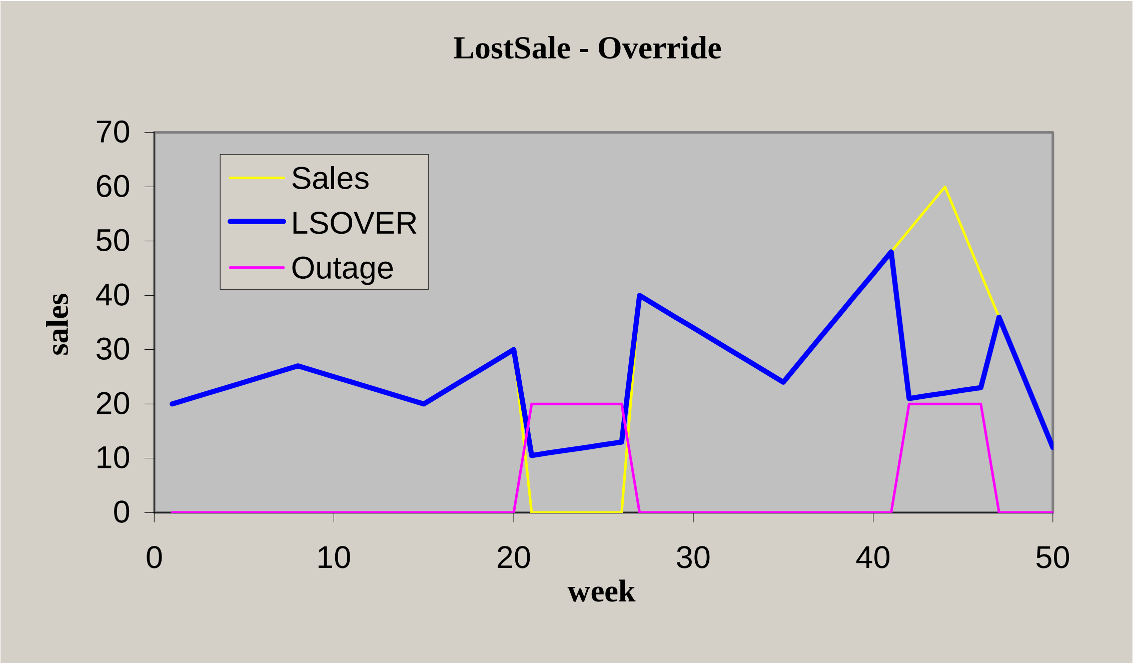
a is an adjustment percentage

The result of the override method is:

D(i) = a * R(i) if M(i) is TRUE.

D(i) = S(i) if M(i) is FALSE.

Example Chart



Lost Sales – Override with delta = 0.5

Example Usage

```
LSOVER:@lsover1, LS:@ls1, TSALERT:@tsalert1 =
preprocess(SRC:@pos, METHODID:@mthid, LSTODAY:@today1,
NPTS:@npts, REFERENCE:@ref1, OUTAGE:@outage1,
DELTA:@delta1)
```

Increment

This method increments or decrements the destination measure by the source measure adjusted by the adjustment percentage, according to the mask. It is recommended for updating outliers or data gaps when an existing reference measure exists as a default adjustment.

- It is a simple data increment of a given percentage of the reference data to copy from.
- May or may not take outage (i.e., event) info as an input to mask the operation.
- One required parameter:
- Reference measure to increment by.
- One optional parameter:
- Ratio of reference to actually increment by.

Mathematical Formulation:

This method uses the following inputs:

- A source measure that can be any measure in the system as long as it has the same intersection as the destination measure.
- A reference measure that can be any measure in the system as long as it has the same intersection as the destination measure.
- A destination measure that can be any measure in the system as long as it has the same intersection as the source measure.
- A mask that is a Boolean measure that has the same intersection as the source and destination measures.
- An adjustment percentage

This method increments or decrements the destination measure by the source measure adjusted by the adjustment percentage, according to the mask.

Let:

S(i) is the value in cell (i) of the source measure

R(i) is the value in cell(i) of the reference measure

D(i) is the value in cell (i) of the destination measure

M(i) is the value of cell (i) of the mask

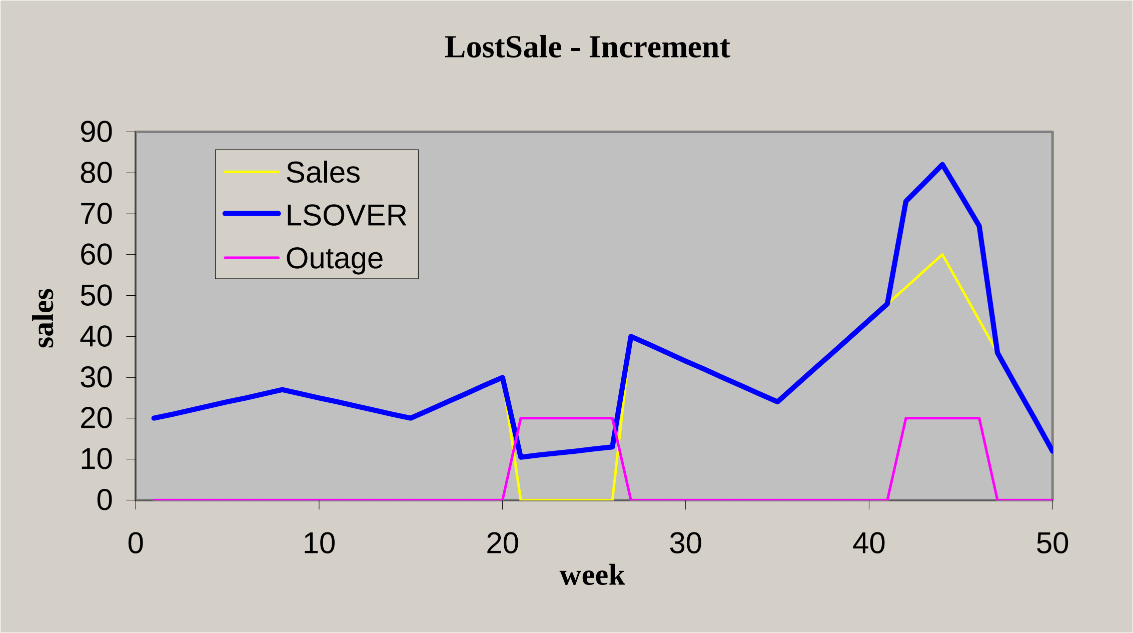
a is an adjustment percentage (can be between (–100%) and (+100%))

The result of the reduction method is:

D(i) = S(i) + a * R(i) if M(i) is TRUE.

D(i) = S(i) if M(i) is FALSE.

Example Chart



Lost Sales – Increment with delta = 0.5

Example Usage

```
LSOVER:@lsover1, LS:@ls1, TSALERT:@tsalert1 =
preprocess(SRC:@pos, METHODID:@mthid, LSTODAY:@today1,
NPTS:@npts, REFERENCE:@ref1, OUTAGE:@outage1,
DELTA:@delta1)
```

Clear

Used for canceling the effect of some former preprocessing adjustments.

- Does not take outage info as an input.
- May or may not take time series mask (does not have calendar dimension) input to retain results for certain time series.
- If time series mask is specified, one duplicated LSOVER measure must be provided in addition to the original LSOVER measure.

Mathematical Formulation

IF TimeSeriesMask is provided && TimeSeriesMask = false **THEN**

LSOVER(t) = LSOVER_REF(t)

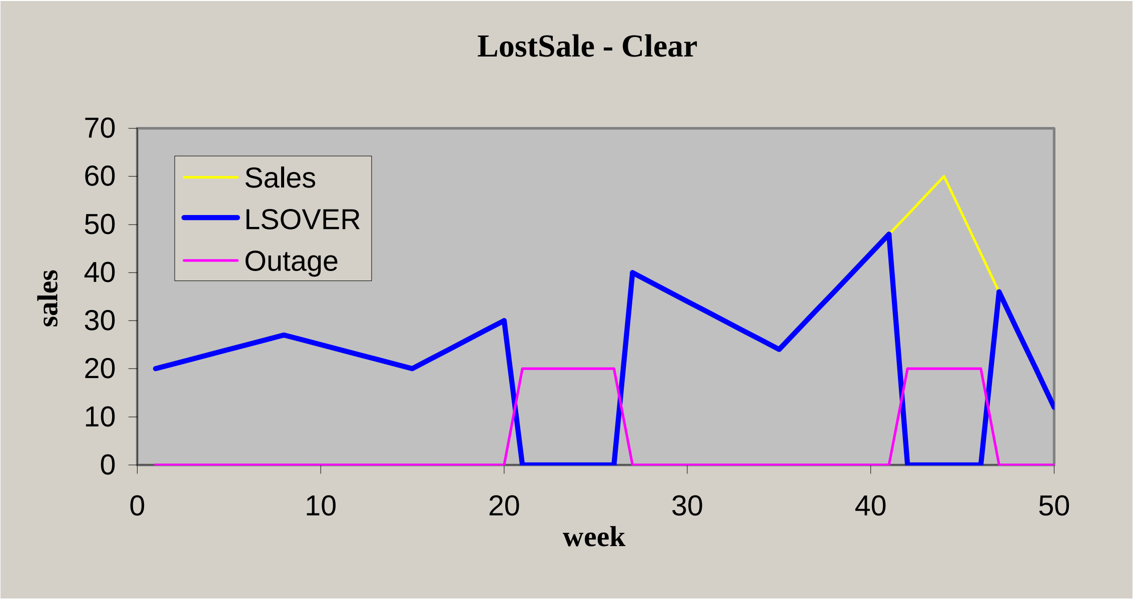
LS(t) = LSOVER_REF(t) – SRC(t)

ELSE

LSOVER(t) = 0

LS(t) = 0

Example Chart



Lost Sales – Clear with TS_Mask

Example Usage

1 Clear all:

```
LSOVER:@LSOVER1, LS:@LS1, TSALERT:@TSALERT1 =
preprocess(SRC:@POS, METHODID:@mthid, LSTODAY:@TODAY1,
NPTS:@npts)
```

2 Partial clear with mask input:

```
LSOVER:@LSOVER1, LS:@LS1, TSALERT:@TSALERT1 =
preprocess(SRC:@POS, METHODID:@mthid, LSTODAY:@TODAY1,
NPTS:@npts, TSMASK_DENSE:@tsMask1,
LSOVER_REF:@lsoverref1)
```

LostSale Function Configuration Notes

The following notes are intended to serve as a guide for configuring the LostSale function within the RPAS Configuration Tools.

1 The LostSale function is a multi-result function, meaning that it can return multiple results within one function call within a rule. In order to get multiple results, the resultant measures must be configured in the Measure Tool and the specific measure label must be used on the left-hand-side (LHS) of the function call. The resultant measure parameters must be comma-separated in the function call as in the example.

2 Because different filtering methods require different input parameters, it is necessary that every input parameter (measure or constant) must be accompanied by the corresponding label. All the input measure parameters must be configured and registered before the function call. The input parameters must be comma-separated in the function call as in the example.

3 The LostSale function library must be registered after the domain build using the regfunction RPAS utility.

4 The LostSale function required all the input and output measures using the same intersections. Mixed input/output measure intersections should be aligned to the same calculation intersection with other RPAS function/procedure before calling the LostSale function. The same procedure can be carried out to the resultant measures to spread or aggregate them to the designated intersections.

5 Because of the limitation that the same measure can not appear on both left-hand-side and right-hand-side simultaneously, the implementation of the CLEAR filter requires the user to provide a LSOVER_REF measure, which is a duplication of the previously calculated LSOVER measure, when the user try to retain the results on certain time series but clear the others by providing a mask measure (TSMASK_DENSE). The LSOVER_REF is not required when the results for all the time series need to be cleared.

6 The LSTODAY measure is used to specify the end date for the filter processing. It only accepts the index number for the end date along the calendar dimension as valid input. If it is desired that the string position name to be used for the end date specification, the available RPAS time dimension translation function "index" can be used to do the name-index conversion before calling the LostSale function.

7 The LSTODAY input parameter is designed to be a measure rather than a constant to provide more flexibility. Current implementation only allows one global LSTODAY index value to be used in processing all the time series. To specify the end date, user just needs to populate its value for the first time series and this index will be applied to all the other time series.

8 The index value in LSTODAY measure started from 1 rather than 0.

9 FLP_FIRST and FLP_LAST are the resultant measures to be used for the First-Last-Populated Location calculation. They do not have the calendar dimension and each of their cell values represents the indices for the first and last populated locations along the calendar dimension from the first time series up to the current time series, respectively.

10 TSMASK_DENSE is a boolean input measure without calendar dimension to specify which time series is going to be processed and which is not. For filtering methods other than the CLEAR method, the true value means if the popcount for the current time series is larger than the hard-coded threshold value, it will be processed. Otherwise, it will not be processed. The false value means the current time series will not be processed. If the TSMASK_DENSE measure is not specified, then all the time series will be processed and the internal hard-coded threshold value will not be considered. For the CLEAR filtering method, the true value means that the previously calculated results for the current time series will be cleared and the false value means the results will be retained. If the TSMASK_DENSE measure is not specified, all the results will be cleared.

11 For all the input measures that do not have the calendar dimension, such as UP_ADJ_RATIO and DELTA, user can use a constant as input. In this case, the constant value will be applied to all the time series.

Configuration Restriction

- '_' (underscore) may not be used in any measure names and rules. The exception to this are measures and rules that are to be expanded using the RDF or Curve solution's classification scheme. The classifications in question are as follows:

  _F: Expand measures and rules across final levels

  _S: Expand measures and rules across source levels

  _B: Expand measures and rules across birth dates

Chapter 7 – Configuring the Markdown Optimizer Function

Markdown Optimization provides merchants with the optimal inventory position and optimal markdown strategy. It helps retailers buy the right amount of inventory and mark it down at the right time with the right discounts to maximize profit. The solution also allows merchants to perform what-if simulations to evaluate the effects of deviating from the optimal strategy if it doesn't fit their markdown budget and business limitations.

The following section contains the specifications and syntax for configuring the MOFunction library in order to support a Markdown Optimization workbook. Note that the syntax is slightly different than the standard RPAS functions and procedures described in the RPAS 11.0.4 Rule Functions Reference Guide.

Syntax Legend

[…] all options listed in brackets are optional

{…|…} options listed in "{}" with "|" seperaters are mutually exclusive (either/or)

{…,…} options listed in "{}" with "," seperators are a complete set.

Bold – labels

Italics – indicates a temporary placeholder for a constant or a measure (will end with 'meas')

Italics/meas – indicates that the placeholder can be either a constant or a measure.

BoldItailics – indicates a numeric placeholder for the dynamic portion of a label. Usually a number from 1 to N

Normal – normal text that has to be there.

Underlined And Large font – Used to identify the function name.

Syntax

PROFITLB:*@PROFITLBMEAS*, **SOLUTIONLB**:*@SOLUTIONLBMEAS*,
OPTSTRATEGYLB:*@OPTSTRATEGYLBMEAS*,
OPTMDSTRATEGYLB:*@OPTMDSTRATEGYLBMEAS*,[
OPTPRICELB:*@OPTPRICELBMEAS*,]
OPTMDPRICELB:*@OPTMDPRICELBMEAS*, **PROJINVLB**:*@PROJINVLBMEAS*,
ADJFRCSTLB:*@ADJFRCSTLBMEAS*, **EOSNUMLB**:*@EOSNUMLBMEAS*,
OPTINVLB:*@OPTINVLBMEAS*=mdoptimize([DEMANDLB:*@DEMANDLBMEAS*,]
[**INVENTORYLB**:*@INVENTORYLBMEAS*,] **RPRICELB**:*@RPRICELBMEAS*,
[**COSTLB**:*@COSTLBMEAS*,][**MASKLB**:*@MASKLBMEAS*,]
[**FLDTLB**:*@FLDTLBMEAS*,][**TOTLIFELB**:*@TOTLIFELBMEAS*,]
[**PRICEPOINTSLB**:*@PRICEPOINTSLBMEAS*,][**PRICELIFTSLB**:*@PRICELIFTSLBMEAS*,]**WHATIFSTRATEGYLB**:*@WHATIFSTRATEGYLBMEAS*,
WHENTOOPTIMIZELB:*@WHENTOOPTIMIZELBMEAS*,
RECEIPTLB:*@RECEIPTLBMEAS*, **ALLOWWEEKSLB**:*@ALLOWWEEKSLBMEAS*,
REGLIFELB:*@REGLIFELBMEAS*, **PSTOCKLB**:*@PSTOCKLBMEAS*,
EOSVALLB:*@EOSVALLBMEAS*, **PCCOSTLB**:*@PCCOSTLBMEAS*,
PCCOSTPERITEMLB:*@PCCOSTPERITEMLBMEAS*,
MAXMDNUMLB:*@MAXMDNUMLBMEAS*, **MINGAPLB**:*@MINGAPLBMEAS*,
MINMDPCTLB:*@MINMDPCTLBMEAS*, **MAXMDPCTLB**:*@MAXMDPCTLBMEAS*,
MARKUPLB:*@MARKUPLBMEAS*, **FFMKDLB**:*@FFMKDLBMEAS*,
MINFMDPCTLB:*@MINFMDPCTLBMEAS*, **MAXFMDPCTLB**:*@MAXFMDPCTLBMEAS*,
MDPCTPTSLB:*@MDPCTPTSLBMEAS*, **MAXINVLB**:*@MAXINVLBMEAS*)

Example Usage

PROFITLB:moprofit, SOLUTIONLB:mosolution, OPTSTRATEGYLB:mooptstrategy, OPTMDSTRATEGYLB:mooptmdstrategy, OPTPRICELB:mooptprice, OPTMDPRICELB:mooptmdprice, PROJINVLB:moprojinv, ADJFRCSTLB:moadjfrcst, EOSNUMLB:moeosnum,
OPTINVLB:mooptinv=mdoptimize(DEMANDLB:modemand, INVENTORYLB:moinventory, RPRICELB:morprice, COSTLB:mocost, MASKLB:momask, FLDTLB:mofldtind, TOTLIFELB:mototlife, PRICEPOINTSLB:mopricepoints, PRICELIFTSLB:mopricelifts, WHATIFSTRATEGYLB:mowhatif, WHENTOOPTIMIZELB:mowhentoopt, RECEIPTLB:moreceipt, ALLOWWEEKSLB:moallowweeks, REGLIFELB:moreglife, PSTOCKLB:mopstock, EOSVALLB:moeosval, PCCOSTLB:mopccost, PCCOSTPERITEMLB:mopcperitem, MAXMDNUMLB:momaxmdnum, MINGAPLB:momingap, MINMDPCTLB:mominmdpct, MAXMDPCTLB:momaxmdpct, MARKUPLB:momarkup, FFMKDLB:moffmkd, MINFMDPCTLB:mominfmdpct, MAXFMDPCTLB:momaxfmdpct, MDPCTPTSLB:momdpctpts, MAXINVLB:momaxinv)

Configuration Parameters and Rules

Input Parameters

Parameter	Description	Intersection	Usage Comment
DEMANDLB	Regular price demand measure, real type. This can come from Legacy or can be calculated in-season using the MO Batch Bayesian Adjustment	Prod/loc/clnd	An optional input. Input value must be >0.
INVENTORYLB	BOP inventory at beginning of a season, real type. When inventory quantity is set to –1, inventory optimized is invoked.	Prod/loc/	An optional input. Input value must be >0
OPTPRICELB	Item regular price, real type	Prod/loc/	An optional input. Input value must be >0. Must be > or = to cost
COSTLB	Item cost, real type.	Prod/loc/	An optional input. Input value must be >0
MASKLB	An optimization Boolean measure indicating which prod/loc combination need markdown optimization	Prod/loc/	An optional input. Boolean must be checked on to see the optimized solution.

Parameter	Description	Intersection	Usage Comment
FLDTLB	Flow date index, the item start date index along the dimension in calendar hierarchy, realtype	Prod/loc or Prod	An optional input. Workbook must include 1 time period prior to the Start Date. As well, you must include all time periods for optimization in your workbook.
TOTLIFELB	The total number of periods for an item to be on sale. Real type	Prod/loc , prod	An optional input. Input value must be >0
PRICEPOINTSLB	Available Price Points to an item, real type	Prod/loc/clnd or Prod/clnd	An optional input. Calculated by ‘copyseries’ rule below. Input value must be >0
PRICELIFTSLB	The ratio lift over regular price demand, real type	on the same intersection as Price points	An optional input. Calculated by ‘copyseries’ rule below. Input value must be >0.
WHATIFSTRATEGYLB	A user-defined or loaded weekly Pricing strategy for an item if ‘Perform Optimize’ is false for a week. Defaults to 0	Prod/loc/clnd, prod/clnd	Interacts directly with ‘Perform Optimize’ measure

Parameter	Description	Intersection	Usage Comment
WHENTOOPTIMIZELB	Defaulted to true. A Boolean measure indicating which week optimization is to be performed on. It is set to false in two cases: 1) When an item is already in-season, the actualized week should be turned off and the actualized pricing stategy is inputed in the whatif strategy. The actualized demand should be calculated in two steps: a) remove inventory from actual sales and b) remove price effects from actual sales 2) For a certain week, user specified strategy needs to be used.	Prod/loc/clnd, prod/clnd	Interacts directly with 'What-if Strategy' measure.
RECEIPTLB	Defaulted to 0. A real measure indicating inventory receipt. The received units are divided evenly into two parts. One part is counted as the BOP inventory of the current week. One part is counted as BOP inventory of the next week	Prod/loc/clnd, prod/clnd	Can be a positive or negative value.
ALLOWWEEKSLB	A Boolean measure that indicates which week is allowed for markdown. Defaults to true.	Prod/loc/clnd	
REGLIFELB	A real measure that indicates how many weeks an item must be kept on regular price. Defaulted to 0	Prod/loc/ or prod	Input value must be > or = to 0
PSTOCKLB	A real measure that contains the weekly presentation stock number. Defaulted to 1. Can come from Legacy or calculated using the MO Pre-season or In-season batch processes.	Prod/loc/week	Input value must be > 1 to get an optimized solution
EOSVALLB	A real measure contains the end of season value for an item. Defaulted to 0	Prod/loc/ or prod	Can be positive or negative value
PCCOSTLB	A real measure contains the fixed price change cost per markdown. Defaulted to 0.	Prod/loc or Prod	Can be positive or negative value

Parameter	Description	Intersection	Usage Comment
PCCOSTPERITEMLB	A real measure contains the price change cost per inventory unit. Defaulted to 0	Prod/loc, prod	Can be positive or negative value
MAXMDNUMLB	Maximum number of markdowns, real type. Defaulted to 9999	Prod/loc or Prod	
MINGAPLB	Minimum length that is allowed for a markdown. Defaulted to 1	Prod/loc or Prod	Input value must be > 1
MINMDPCTLB	Minimum markdown percentage between markdowns. Defaulted to 0	Prod/loc, prod	Input value must be at or between 0 and 1.
MAXMDPCTLB	Maximum markdown percentage, Defaulted to 1	Prod/loc, prod	Input value must be at or between 0 and 1.
MAXINVLB	The maximum inventory to be used in inventory optimization. Defaulted to 200,000, real type	Prod/loc, prod	Input value must be > 1
MARKUPLB	A Boolean measure indicating whether mark up is allowed. Defaulted to false.	Prod/loc, prod	
FFMKDLB	A Boolean measure indicate whether a markdown is to be forced at the first allowed week. Defaulted to false	Prod/loc, prod, loc or scalar	
MINFMDPCTLB	Minimum first markdown percentage. Defaulted to 0, real type	Prod/loc, prod, loc or scalar	Input value must be at or between 0 and 1. % Min 1st Markdown must be < or = to % Max 1st Markdown

Parameter	Description	Intersection	Usage Comment
MAXFMDPCTLB	Maximum first markdown percentage. Defaulted to 1, real type	Prod/loc, prod, loc or scalar	Input value must be at or between 0 and 1. % Max 1st Markdown must be > or = to % Min 1st Markdown
MDPCTPTSLB	A list of markdown percentage points available for an item. This list intersects with price points to provide a list of available prices for an item	Prod/loc/clnd, prod/clnd	Calculated by 'copyseries' rule below. Must be > or = to 0

Input Measures to support optimization rules

The following measures names and labels are configurable, however these are included to provide examples of calc rules that may be needed to support the workbook template.

Measure Name	Measure Label	Description	Intersection	Usage Comment
morltvpricepts	Rel. Price Points	Available Price Points to an item, real type, along Relative Week. This allows for the price table to be applied to any markdown time period.	Prod/loc/clnd orProd/clnd	An optional input. Input value must be > 0
morltvpricelifts	Rel. Price Lifts	Demand lift corresponding to price changes, real type, along Relative Week. This allows for the price table to be applied to any markdown time period.	on the same intersection as Price points	An optional input. Input value must be > 0

Measure Name	Measure Label	Description	Intersection	Usage Comment
morltvmdpctpts	Rel. Markdown Percentage Points	A list of markdown percentage points available for an item. This list intersects with price points to provide a list of available prices for an item, along Relative Week. This allows for the price table to be applied to any markdown time period.	Prod/loc/clnd, prod/clnd	Input value must be at or > 0
mope	Price Elasticity	Price Elasticity measures the responsiveness of demand to a change in price. It is a unit less value, used in the calc rule to support the calculation of the relative price points to relative price lift. This can come from Legacy or can be calculated using the MO Pre-season or In-season Batch processes.	Prod/loc/clnd or Prod/clnd	Must be a positive value. We suggest the range to be set to 0.5:5

Calc Rules to support the configuration of input parameters:

1 calcrltvplift: Rule to support the calculation of the relative price points to relative price lift

- morltvpricelift =pow(morltvpricepts /morprice , -mope)

2 copyrltvpts: Rule to support translation of Price Points from Relative Week to Week using the copyseries function

- DESTINATION:mopricepoints <- copyseries(SOURCE:morltvpricepts)

3 copyrltvlift:: Rule to support translation of Price Lifts from Relative Week to Week using the copyseries function

- DESTINATION:mopricelifts <- copyseries(SOURCE:morltvpricelift)

4 copyrltvpct: Rule to support translation of MD Percentage Points from Relative Week to Week using the copyseries function

- DESTINATION:momdpctpts <- copyseries(SOURCE:morltvmdpctpts)

5 mofldtindex: Rule to support selection of Flow Date using the RPAS Calendar Interface:

- mofldtind = index([CLND].[week],mofldt)

Output Parameters

Measure Name	Description
PROFITLB	A real type measure contains calculated profit
SOLUTIONLB	A string type measure containing data validation information and results information
EOSNUMLB	A real type measure contains information about how many units is left at the end of season
OPTINVLB	A real type measure contains the optimized total inventory units quantity for the entire markdown period.
OPTSTRATEGYLB	A real type measure contains optimized weekly pricing strategy. (as markdown percentage)
OPTPRICELB	A real type measure contains optimized weekly price
OPTMDSTRATEGYLB	A real type measure contains optimized weekly mark down pricing strategy (as markdown percentage) A strategy will be returned only for time periods (weeks) that are not indicated to be at regular price.
OPTMDPRICELB	A real type measure contains optimized weekly markdown prices. A price will be returned if the markdown is allowed within this time period (week). A price will be returned only for time periods (weeks) that are not indicated to be at regular price.
ADJFRCSTLB	A real type measure contains forecasted sales following the optimized strategy and using optimized inventory or input inventory

Measure Name	Description
PROJINVLB	A real type measure contains projected inventory following the optimized strategy and using the optimized total inventory or input inventory. This is per time period (per week)

Configuration Restriction

- '_' (underscore) may not be used in any measure names and rules. The exception to this are measures and rules that are to be expanded using the RDF solution's classification scheme. The classifications in question are as follows:

  _F: Expand measures and rules across final levels

  _S: Expand measures and rules across source levels

  _B: Expand measures and rules across birth dates