Retek® Predictive Application Server™ 11.2

Configuration Tools User Guide

Customer Support

Customer Support hours

Customer Support is available 7x24x365 via email, 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
Hong Kong	800 96 4262
Korea	00 308 13 1342
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

Overview

The Retek Predictive Application Server (RPAS) configuration tools provide a flexible means to configure and build RPAS-based applications with customer-specific business parameters. The configuration tools provide a streamlined, user-friendly interface to utilize RPAS functionality. Once a configuration has been created, an installer script is used to build an RPAS domain.

The configuration tools consist of an integrated set of task-specific configuration aids that are used to configure a solution template, or to modify an existing solution template.

RPAS functionality is exposed to the configuration tools via Application Programming Interfaces (APIs).

A configuration is typically created and maintained by an application administrator or solution expert. Details of the configuration are stored locally on the administrator's PC, on the network, or in a source control system. Once the configuration is complete, the administrator uses the configuration to create a new domain or update an existing domain.

Users of the configured solution will access the RPAS domain by using the RPAS client that is installed on their machines. The domain that they access will represent the business process and environment that was configured in the solution by the Configuration Administrator.

Once the domain is created, some administrative RPAS processes (such as hierarchy maintenance and user administration) are accomplished by an RPAS Administrator who uses the utilities on the server. For more information on RPAS Administration, refer to the RPAS Administrator's Guide for your RPAS version.

Business Process

Configuration tools
Set up system properties and source control
Create a configuration
Create a project
Create solutions
Configure hierarchies and dimensions
Configure measure components and measures
Configure rules sets, rule groups, and rules
Configure workbooks, workbook tabs, and worksheets
Configure wizards
Define interfaces used to import data
Build an RPAS domain

Audience

The information presented here describes how to use the RPAS configuration tools and how to build an RPAS domain.

It is designed for anyone who is responsible for creating or maintaining RPAS configurations. You should know basic RPAS concepts; including hierarchies, dimensions, measures, rules and expressions, aggregation and spreading, domains, workbooks and worksheets, and how the calculation engine works.

Sample configurations

The examples illustrated utilize a Sample Configuration, which is delivered with the RPAS platform and can be installed along with the RPAS and Configuration Tools. The Sample Configuration may not match every illustration because RPAS and Tools software version might vary between users. The illustrations are meant to provide a context to the reader. Detailed documentation of the Sample Configuration is provided in the RPAS Installation Guide.

Know the Configuration Manager

As you create new configurations; you name the configuration, the project, and the solution in the configuration. You can drill down through the configuration to work in specific areas. The icons in Configuration Manager help you know where you are working, and navigation statements use these icons to help you follow procedures. If an icon has been modified, it contains a modification flag and it must be saved. See the hierarchies below.

Configuration Manager

- Sample_Config
  - Hierarchies
  - Data Interface
  - Styles
  - Solution_1
    - Measures
    - Rules
    - Workbooks
    - Wizards

Icon Name	Window displayed in the workspace
Project	None
Hierarchies	Hierarchy Definition window
Data Interface	Data Interface Manager window
Styles	Style Configuration Window
Solution	None
Measures	Measure Definition window
Rules	Rule Definition window
Workbooks	Workbook Designer window
Wizards	Wizard Designer window

Navigating the configuration tools

Once you log into the Configuration Tools, you have access to the RPAS Configuration Tools window.
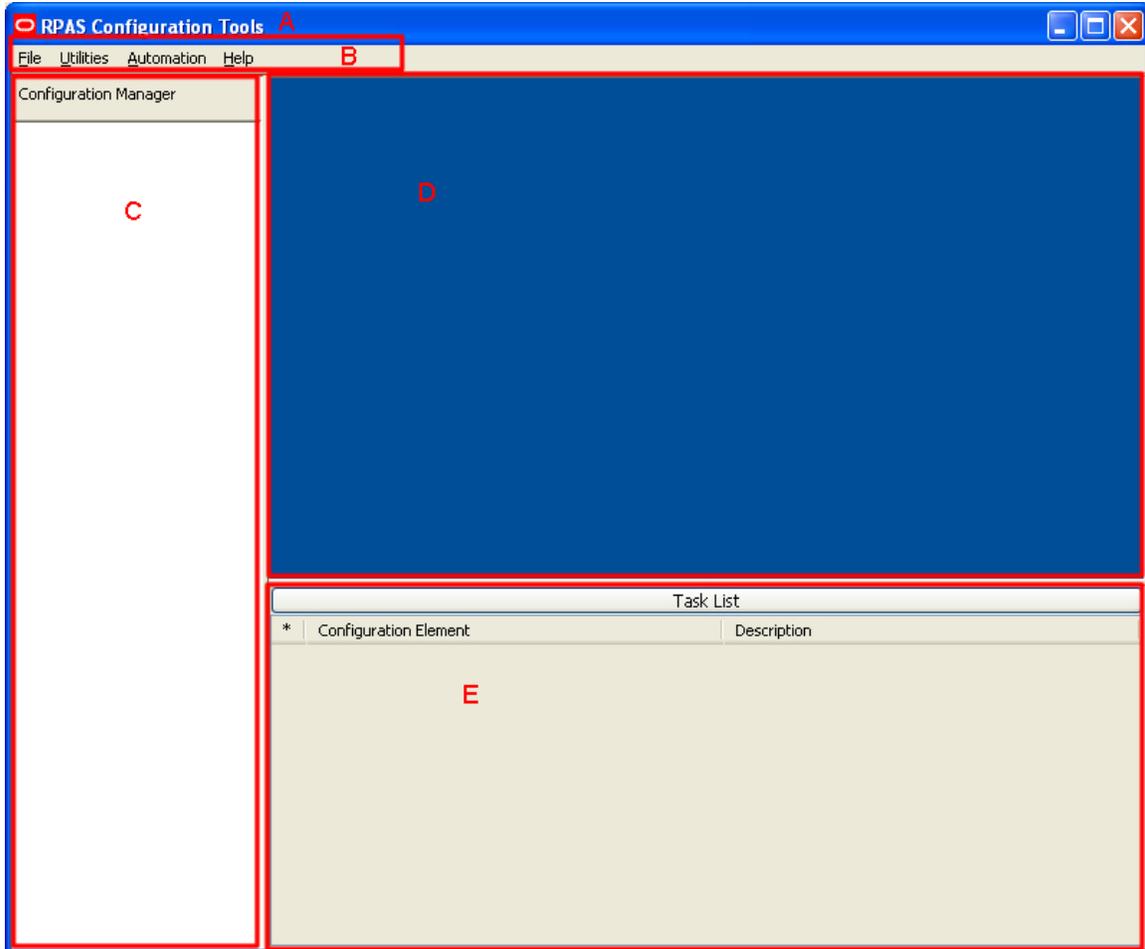
The primary elements in the application window are as follows:

Element	Purpose
Title Bar (A)	• Located at the top of the application window • Displays the product name • The three buttons at the far right on the title bar allow you to minimize, restore, maximize, and close the application window
Menu Bar (B)	• Contains the menus that are used in the configuration tools • Each menu contains a set of commands that allow you to operate the configuration tool
Configuration Manager (C)	• Displays information about configurations, projects, and solutions that you are currently using • Configuration information is not displayed until you open a configuration.
Workspace (D)	• As you select the different elements of a configuration in the Configuration Manager, the related windows are displayed in the workspace.
Task List (E)	• Displays errors within the configuration

Chapter 2 – Configuration manager

Overview

The configuration manager is the starting point for creating a new configuration or for opening an existing configuration. It provides a high-level view of all the components that are necessary to configure an RPAS application, and it is used to navigate to the various tools that are used to configure those components.

The configuration manager is the core of the RPAS configuration tools, and it provides an overall view of the configuration components. Each configuration contain one project and one or more solutions.

Projects

Each project represents an RPAS domain. A project contains one or more solutions.

Hierarchy

For each project, you create hierarchies and dimensions within those hierarchies. Hierarchies are the structures used by an organization to describe the relationships that exist between the dimensions. The following hierarchies are automatically created:

- Calendar

- Product

- Location

- ADMU

ADMU is not a user defined hierarchy, so no one can create or modify it. ADMU is built by RPAS, and tools make it available for use in configurations. ADMU is the 'user' hierarchy, and it exists to allow a measure to have a base intersection of user.

You create and define dimensions for these hierarchies, and for any additional hierarchies that you create.

 Note: There are language issues for the automatically generated hierarchies (CLND, PROD, LOC, and ADMU). These names cannot be changed, but the default user labels (Calendar, Product, and Location) can be changed with the exception of ADMU. If a user wants a user label other than English or a different terminology (for instance, Service instead of Product), he/she should change the user label. A user cannot delete a hierarchy (for instance, PROD) and replace it with one called SVCE.

Data Interface

The data interface tool is used to define what the files (and some parameters) look like in order to tell RPAS how to deal with various things like data that is sourced below its base intersection. It sets measure attributes that are referenced when loading measure data into the domain. The information entered into the data interface will be referenced when the 'loadmeas' utility is used to load data for a measure.

Solutions

A solution corresponds to an application configuration (for example, Financial Planning or Item Planning). For each solution, you configure:

- measures / expressions
- rules / rule groups
- workbooks / worksheets
- wizards (optional)

Measures

Measures (multidimensional variables) are any item of data that can be represented on a grid in a worksheet. Measures are the data points that are used in the customer's business process.

Rules

Rules are collections of expressions (the basis of all calculations) that describe the relationships between measures. They are evaluated by the calculation engine during a calculation. Rules can consist of multiple expressions as the following example represents:

Expression 1: ReceiptUnits = ReceiptsValue / ReceiptsPrice

Expression 2: ReceiptsValue = ReceiptUnits * ReceiptsPrice

Expression 3: ReceiptsPrice = ReceiptUnits / ReceiptsValue

The collection of expressions represents a rule. These three expressions state the relationship between ReceiptUnits, ReceiptsValue, and ReceiptsPrice. Each expression solves for a different measure.

A rule group is a collection of rules that are treated as a unit by the calculation engine. The rules in the rule group must be considered together to satisfy the calculation requirements for a specific business process. The sequence of rules in a rule group determines the calculation sequence unless the sequence is forced.

Rule sets are a collection of rule groups that are used for organizational purposes by the tools.

Workbooks

A workbook is the multidimensional framework that is used to perform specific business functions, such as creating a merchandise plan and reviewing availability data. Workbooks are easily viewed and manipulated.

A workbook can contain any number of multidimensional spreadsheets, called worksheets, to present data. Measures and rules are used to generate the measure data. All of these components work together to facilitate the viewing and analysis of business functions. The configuration tools allow you to configure workbook templates incorporating these various components.

Wizards

A wizard is a feature that guides you through the process of building a new workbook. A wizard displays successive dialogs that require you to answer a sequence of questions or enter information regarding the content of the workbook. Responses to these questions are used to automatically format and populate the workbook. The configuration tools allow you to create custom wizard formats.

The main purpose of the wizard is to allow the end user to make choices regarding the shape of the workbook. For example, the first wizard might ask you to select the SKUs to include in the workbook. The second might ask you to select the stores to include in the workbook, and the third wizard might ask you to select the dates to include in the workbook. At the end of the series of wizards, a workbook will be created that has data for the SKUs, stores, and dates that the user selected.

How RPAS uses the configurations

The RPAS calculation engine

The RPAS calculation engine is an engine that is built to support On-Line Analytical Processing (OLAP) type calculations against a multi-dimensional model. The engine is very powerful and flexible.

In the OLAP model, individual pieces of data, called cells, apply to a single position in one or more hierarchies or dimensions. Cells typically include:

- A measures dimension
- A calendar or time hierarchy
- Other hierarchies such as products and locations.

The measures dimension is fundamentally different to the other hierarchies, since measures represent the fundamental events or measurements that are being recorded. The positions in the other hierarchies provide a context for the measurement: where, when, what, and so on. Measures relate to one another through rules and expressions: positions in all the other hierarchies relate to each other through hierarchical relationships.

Aggregation and spreading

The RPAS calculation engine is designed to be robust and extensible, but in complete control of the calculation process. It enforces integrity of the data by ensuring that, all known relationships between cells are always enforced. Much of the logic of the processing of rules and rule groups depends on this basic principal. RPAS supports two different forms of relationships between cells:

1. Hierarchical relationships that require aggregation and spreading
2. Measure relationships that require rules and expressions.

Aggregation and spreading are basic capabilities of the engine that require no coding by the implementer, other than the selection of aggregation and spreading types to use for a measure. Hierarchical relationships, such as weeks rolling up to months, or stores rolling up to regions, require the aggregation of data values from lower levels in a hierarchy to higher levels, using a variety of methods, as appropriate to the measure.

To enable such data to be manipulated at higher levels, RPAS supports spreading the changes, also using a variety of methods.

The inherent relationships between measures can be modeled through a rich rule and expression syntax. Most of the effort in configuring an application model is in modeling these relationships.

RPAS functions

RPAS Functions are mechanisms for performing operations within an expression that are controlled and executed by the calculation engine.

- There is always a single measure on the left hand side of a normal expression; that is, a function always has only one measure output.

- The calculation engine controls and executes the evaluation of a function.

- Functions may be used in long expressions with other functions and keywords.

- The data that can be referenced is limited to the scope of the workbook.

See RPAS Rules Function Reference Guide for details about RPAS functions.

Right-click menus in the configuration manager

Procedures

Rename a project

1. Select the project that you want to rename.

2. Right click on the project.

3. Select Rename. The Rename window opens.

4. In the Project field, enter new name for the project.

5. Click **OK** to save any changes and close the window.

Moving a project

1. Select the project that you want to move.

2. Right click on the project.

3. Select Move. The Move window opens.

4. In the Project field, enter the name for the project after it has been moved.

5. In the Workspace field, select the new project to which you want to move the project.

6. Click **OK** to save any changes and close the window.

Copy a solution

1. Select the solution that you want to copy.

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

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

4. Select the project in which to create the copied solution.

5. Click **Finish** to save the copied solution in the specified project.

Move a solution

1. Select the solution that you want to move.

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

3. Type the name of the solution in the text box.

📖   **Note:** This is the name the solution is called after it has been moved.

4. Select the project to which you want to move the solution.

5. Click **Finish** to move the solution to the specified project.

Chapter 3 – Projects

Work with projects

Overview

A project is used to configure the structure of a domain. The hierarchies and dimensions are defined within a project and are available for use within all solutions in a given project.

    Note: A solution can use a subset of the hierarchies and dimensions defined within the project. Within a project, you can additionally define certain properties that describe how measures will be loaded into the domain, which are part of the Data Interface Tool.

Procedure

Create a project

⇒ **Navigate:** From the File Menu, select New > Project. The New Window opens.

1. In the Configuration field, enter the name of the configuration/project.

2. Select the Use Defaults checkbox to use the default paths that are currently active; to specify an alternate location uncheck Use Defaults and select the path.

3. Select the language in which the configuration will be created.

4. Select the options for Global Domain, Multilanguage, and Hypersparse as necessary. Refer to Chapter 5 – System preferences for additional information about these options.

  📖 **Note:** The Global Domain and and Multi-Language settings must be defined before the domain is built. A domain can be converted to or from a hypersparse domain after the domain is built. Changes to the Global Domain and MultiLanguage properties are ignored when modifying the configuration of an existing domain.

5. Click **OK** to save any changes and close the window.

Move a project to another project

You can move or copy a project from one configuration to another. When you move a project, the project is removed from the first configuration and created in the second configuration.

⇒ **Navigate:** In the Configuration Manager, right-click the project that you want to move, and select Move. The Move dialog box opens.

1. In the project field, enter the name of the project that you want to move.

2. In the Workspace field, use the drop-down box to select the new project that you want to move the project to.

3. Click **OK**. The project has now been moved to the new project.

Copy a project to another project

⇒ **Navigate:** In the Configuration Manager, right-click the project that you want to copy, and select Copy. The Clone dialog box appears.

1. In the project field, enter the name of the project that you want to copy.

2. In the Workspace field, use the drop-down box to select the new project that you want to copy the project to.

3. Click **OK**. The project has now been copied to the new project.

Hierarchies

Overview

A hierarchy is a top-to-bottom set up of parent-child relationships between groups of things of the same type, and it provides a means to define relationships between dimensions (aggregates, roll ups, and alternate roll ups). The default hierarchies in the configuration tool are PROD (Product), LOC (Location), and CLND (Calendar). These hierarchies are required by RPAS-based solutions and cannot be removed, but additional hierarchies can be added to support the required business process.

Hierarchies define the path of data aggregation and spreading. In a workbook, the user can view data at any required level of detail by drilling down or rolling up through dimensions in the hierarchy.

The hierarchy definition tool:

- Allows users to define and construct hierarchies, dimensions within those hierarchies, and relationships between those dimensions
- Provides a visual representation of a hierarchy and its dimensions
- Provides a means to define the hierarchy data load file
- Allows existing hierarchies/dimensions to be reused in a new configuration

The following diagram represents a common structure of an organization's product hierarchy. In this example, the SKU dimension contains an alternate roll-up to the Supplier dimension. Each position in the SKU dimension will have a parent position in the Style dimension and one in the Supplier dimension.
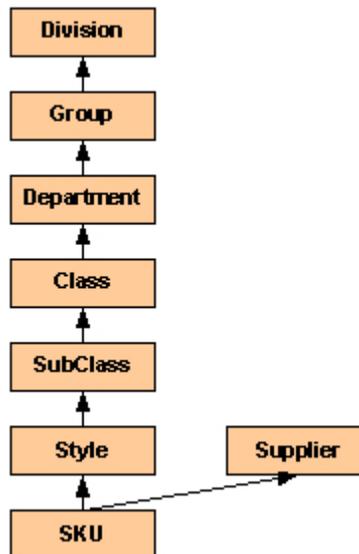
Product hierarchy and dimensions

Work with hierarchies

Overview

When a project is created, the Calendar (CLND), Product (PROD), Location (LOC), and User (ADMU) hierarchies are automatically created by default. These hierarchies are required as they are common to all RPAS-based solutions. Additional hierarchies and dimensions can be created as well.

The determination of which hierarchies to use in a solution is driven by the base intersection defined for the measures of a given solution. For example, a project might contain five hierarchies, but the measures in one of the solutions might use only four of those hierarchies in the base intersections of the measures, so even though a fifth hierarchy exists it is not actually used in the solution.

Procedures

Create a new hierarchy

⇒ **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > ☐Hierarchies. The Hierarchy Definition window opens in the workspace.
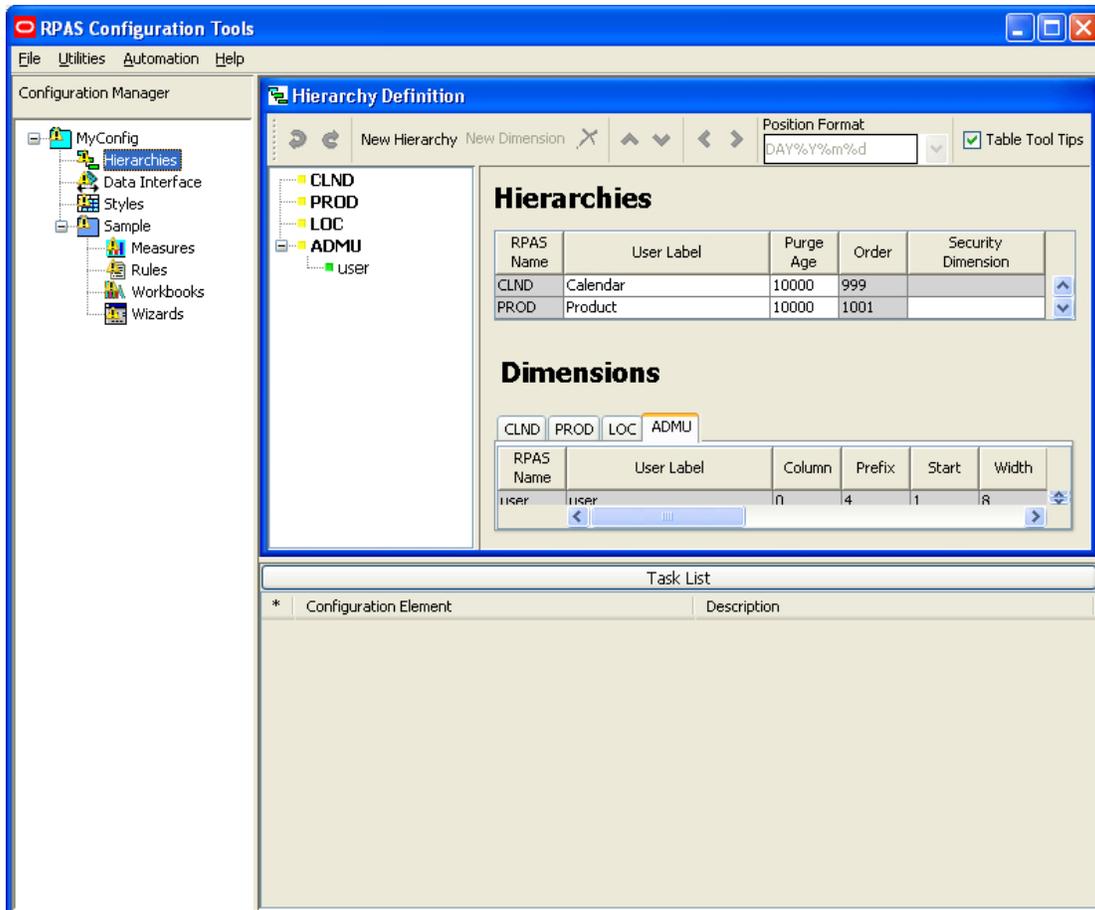
1. Click New Hierarchy.

2. To change the name of the newly created hierarchy in the left pane of the Hierarchy Definition window:

    a. Either right-click on the hierarchy name, and select Rename

    OR

    b. Double click the hierarchy name to enable it.

    c. Enter the new name.

    📖   **Note:** The RPAS name can only be up to four characters long.

3. Press Enter or click outside the hierarchy name.

Specify hierarchy information

1. In the left-hand pane, select the name of the hierarchy for which you want to specify information.

2. In the Hierarchies area, enter the following:

    ▪ **RPAS Name** – The RPAS internal name of the hierarchy.

    📖   **Note:** The RPAS name can only be up to four characters long. You cannot change the name of the default hierarchies.

    ▪ **User Label** – The hierarchy label that is displayed to RPAS users.

    ▪ **Purge Age** – The purge age determines when a position and its corresponding measure data is removed from a domain. Specifically it represents the number of days before the data is purged from the last time the position was included in the hierarchy input file that is loaded with the loadhier utility (most commonly on a nightly or weekly basis). Setting this value to zero means that a position and all of its data will be immediately purged if it is not included in the hierarchy file.

    📖   **Note:** The value set in this field serves as the default value to use when loading the corresponding hierarchy. This value can be overwritten by one of the arguments of the loadhier utility each time the utility is called.

    Example 1: A purge age of "0" will purge positions the first night they are not in the input file.

    Example 2: A purge age of "1000" will purge the positions the 1000th night after they are last seen on the input file.

    ▪ **Order:** Hierarchy order determines the ordering of dimension fields in the physical storage of data in the RPAS domain. This ordering is the traversal order of data for calculations, which relates to how RPAS iterates over data when performing calculations. Data in the domain is stored in multi-dimensional arrays, with each dimension belonging to a different hierarchy.

    For performance reasons the Calendar hierarchy (and therefore all of its dimensions) is always the "innermost" dimension and defaults to an uneditable number of 999. The ordering of any hierarchy can be changed except Calendar (CLND). The lower the order number the nearer the hierarchy is to the innermost.

Consider the following example for the Calendar, Product, and Location hierarchies:

- CLND order = 999

- PROD order = 1001

- LOC order = 1002

Two products: P1 and P2

Two locations: L1 and L2

Two calendar periods: C1 and C2

The sequence of physically storing and iterating over the data with calendar as the innermost dimension and location as the outermost dimension would be:

- L1/P1/C1

- L1/P1/C2

- L1/P2/C1

- L1/P2/C2

- L2/P1/C1

- L2/P1/C2

- L2/P2/C1

- L2/P2/C2

With Calendar being the innermost dimension, data is first processed for all positions in the Calendar hierarchy and for the first position of the other hierarchies. In this example, data would be processed for all calendar positions for the first product and first location, then all calendar positions for the second product and first location, and so on.

It is recommended that customers order their hierarchies with Calendar as the innermost dimension (required), followed by other hierarchies in order of importance/traversal – most commonly Product, Location, then other hierarchies (if applicable). Note that many RPAS-based solutions (such as Advanced Inventory Planning and Demand Forecasting) have additional hierarchies that are in a pre-defined order that cannot be changed. Oracle Services can provide additional information or advice if there are compelling reasons or concerns to consider using a different hierarchy order.

The Order column also indicates the order in which the hierarchy information is expected in the file used for measure data loading purposes.

Note that the values "1000" or "1020" are not used as a hierarchy order as they are used internally by RPAS.

- **Security Dimension** – Selecting a Security Dimension for a hierarchy enables position-level security in the domain for the corresponding hierarchy. Any dimension along any hierarchy except the Calendar hierarchy is valid. For example, if the security dimension for the product hierarchy is set to "Dept" (Department Level Security); within the domain, access to departments can be granted/denied by the administrator for individual users, user groups, or all users. If position-level security is to be enabled in RPAS, select the security level. Refer to the RPAS 11.2 Administration Guide for additional information about position-level security.

3. Click the up [up button] button and down [down button] button to move the hierarchy to the desired location, which changes the Order hierarchy property. You can also reorder the hierarchy through drag and drop in the hierarchy tree pane. Whichever technique you use, the order numbers are automatically changed/generated.

    📖   **Note:** The [button] and [button] buttons are used for defining the relationships between dimensions.

    📖   **Note:** The name of a hierarchy is un-editable if it is shaded gray; however, you can change other attributes (for instance, the label). CLND is always the innermost dimension, and ADMU is always the outermost dimension. The order of the other hierarchies (PROD, LOC, and others that you created) can be changed.

Delete a hierarchy

⇒ **Navigate:** In the Configuration Manager, select [icon] Project > [icon] Solution > [icon] Hierarchies. The Hierarchy Definition window opens in the workspace.

1. In the Hierarchies area, select the hierarchy that you want to delete.

2. Click the delete [delete button] button. The hierarchy is removed.

    📖   **Note:** You cannot delete the CLND, PROD, LOC, or ADMU hierarchies.

Edit a hierarchy

⇒ **Navigate:** In the Configuration Manager, select [icon] Project > [icon] Solution > [icon] Hierarchies. The Hierarchy Definition window opens in the workspace.

1. In the Hierarchies area, select the hierarchy that you want to edit.

2. Click on the field that you want to updated.

3. Enter the new information.

4. To change the order of the hierarchy, click the up [up button] button or down [down button] button as necessary.

Work with dimensions

Overview

Dimensions are the components within a hierarchy that define the structure and rollup within a hierarchy. For example, the dimensions for a calendar hierarchy can be day, week, month, and year; or they can be accounting periods.

Procedures

Create a dimension

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤Hierarchies. The Hierarchy Definition window opens in the workspace.
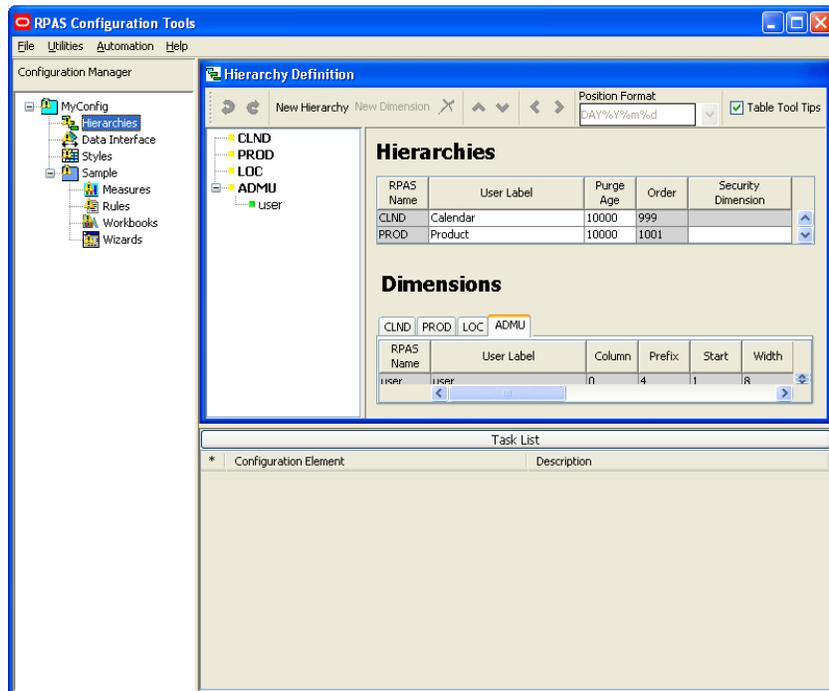
1. Highlight the hierarchy or dimension under which to create the new dimension.

📖 **Note:** When referring to top and bottom level dimensions; visually, the bottom level is at the top of the hierarchy, and the top levels are at the end of the hierarchy branches. For instance, Day is the bottom level of a Calendar hierarchy, but it falls directly beneath CLND.

2. From the Hierarchy Definition right-click menu; select New Dimension or click the New Dimension button on the toolbar. You can also create a new dimension for a new hierarchy by pressing the insert key on the keyboard when positioned on the hierarchy. This holds true unless there is already one dimension in the hierarchy. For example, after you add 'Day' to the CLND hierarchy, you are unable click CLND again and add 'Week.' However, you can add 'Week' under the 'Day' dimension. You can only have one bottom-level dimension per hierarchy.

Specify dimension information

⇒ **Navigate:** In the dimension area (hierarchy tree structure), select the dimension that you want to specify information for.

1. In the Dimension Info frame, double-click in the field box to edit. Scroll to the right to see all of the fields. You can also resize the columns by placing the cursor over the column until the double sided arrow appears. Drag the column until the desired width is reached.

    📖 **Note:** Most of the information specified for the dimensions pertains to the data file from which hierarchies and dimensions will be loaded.

2. Other than the RPAS name, the columns can be reordered by dragging and dropping the headings. The column positions will return to the default order when the session is closed.

3. Specify the following dimension information:

   ▪ RPAS Name – The RPAS internal name of the dimension.

       📖 **Note:** The RPAS name can contain a maximum of four characters; such as SKU, Styl, Clss, and Scls.

   ▪ User Label – The dimension label that is displayed to RPAS users.

       📖 **Note:** Any string is allowed, but no single or double quotes are allowed.

   ▪ Column – Identifies the order in which the dimension's positions fall in the meta-data load file. Use the up/down arrows in the dimension info table to change the order of the dimensions. Changing the column value will not impact the hierarchy structure or aggregates.

   Example:

   Change the column values if the dimensions in the data load file are not in the same order as in the tree structure. Dimensions will be moved up or down in the table without impacting the aggregates.

   ▪ Prefix – The number of characters (0 to 4) for an automatic prefix to be put on all position names loaded and stripped from all position names exported. The prefix is taken from the internal dimension name. If the number of characters is greater than the dimension name, an underscore is used to pad the prefix. If the number of characters is less than the dimension name, the right-most characters from the dimension name are used. The default is 4. The DAY dimension in the CLND hierarchy is set to 3 by default and cannot be changed. For example, if data loads and exports use numeric SKU identifiers, such as 123456; the internal representation in RPAS could be to prefix this identifier with SKU,_such as SKU_123456.

       📖 **Note:** This is optional. There is no prefix if a prefix of 0 is defined.

   ▪ Start – This field identifies the start position of the dimension name in the hierarchy load file. It is calculated from entries in the width column.

   ▪ Width – The width of position names for this dimension in the hierarchy load file.

   ▪ Label Start – Identifies the start position for the label in the hierarchy load file. This field is calculated from entries in the label width column.

   ▪ Label Width – Width of position labels for this dimension in the hierarchy load file.

- Aggs – Establishes the relationship of the dimension with the other dimensions in the same hierarchy. Specifically, this field represents the child dimension, which aggregates up to this dimension (the parent dimension). This field is not applicable if the dimension is at its lowest level (for instance, a dimension that has no child dimension). It can be edited by using the drop down list or you can drag and drop in the hierarchy pane.

- Database – The name of the database in which the dimension information is stored. There are multiple databases for each domain, and the database for a measure should go under the sub-directory of a domain. For instance, if there is a database called "Sales," the sub-directories may include "regular sales," "promotional sales," and "clearance sales."

    Note: The size of a database cannot exceed 2 GB. Create new databases when this size limit is approached. In the previous example, you may want to create databases for each of the sub-directories (regular sales, promotional sales, and clearance sales).

- User Dimension – User-defined dimensions are used to group items in the same hierarchies in any manner that the user desires. For instance, a user may wish to group SKUs by margin or stores by consumer volume. When selected, this field indicates that the dimension is user maintained. Positions and position mappings for user-defined dimensions are defined in the RPAS administrative workbook template "Hierarchy Maintenance." This data cannot be loaded like regular (non-user-defined) dimensions in the hierarchy load process.

- Translate – When selected, this field enables the position labels for the dimension to be translated into multiple languages (if using a multi-lingual environment which is set as a Workspace Property for a given project). Positions are loaded into the domain in the native language of the domain via the standard hierarchy load process; position labels for additional languages are loaded into special measures that are used in multi-lingual domains. With the proper setup these translated position labels can be displayed in workbooks in the RPAS client instead of the loaded position labels.

  Contact Oracle's Translation department for additional information about multi-lingual domains and localization. Note this option must be selected before building a domain for it to take effect.

Delete a dimension

⇒ **Navigate:** In the Configuration Manager, select 🟦 Project > 🟦 Solution > 📊Hierarchies. The Hierarchy Definition window opens in the workspace.

1. In the Dimensions area, select the dimension that you want to delete.

2. Click the delete ✖ button.

    Note: If you delete a dimension, all of the dimensions that are structurally dependant on it are also deleted. Since this is a single-rooted tree, and we work from the bottom up, you will loose the entire parent dimensions that are associated with the tree. Therefore, in a product hierarchy; if you delete department, group and division will also be removed without warning, and there is no undo.

    Note: You cannot delete the user dimension, which is contained in the ADMU hierarchy.

Edit a dimension

⇒ **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > ☐Hierarchies. The Hierarchy Definition window opens in the workspace.

1. In the Dimensions area, select the dimension that you want to edit.

2. Click on the field you that want to updated.

3. Enter the new information.

4. To change the order of the dimension, click the left [‹] button or right [›] button.

📖 **Note:** The [∧] button and the [∨] button are used to change the order of hierarchies only.

Create a branched hierarchy

⇒ **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > ☐Hierarchies. The Hierarchy Definition window opens in the workspace.

1. In the Hierarchy Definition window, select the dimension that will be the base of the branched hierarchy.

📖 **Note:** Make sure that you select a dimension that will have more than one parent (that is, where the branch starts), and create another parent dimension. Branches can never join together. For example, you cannot have style-subclass-class and style-supplier-class.

2. From the Hierarchy Definition right-click menu, select New Dimension. The new dimension will be created below the selected dimension.

3. If necessary, you will need to review and fill in the Dimension Info table for the new dimension. The system sets the aggs field to the dimension that was selected to be the base dimension.

Dimension attributes

Dimension attributes allow for additional information to be defined for the positions of a given dimension. These attributes are commonly used to define and display an alternate label for a position (other than the loaded position label) or to display supplemental information about a position, such as the status of a given product or store. Data for dimension attributes is stored in measures; information about defining dimension attributes is available in Dim Attr.

Data Interface Tool

Overview

The data interface tool is used to specify information about measures that will be loaded into the domain. This includes the intersection into which data will be loaded and information about the input file.

Procedures

Data can only be loaded into stored, realized measures in the domain; therefore only such measures can be used in the data interface tool. Please refer to the section "Work with measures" (in the Measures overview section of the Solutions chapter).

Specify the data interface for a measure

⇒   **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ⊕ Data Interface. The Data Interface Manager window opens in the workspace.
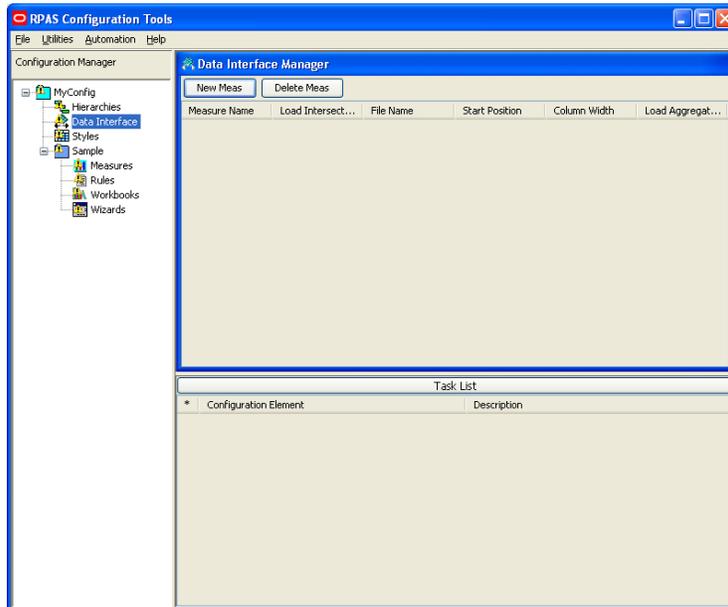
1.   Click **New Meas**. The New Measure Specification window opens.
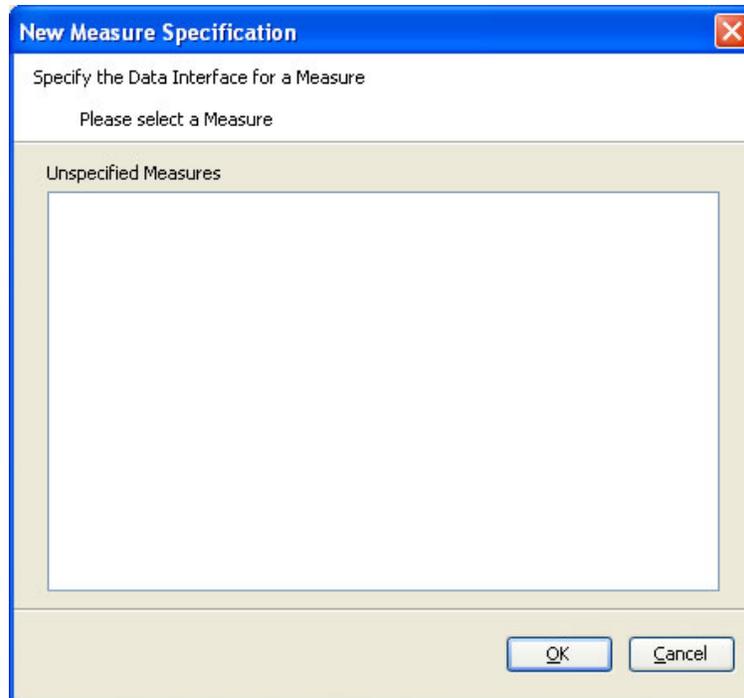
    Note: This is a filtered list of all possible measures in the configuration; measures will be displayed in this list if they are: realized, stored (defined databases), and not already defined in the data interface tool.

2. Select the measure whose data interface you are defining.

3. Click **OK**. The measure is displayed in the Data Interface Manager window.

Edit data interface information for a measure

⇒ **Navigate:** In the Configuration Manager, select 🔲 Project > 🔲 Solution > 🔀 Data Interface. The Data Interface Manager window opens in the workspace.

1. By default the Load Intersection field is populated with the base intersection of the measure. If the data for a given measure is being loaded from a lower dimension/level than the base intersection of the measure, this value can be overridden to specify the level at which the data will be loaded. For example, you would override this field and select the "week" dimension if the data being loaded is representative of the "day" dimension but into a measure with a base intersection of week.

In this illustration, the Load Intersection is "Day_sclshnl."
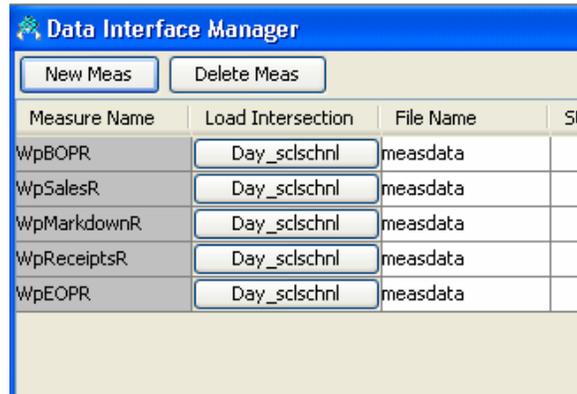
To override this value, click on the Load Intersection field. The Select Intersection window opens.
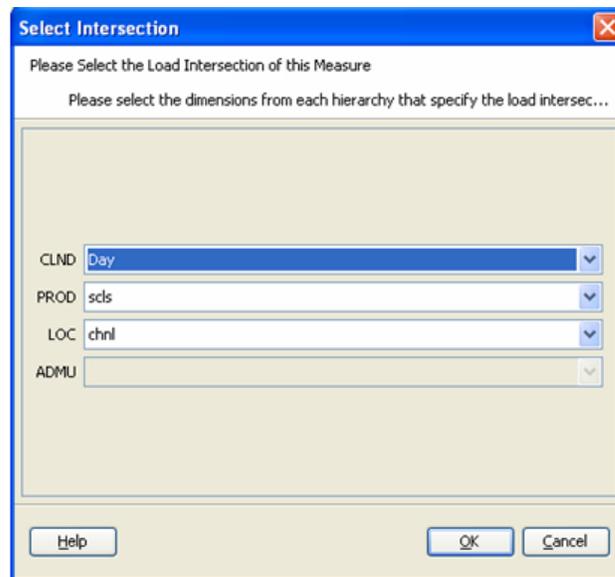
    Note: The load intersection can only be overwritten to load the data into the measure from below the base intersection. To specify the load intersection: :

   a. Select appropriate dimensions using the list boxes.

    Note: Only those dimensions that are at the same level as the base intersection or below will be displayed for the load intersection.

   b. Click **OK** to save any changes and close the window.

2. In the File name field, enter the file name from which data for the measure will be loaded.

3. In the Start Position field, enter the file start position for the measure data.

4. In the Column Width field, enter the file column width for the measure data.

    Note: The column width defaults to 8, but can be changed.

5. If the Load Intersection was overridden to specify that the data is to be loaded from a level below the measure's base intersection, the measure's default aggregation method is used to aggregate the data unless the Load Aggregate field is populated to specify an alternate aggregation method. Select the appropriate aggregation method by clicking on the drop-down box.

Delete a measure

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ⬥Data Interface. The Data Interface Manager window opens in the workspace.

1. Select the measure's data interface parameters that you want to delete.

2. Click Delete Meas. You are prompted to confirm your choice.

3. Click **Yes**. The measure is removed from the table.

Chapter 4 – Solutions

Measures and components

Overview

The measure definition tool allows you to define components of measures and to specify default properties for each component. After you have defined the component structure, the measure definition tool generates prototype measures by combining the components that you select. You can also specify properties for individual measures, which will override the component default properties for those measures.

A major component is the highest level in the component inheritance hierarchy. Properties defined at this level are inherited by all minor components created under a major component.

Within each major component, you create one or more minor components. You can also create a minor component underneath a minor component. You can also modify properties at the minor component level.

Once you have defined your major and minor components, the measure definition tool generates prototype measures that are based on the combination of components you select. These prototype measures cannot be used elsewhere in the configuration tools until you have realized them. See Realizing and Unrealizing measures for more information.

Component Process
Create major components, from which measures are comprised.
Create minor components, which are sub-groupings or specific items in a major component.
Define measure properties at the major component level. The minor components will inherit the properties associated with the major component they belong within.
If necessary, modify the measure properties at the minor component level.

    Note: Components have no structural impact on the built solutions, and they are not exposed to end users. Components are intended to be a convenience to aid the configurer (configuration tools user) to easily group measures together and to set measure properties at higher levels.

Measure properties

Inheritance

Measure properties are inherited at the component level. Properties defined for a component are spread down to all of the minor components that belong to that component and also to the measures that are associated with that component.

Over-writing

Measure properties can be over-written either at the minor component or measure level. Once a property is changed at the measure level, changes made at the component level will no longer be inherited by that measure.

Measure validation within the measure tool

Various types of validation take place take place in the configuration tools. Heavy validation occurs in measures. The rule tool is heavily dependent on the measure tool for validation of the measure used in the rules. If a measure is deleted and used in a rule, the rule becomes invalid.

When working with a configuration that has large measure and rule sets, real time validation performed in the measure tool may impact performance. Real time validation may be turned off in order to reduce the potential performance impact in the measure tool. If it is turned off, no real time validation will occur when measure changes are made. Validation can then be manually initiated in the Rule Tool by hitting the toolbar button "Perform measure content validation."

To turn off real-time validation, uncheck the "Measure Content Validation" checkbox in the Tools Preferences of the File menu.

Procedure

To turn off real time validation:

Clear the Measure Content Validation Enabled check box at the bottom of the measure tool.

OR

In the Workplace Preferences dialog box, clear the Measure Content Validation check box.

Work with components

Measure component design

The following two basic principles should be kept in mind to make the measure tool as powerful as possible.

1. Major and minor components should be designed with the idea of maximizing default properties.

2. Use minor components to make measure definition manageable.

📖   **Note:** Refer to the following example when you read the following two paragraphs:

Major and minor components should be designed with the idea of maximizing default properties.

The measure property for real numbers should be explicitly set a minimal number of times at the minor component level. Measures are the composition of minor components, and defaults can be set at the minor components level.

Use minor components to make measure definition manageable

Consider a configuration has 2000 measures, and 1500 of them are of data type real. Avoid grouping all 1500 measures into a single minor component, because you cannot easily edit smaller subgroups of 1500 measures. Minor components can also have minor components; so within the 1500 measures, you could break them out further. This is based on the aggregation method, such as total, max, and recalc. Ideally, the "checking" of a lowest level minor component should allow you to easily view and manage every resulting measure within that minor component.

Procedures

Create a component

⇒ **Navigate:** In the Configuration Manager, select 🗂 Project > 🗂 Solution > ▥ Measures. The Measure Manager window opens in the workspace.
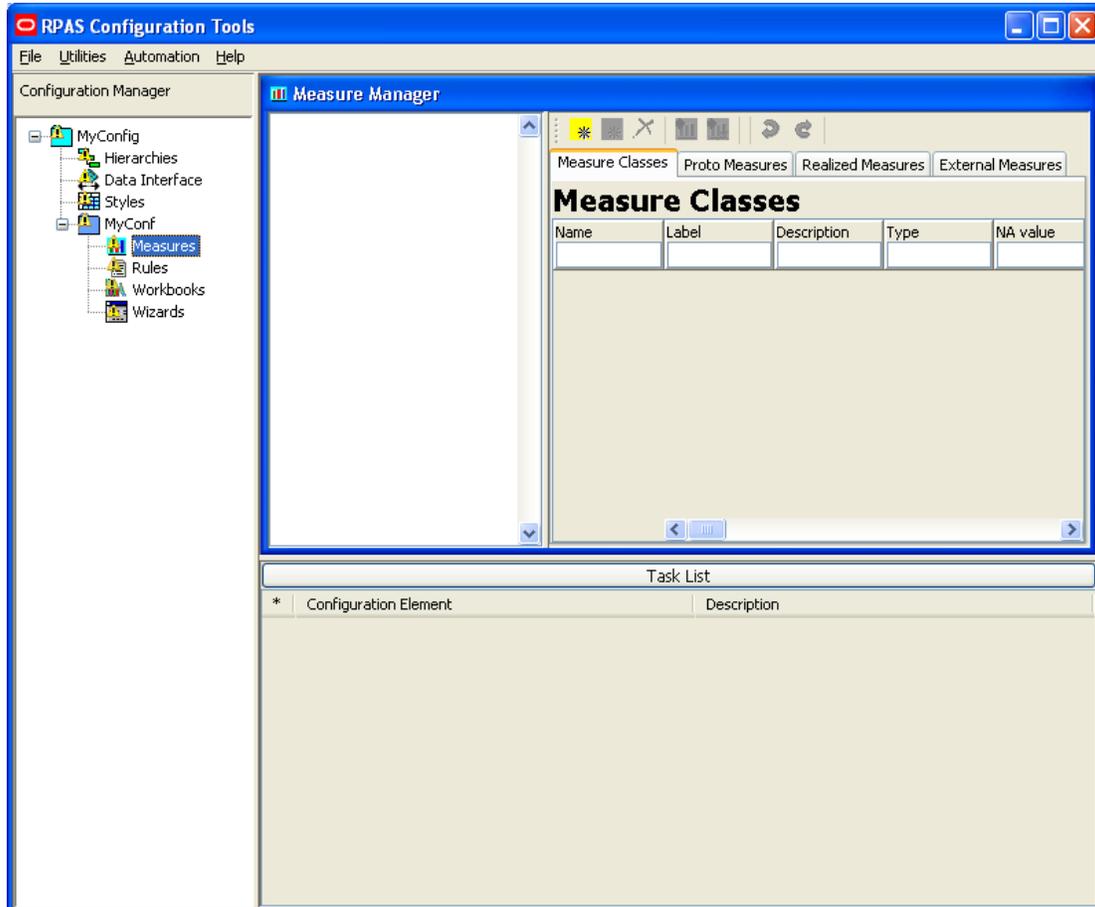
1. Right click in the left hand pane of the Measure Definition window.

2. Perform one of the following:

   ▪ Select Add Major Component. The component is displayed in the left hand pane.

   ▪ Select Add Minor Component. The component is displayed in the left hand pane.

Specify component information

1. Select the Components tab.

2. Specify the following information, which is applicable to the measures that are inheriting from the component.

    📖   **Note:** The defaults that are entered for major components are inherited by the child minor components and the auto-generated measures. Not all properties need to be entered for all components. Where the same property is inherited from two or more components, the property used for a measure is that from the component that is highest up the list of components.

- Identifier – The identifier of a component is used to identify the component within the configuration tools, but it is not used in the configured solution. Measure names (identifiers), which are used in the configured solution, are built by concatenating the identifiers of the components that the measure is built from. They are built in the sequence that the components appear in the list of components.

- Label – The label of the component is used to generate measure labels in a similar way that measure names are generated. These are displayed to RPAS users. There is no maximum size limit, but you should keep grid display limitations in mind when creating a measure label.

- Description – A description of the component is used to generate measure descriptions in the same way that measure labels are generated. This is used for information within the configuration tools only and can be a long description.

- Data Type – Select:

  - Real – Floating point numeric values. Most measures are of this type.

  - Int – Numeric integer values. There are no special 'spreading' algorithms for integer measures, which should normally be used only for measures that are calculated 'bottoms up.' Formatting can be used to display real measures as integer value in the RPAS client.

  - Date – Date and time. Can easily be converted to position names using standard RPAS functions.

  - String – Variable length strings, which are typically used for notes and names.

  - Boolean – True or false values, which are typically used for flags and indicators.

- NA Value – This is a value (typically zero for numeric measures) that is not physically stored, but is inferred. It is used to help with storage and calculation efficiency, and it may be changed by RPAS (in full-evaluation mode) if better efficiencies can be obtained with a different value. See the Calculation Engine Users Guide for more information.

- Base Intersection – The lowest level at which data is stored for a measure. In the domain, the measure is only stored at the base intersection. Inside a workbook (for performance reasons), values for the measure may be stored above the base intersection. Nevertheless, whether stored or not, values for aggregated levels may be viewed in a workbook and used in calculations in workbooks or domains. Double click in the field to open the Select Intersection window. The hierarchies that were defined using the Hierarchy tool are displayed. One dimension from each hierarchy can be selected.

▪ Default Agg Method – Select the default aggregation method for the component. Information about aggregation methods and their usage is available in the RPAS Rule Functions Reference Guide and the RPAS Calculation Engine Users Guide.

▪ Spread Method – Select the default spread method for the component. Information about aggregation methods and their usage is available in the RPAS Rule Functions Reference Guide and the RPAS Calculation Engine Users Guide.

    Note: The spread method is the default spread method, which can be overridden by the user on edit in the RPAS UI. For all 'populated' spread methods (ending with "pop"), the spread method is the same as the underlying method (for instance, prop_pop is like prop), except that only cells with a value different from the naval are used in the spreading, and cells with a value equal to the naval are ignored.

▪ Base State – The editability of the measure at the base level (read or write).

▪ Agg State – The editability of the measure at the aggregate level (read or write).

    Note: Set the Agg State to read for those measures that need to be manipulable, but where there is no business requirement to manipulate them other than at their base intersection. Often, there would be no sensible way to spread such measures.

    Note: The manipulability of measures will change according to 'protection processing' principles. Therefore, base state and agg state should only be used to override the result of protection processing. In particular, to make a measure non-manipulable that protection processing would otherwise allow to be manipulated. See the calculation engine user guide document for more information.

▪ View Type – The View Type field holds properties for two types of measures: those that are calculated when viewed, and those that are synchronized with other measures. If the view type is none, the measure is of neither type.

If the View Type is view_only, the measure is not calculated during a normal calculate cycle, and it is calculated on-the-fly when required (For instance, for viewing). Such measures must have an aggregation type of recalc and should appear on the left hand side of only one expression in a rule group. It may not appear on the right hand side of any expressions. The measure should not have a database assigned.

Synchronized measures are effectively viewed across time of two or more other measures where changes, locks, etc. to those other measures are immediately reflected in the synchronized measure, and vice versa. For example, a 'closing stock' measure may be synchronized with a "season opening stock" measure and an "opening stock measure" so that a change to "opening stock" in week 3 will immediately cause the same change to be applied to "closing stock" in week 2 (since closing stock in week 2 and opening stock in week 3 are the same). Synchronized measures require a synchronization type in the View Type property, which must be one of sync_first_lag, sync_lead_last, sync_first or sync_last, and a list of measures to synchronise with in the Sync With property. For more details see the calculation engine user guide.

- none – The measure is calculated normally.

- view_only – The measure is calculated when viewed.

- sync_first_lag – Period 1 is from the first measure (no calendar); Periods 2..N is from the second measure 1..N-1 (lag) [for example, bop yields os & eop]

- sync_lead_last – Periods 1..N-1 is from the first measure 2..N (lead); Period is N from the second measure (no calendar) [for example, eop yields bop & cs]

- sync_first – Gets period 1 from the measure (similar to pst along calendar dimension) [for example, os yields bop]

- sync_last – Gets period N from the measure (equivalent to pet along calendar dimension) [for example, cs yields eop]

- Sync With – A comma-separated list of measures used for synchronization. Depends on view type.

- Insertable – Indicates whether the measure can be inserted as an extra measure in workbooks built from templates that are not configured to contain the measure. "Insertable" measures can be added to a workbook during the wizard process on the "Extra Measures" wizard page before a workbook is built, or by inserting the measure in the Show/Hide dialog window in the RPAS client inside a built workbook. Measure security must also be defined in for "Insertable" measures in the RPAS Security Administration workbook template. Possible values are true and false. See the RPAS Administration Guide for additional information about measure security.

- UI Type – Indicates whether the measure is a picklist or not. If a measure is defined as a picklist, users must choose a value for the measure from a list of valid values defined using the range property. Possible values are "picklist" or blank.

- Range – Specify suitable range for the measure at edit time. For numeric values, the syntax is as Lower Bound : Upper Bound. For strings, it is the maximum length. For picklist measures (both numeric and string) values can be explicitly listed as "a,b,c,d…"; Additionally, labels can be specified for each value, such as "valuea (labela), valueb (labelb), …" where the labels will be displayed in the user interface.

- Purge Age – The number of days (without a load) before measure data is purged.

- Refreshable – No longer used in the RPAS 11.x architecture. Instead, the refresh rule group is used to determine which measures are refreshed. This measure property will be depreciated soon.

- Lowerbound – A valid, realized measure name that provides the minimum bounding values for this measure's data cells. The lowerbound measure and the measure that contains the lowerbound property must be either type integer or real. Integer value is not accepted as a value. A valid, realized measure name must be used as the value. Lowerbound is the name of the measure that holds the lowest legal value.

- Upperbound – A valid, realized measure name that provides maximum bounding values for this measure's data cells. The upperbound measure and the measure containing the upperbound property must be of either type integer or real. An integer value is not accepted as a value. A valid, realized measure name must be used as the value. Upperbound is the name of the measure that holds the highest legal value.

  📖   **Note:** If one, but not the other, is specified, only one limit is checked. For example, if a lowerbound is set, but an upper bound is not, valid values are greater than or equal to the value held in the lowerbound measure, but with no upper limit. The lowerbound/upperbound measures can be non-conforming with respect to the measure that has bounds, and the value to be used will be obtained by normal non-conforming processing – For example, 'replicated down' from higher levels or 'aggregated up' from lower levels.

- Sp Val Type and Sp Val – Sp Val Types are the different types that can be used to specify special values. The different types include null, cellvalue, and user-entered. Sp Val is used to allow the user to enter the value when the Special Value Type is set to user-entered. These two properties can be used together to specify how to display cell values on the User Interface (UI) that have a value equal to the naval of the measure. The default is that such cells will have their cellvalue shown, which is what the value of cellvalue in the property sp val type means; however, these properties can be used to override the default to either show 'null,' which is defined below, or to show a specific value. To configure the values to display null, put the value null into the sp val type property. When null is configured, the cell will be blank in the UI when the value equals the naval for a numeric, a date, or string measures. For Boolean measures, it will be a grayed-out check box. When any other value is required, the configurer should select user-entered for the sp val type, and enter the value to be displayed in the sp val.

  - For the Special Value field; the entry in this field must be of the same data type as the measure, and validation is enforced in this field.

  - For a Boolean measure; when the Special Value Type field is set to User Entered, the only valid entry for the Special Value field is either true or false.

  - For a Date measure; when the Special Value Type field is set to User Entered, the user must enter a date in the format of YYYYMMDD.

  - By default each measure will be registered with cellvalue as the default special value behavior. For cases where a Special Value Type setting other than User-Entered is used, but a Special Value entry is provided, the Special Value entry will not be used. Instead the measure will be registered with cellvalue as the default behavior.

  - When the user sets the Special Value Type setting to User-Entered, but does not provide a Special Value entry, the Special Value Type will not be used. Instead the measure will be registered with cellvalue as the default behavior.

  - When a domain has already been built, which includes a measure with a special value setting, and the user removes that special value setting for that measure and patches the domain, the measure will get updated with cellvalue as the default special behavior.

  - RPAS allows for the special value measure property to be updated.

- Dim Attr – a checkbox indicating that the measure should be registered as a dimension attribute. Dimension attributes allow for additional information to be defined for the positions of a given dimension. This is commonly used to define and display an alternate label for a position (other than the loaded position label) or to display supplemental information about a position (such as the status of a given position).

For a measure to be eligible to be a dimension attribute, the following requirements must be met:

    Note: There is no restriction on the type of measure that can be specified as a dimension attribute.

- ⁕ Must be realized

- ⁕ Must be 1-dimensional (base intersection has only one dimension)

- ⁕ Must be stored (must have a defined database)

- Dim Attr Name – (optional field) This is only to be used if the measure is set to be a dimension attribute measure.

- Dim Attr Label – (optional field) This is only to be used if the measure is set to be a dimension attribute measure.

- Allowed Aggs – The set of the allowable aggregation methods for the measure based on the measure data type. You can add so called extended measures to UI views that are normal measures but with aggregations based on different aggregation methods. The aggregation methods that are available for selection are based on the Allowed Aggs of the base measure. The same base measure can have multiple extended measures based on different aggregation methods.

Edit the components

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▮▮ Measures. The Measure Manager window opens in the workspace.
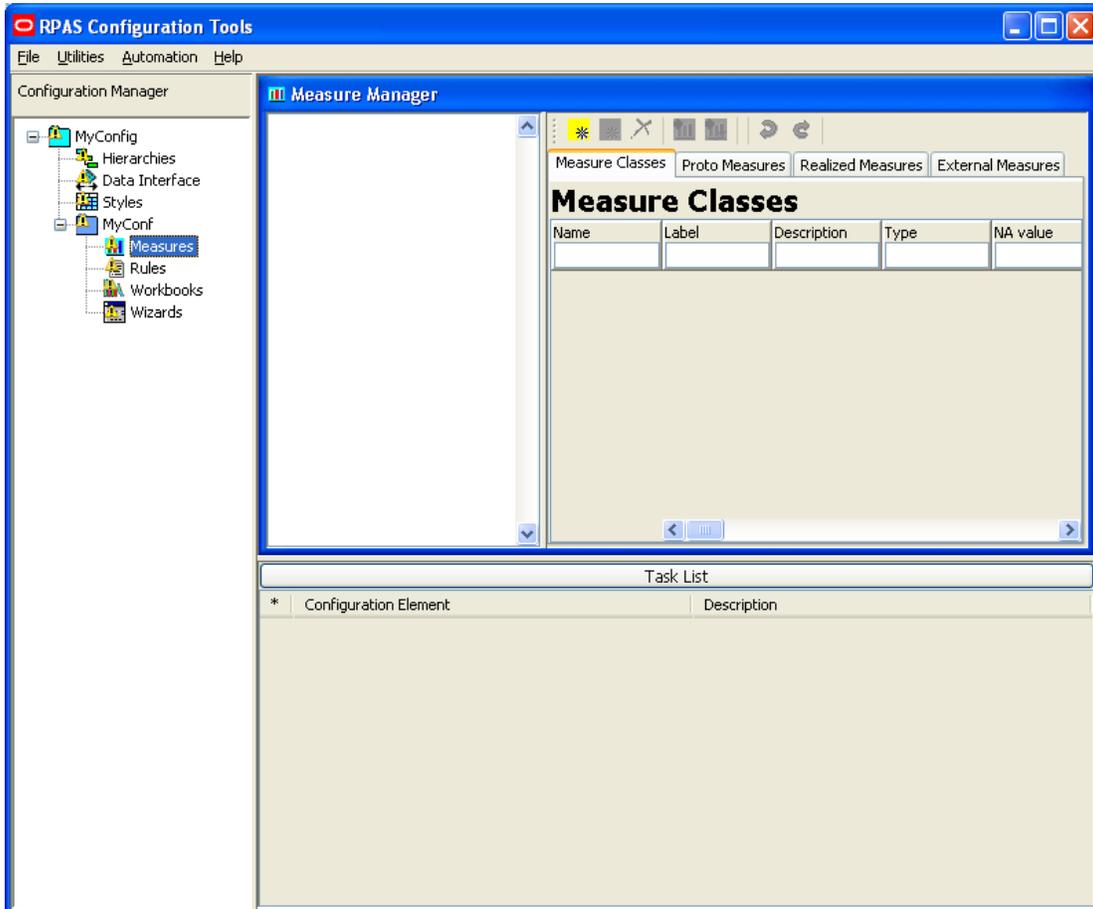
Push components down

1. In the Measure Definition window, select the component that you want to push down.

2. Right-click in the Measure Definition menu, and select **Push Down**.

    📖   **Note:** The component is pushed down one level in the component hierarchy, and a new component is created to take the place of the pushed down component.

    📖   **Note:** You cannot push down a major component.

Pull components up

1. In the Measure Definition window, select the component that you want to pull up.

2. Right-click in the Measure Definition window and select Pull Up. The component is pulled up one level in the component hierarchy.

    📖   **Note:** You cannot pull a minor component up to become a major component.

Display or hide components

- To display information about a component, select the check box next to the component name.

- To hide information about a component, clear the check box next to the component name.

> 📖 **Note:** Selecting or clearing a check box for a major component causes the check boxes for all minor components underneath it to be selected or cleared. This check box is also used to enable a component so it becomes active when measures are generated.

Rename a component

⇒ **Navigate:** In the Measure Definition window, select the component that you want to rename.

1. Right-click in the Measure Definition window, and select Rename.

2. Type the new name for the component, or double-click the component, and type the new name.

> 📖 **Note:** Changing the component name also changes the component label, but not the identifier.

Move components

⇒ **Navigate:** In the Measure Definition window, select the component that you want to move.

Drag the component to its new location and release it.

> 📖 **Note:** You cannot move a major component.

Work with measures

Overview

The Measure Tool allows you to create and name the measures by selecting major and minor components that were already defined. By default, the measures inherit the properties that were defined for the component. To create a measure, you selecting the components you want to use to construct measures. The measure tool will generate prototype measures (all different measure classes; realized and unrealized) for all of the combinations that you select. This saves you from creating all of the measures manually.

You can then overwrite the properties for individual measures by entering them the same way as on the components tab. Once you over-write properties at the measure level, changes made at the component level will no longer spread down to that measure as it will retain the over-written value.

When a measure is auto-generated by the measure tool, it is considered a prototype measure and cannot be edited or used in any other configuration component. To enable the measure to be edited and/or used elsewhere in the configuration, it must be realized first. If the measure is not going to be used, it does not need to be realized. Only realized measures are recognized and used in configuration components, such as expressions and workbooks.

Procedures

Realize and unrealize measures

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▐▌Measures. The Measure Manager window opens in the workspace.
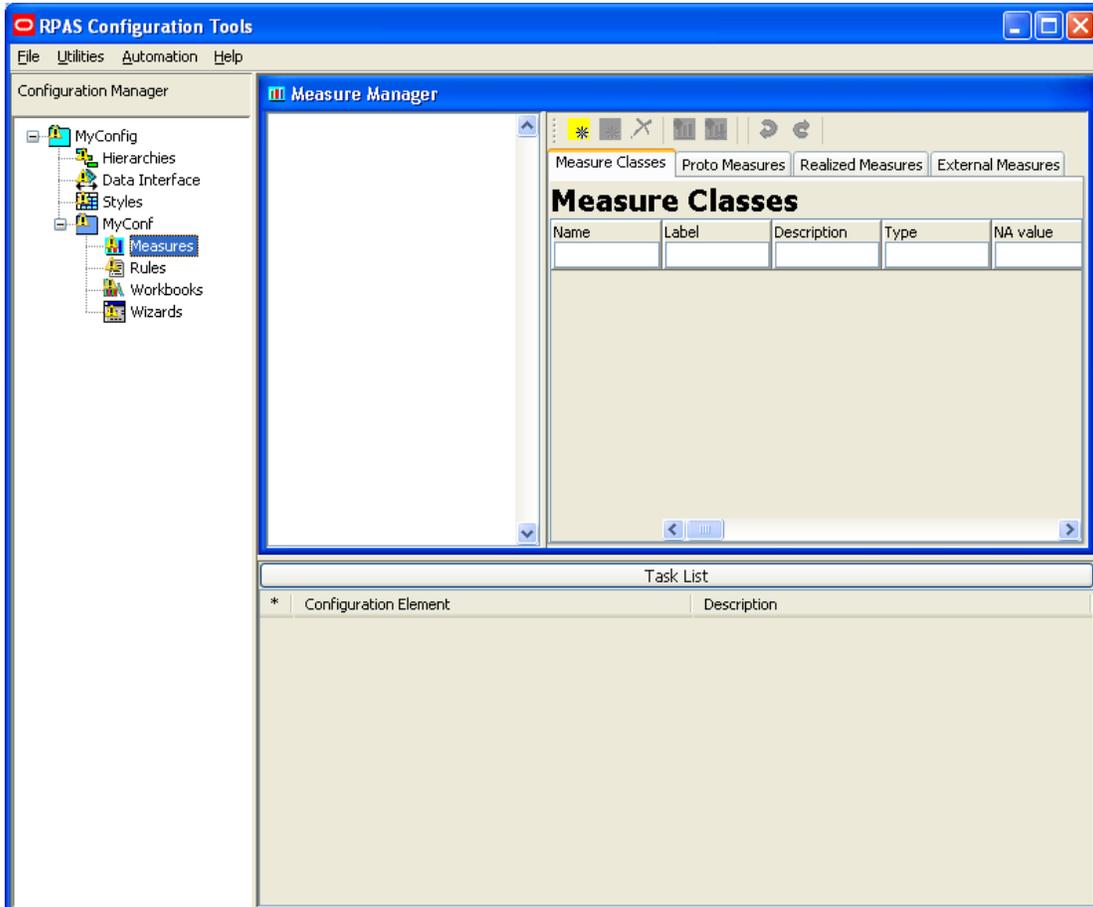
1. Select the measures tab.

2. In the components tree, select the check boxes for the components that you want to use to create measures. The measure tool will create measures using all of the combination of components that you selected. This process filters the list of prototype measures that are shown from all combinations of components to the combinations of components that have been selected. It uses those components to determine which prototype measures to show.

Realize a measure

1. Select the components that contain the measures you want to realize.

2. Select the check box in the Realized column for the prototype measure that you want to realize.

Unrealize a measure

1. Select the components that contain the measures that you want to unrealize.

2. Unselect the check box in the Realized column for the prototype measure that you want to unrealize.

Rename a measure

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▮Measures. The Measure Manager window opens in the workspace.

1. In the Measures tab, click the name of the measure that you want to rename.

2. Type the new name for the measure.

📖 **Note:** The measure must be realized before you can rename it.

Show measures

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▮Measures. The Measure Manager window opens in the workspace.

1. In the Measure Manager window, select the measures tab.

2. Select the check boxes next to the components whose measures you want to show.

OR

1. Select the components containing the measures that you want to show.

2. Right-click the Measures Definition menu, and select Show.

Show all measures

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▮Measures. The Measure Manager window opens in the workspace.

Right-click in the Measure Manager window and select Show All.

📖 **Note:** Due to memory constraints, all prototype measures may not be displayed. In this case, you will receive an error message instructing you to deselect some measure components.

Hide measures

⇒ **Navigate:** In the Configuration Manager, select ▣ Project > ▣ Solution > ▥Measures. The Measure Manager window opens in the workspace.

1. In the Measure Manager window, select the Measures tab.

2. Uncheck the check boxes next to the components whose measures you want to hide.

   OR

1. Select the components that contain the measures you want to hide.

2. Right-click in the Measure Manager menu, and select Hide.

Hide all measures

⇒ **Navigate:** In the Configuration Manager, select ▣ Project > ▣ Solution > ▥Measures. The Measure Manager window opens in the workspace.

1. In the Measure Manager window, select the measures tab.

2. Right-click the Measure Manager window, and select Hide All.

Filter measures

⇒ **Navigate:** In the Configuration Manager, select ▣ Project > ▣ Solution > ▥Measures. The Measure Manager window opens in the workspace.

In the Realized Measure tabs, you can sort measures by property value.

1. In the Realized Measures tab, click in the filter field at the top of the table for the property that you want the measures filtered by.

2. Enter the value on which measures are to be filtered. The measures are filtered according to the value of the property entered.

　　📖　　**Note:** You only get the measures whose value for the selected property is the same as the value entered in the filter box. This field is case sensitive.

Rules

Overview

The rule definition tool allows you to define, organize, and manage rule sets, rule groups, and rules. It also allows you to create expressions and add expressions to rules. The RPAS calculation engine has only one active rule group. Even if all that is required is a single expression, that single expression will be in a rule, and that single rule will be in a rule group. See the Calculation Engine User Guide for more information.

Rules are groups of expressions that describe the relationship between measures.

When a rule has multiple expressions, those expressions are given a priority sequence to help the calculation engine select a calculation path that follows business priorities. When given a choice, the calculation engine will always select the highest priority expression in the rule that is available to be selected. Considerable care should be taken in the design of rules to ensure that appropriate expression priorities are established. The business priority may vary from implementation to implementation, and it may vary from one type of plan to another in the same implementation.

Use great care when you have expressions in a rule that use different measures, because the basic workings of the calculation engine assume that all expressions use the same measures. Also, it is possible to have an expression with multiple LHS measures.

   Note: This implies each expression only has a single LHS measure.

See the Calculation Engine Users Guide for more information.

Expressions

An expression describes and solves the relationship between measures in a way that causes a measure to be calculated through the expression. In some cases, there may be business reasons for wanting more than one of the measures in a relationship to be calculable or solvable through that relationship.

Expressions are the basis of all calculations of the relationships between measures, and they are evaluated by the calculation engine during a calculation. They are written in a syntax that allows for the calculation of a single measure from other measures, constants, and parameters by using standard arithmetical functions and a rich set of mathematical, technical, and business functions. Expressions have multiple results, for example:

Expression 1: ReceiptUnits = ReceiptsValue / ReceiptsPrice

This expression specifies the way ReceiptUnits are calculated. ReceiptUnits are calculated by the division of ReceiptsValue over ReceiptsPrice.

   Note: Measures are not only calculated based on expressions. They are also calculated based on spreading and aggregating.

Rule groups

In RPAS, a rule group is a collection of rules that are treated as a unit by the calculation engine with the integrity of all the rules in the rule group being maintained together.

Rules within a rule group are given a priority. The calculation engine uses this to select a calculation path that follows business priorities. It does this by using rule priorities to determine which rule to enforce when there is a choice to be made. Although there may only be one active rule group at any time, you can define multiple rule groups to satisfy different calculation requirements.

Types of rule groups

Rule groups may be one of four different types:

- Load – The RPAS application automatically uses the load rule group when loading data into the workbook.

- Calculate – The RPAS application uses calculate when the user needs to apply any changes (not just from the worksheet).

- Refresh – The RPAS application automatically uses the default refresh rule group to refresh data.

- Commit – The RPAS application automatically uses the commit rule group when committing data to the domain.

You can commit data to the domain for a measure that does not exist in the workbook. Do this by including the appropriate rule to calculate the measure (with the master modifier) in the commit rule group. Similarly, a measure may be loaded into a workbook that does not exist in the domain. Do this by including an appropriate rule to calculate the measure in the load rule group.

RPAS supports multiple calculation rule groups. Menu options may be configured to allow the user to select a different calculation rule group. RPAS ensures a smooth transition from one calculation rule group to another.

Rule group validation

Within a solution, there may be many rules defined, and each rule is validated individually. Rules within a rule group are also validated in the context of all the other rules in that rule group. While a rule may be perfectly valid syntactically, it may not be valid within a particular rule group. Rule group validations include:

- Each rule in a rule group must represent a completely different measure relationship. This means:
  - No two rules in a rule group may use exactly the same collection of measures.
  - A rule group cannot use a collection of measures that is a sub-set of the collection of measures in another rule.

- There must be an expression that calculates each recalc measure.

- Any measure that is on the left-hand side (LHS) of the only expression in a rule may not be on the LHS of any other expression.

Multiple Refresh Rule Groups

The Refresh Rule Group 'refreshes' (supplies new values for) data, which can generally be thought of as being 'external' to the workbook. An example would be "actuals" where you build the workbook in week 5, so you would have actuals for weeks 1-4. In week 6, you want to refresh the actuals to get the actuals for week 5.

The Configuration Tools and RPAS support the use of multiple refresh rule groups. Within the rule tool, there is the ability to create multiple refresh rule groups within a rule set. These multiple refresh rule groups can then be assigned to a workbook template using the Workbook Designer, and they will be available for the user to select from in the RPAS client.

- A workbook contains all the rule groups in a single rule set, so if multiple refresh rule groups are required in a workbook, they must all be in the same rule set.

- You may consider naming your rule groups so the usage of the refresh rule groups are reflected in their names; such as refresh_all, refresh_actuals, and refresh_manager.

Rule group transitions

    📖   **Note:** Only one rule group may be active at a time.

Automatic rule group transitions occur:

- On data loading – Data is loaded using the load rule group. This will typically load measures by calculating them from the data values held on the domain using the master modifier, but may also calculate other measures that are not explicitly loaded. When the load is complete, the system will automatically transition to the calculate rule group.

- On data refreshing – Data refreshing causes some measures to be updated from values held on the domain. You select the measures to refresh. Refreshing uses the refresh rule group, but there is no real transition. The measures that are affected by the refreshed measures are treated as affected in the calculate rule group, and a normal calculation of that rule group follows. Effectively, data refreshing causes a calculation using the calculate rule group as if the cells that were refreshed were directly changed by the user.

- On data committing – There is a normal transition from the current calculate rule group to the commit rule group. This typically commits measures by calculating them on the domain by using the master modifier. If any non-master measures are calculated, they will be treated as refreshed, and there will be a 'refresh' transition back to the calculate rule group.

- On workbook building – When a workbook is built, there is an automatic transition from the load rule group to the calc rule group.

Rule sets

A rule set is a collection of rule groups. It is used as a placeholder for containing rule groups, which makes the visual display of rules easier for the end-user.

    📖   **Note:** A rule set is a tools concept only. It does not appear in RPAS.

The RPAS calculation engine

The RPAS calculation engine is an engine that is built to support OLAP type calculations against a multi-dimensional model. The engine is very powerful and flexible.

In the OLAP model, individual pieces of data, called cells, apply to a single position in one or more hierarchies or dimensions. Cells typically include:

- A measures dimension

- A calendar or time hierarchy

- Other hierarchies such as products and locations.

The measures dimension is fundamentally different to the other hierarchies, since measures represent the fundamental events or measurements that are being recorded. The positions in the other hierarchies provide a context for the measurement (for instance; where, when, and what). Measures relate to one another through rules and expressions; and positions in all the other hierarchies relate to each other through hierarchical relationships.

Aggregation and spreading

The RPAS calculation engine is designed to be robust and extensible, but in complete control of the calculation process. It enforces integrity of the data by ensuring that all known relationships between cells are always enforced. Much of the logic of the processing of rules and rule groups depends on this basic principal. RPAS supports two different forms of relationships between cells:

1. Hierarchical relationships that require aggregation and spreading

2. Measure relationships that require rules and expressions

Aggregation and spreading are basic capabilities of the engine that do not require coding by the implementer other than the selection of aggregation and spreading types to use for a measure. Hierarchical relationships, such as weeks rolling up to months or stores rolling up to regions, require the aggregation of data values from lower levels in a hierarchy to higher levels. A variety of methods are used as appropriate to the measure.

To enable such data to be manipulated at higher levels, RPAS supports spreading the changes, which also uses a variety of methods.

The inherent relationships between measures can be modeled through a rich rule and expression syntax. Most of the effort in configuring an application model is in modeling these relationships.

RPAS functions

RPAS Functions are mechanisms for performing operations within an expression that are controlled and executed by the calculation engine.

- The calculation engine controls and executes the evaluation of a function.

- Functions may be used in long expressions with other functions and keywords.

- The data that can be referenced is limited to the scope of the workbook.

See RPAS Rules Function Reference Guide for details about RPAS functions.

Work with expressions

Overview

An expression describes and solves the relationship between measures in a way that causes a measure to be calculated through the expression. In some cases, there may be business reasons for wanting more than one of the measures in a relationship to be calculable or solvable through that relationship. Expressions are the basis of all calculations of the relationships between measures, and they are evaluated by the calculation engine during a calculation. They are written in a syntax that allows for the calculation of a single measure from other measures, constants, and parameters by using standard arithmetical functions and a rich set of mathematical, technical, and business functions. Expressions have multiple results, for example:

Expression 1: ReceiptUnits = ReceiptsValue / ReceiptsPrice

This expression specifies the way ReceiptUnits are calculated. ReceiptUnits are calculated by the division of ReceiptsValue over ReceiptsPrice.

Note: Measures are not only calculated based on expressions. They are also calculated based on spreading and aggregating.

Procedures

Expression ordering

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.
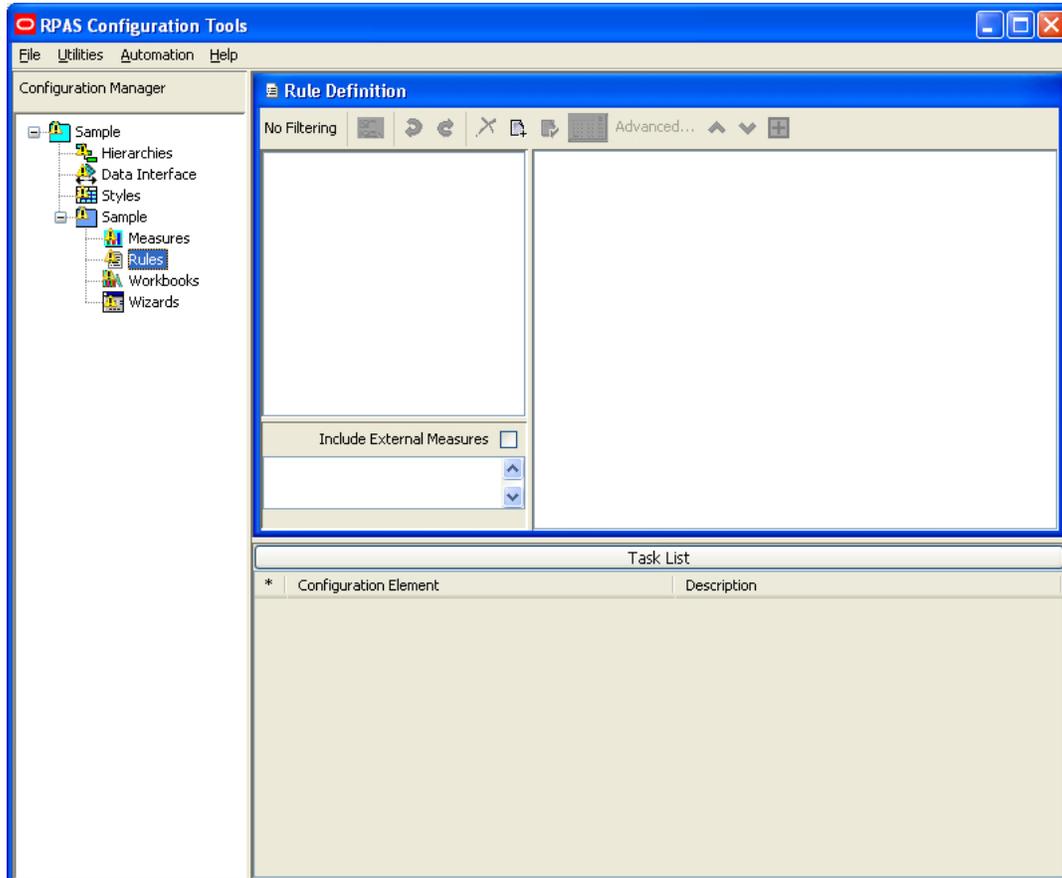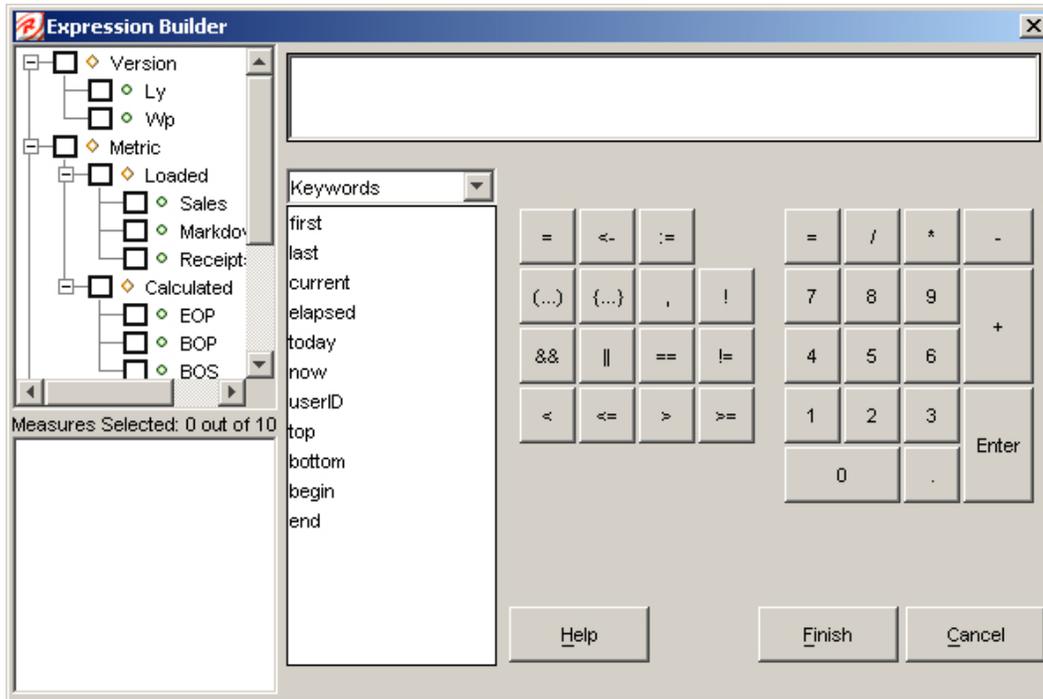

1. In the Rule Definition window, select the rule group that contains the rule that you want to move.

2. Expand the rule to view the expressions associated with the rule.

   - Click the down arrow to move the expression down the list.
   - Click the up arrow to move the expression up the list.

Edit an expression

⇒ **Navigate:** In the Configuration Manager, select 📁 Project > 📁 Solution > 📄 Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that contains the rule you want to edit.

2. From the toolbar, click the expression builder ⊞ button. The Edit Expressions window is displayed.
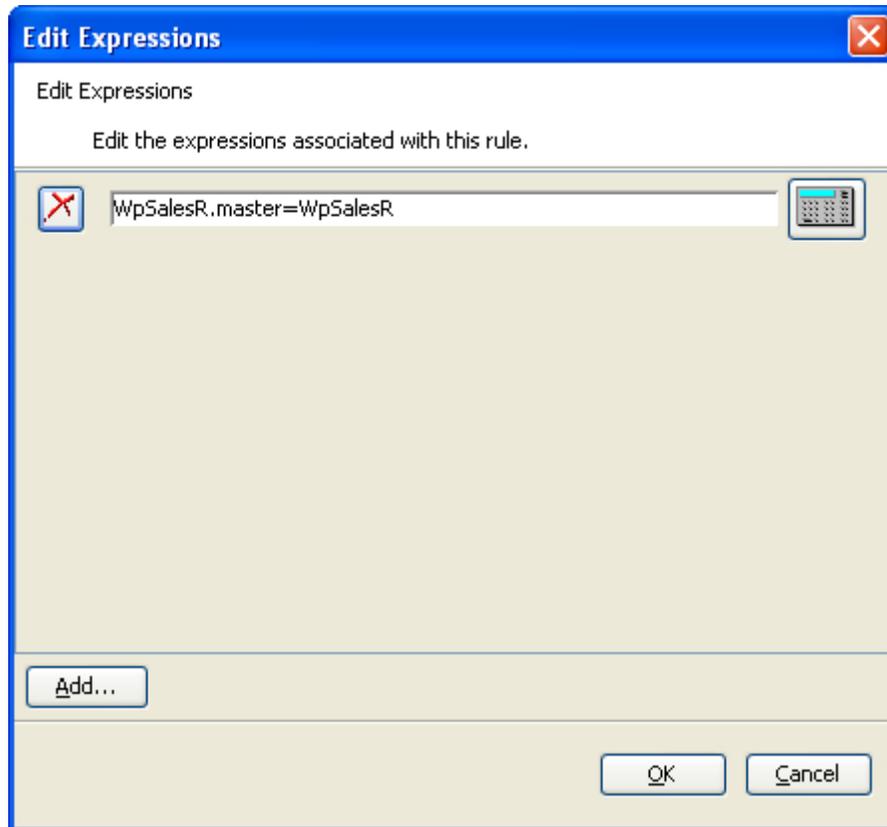
1. To edit an expression:

    a. Edit the expression in the text box.

    b. Click the expression builder [button image] button. The Edit Expressions window is displayed. Use the Expression Builder to make your changes.

2. Click **OK** to save and changes and close the window.

Delete an expression

⇒ **Navigate:** In the Configuration Manager, select [icon] Configuration > [icon] Project > [icon] Solution > [icon] Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that contains the expression to delete.

2. From the toolbar, click the expression builder [button image] button. The Edit Expressions window is displayed.

3. Click the Delete button to the right of the expression in the Edit Expression box.

4. Click OK to delete the expression. Once you click OK, the expression will be permanently deleted from the rule.

Work with rule groups

Overview

A rule group is a collection of rules that are treated as a unit by the calculation engine. The rules in the rule group must be considered together to satisfy the calculation requirements for a specific business process. Rules within a rule group are given a priority, which are used to determine which rule to enforce when there is a choice to be made. Although there may only be one active rule group at any time, you can define multiple rule groups to satisfy different calculation requirements.

Procedures

Create a rule group

⇒ **Navigate:** In the Configuration Manager, select ▢ Configuration > ▢ Project > ▢ Solution > 🗎 Rules. The Rule Definitions window opens in the workspace.
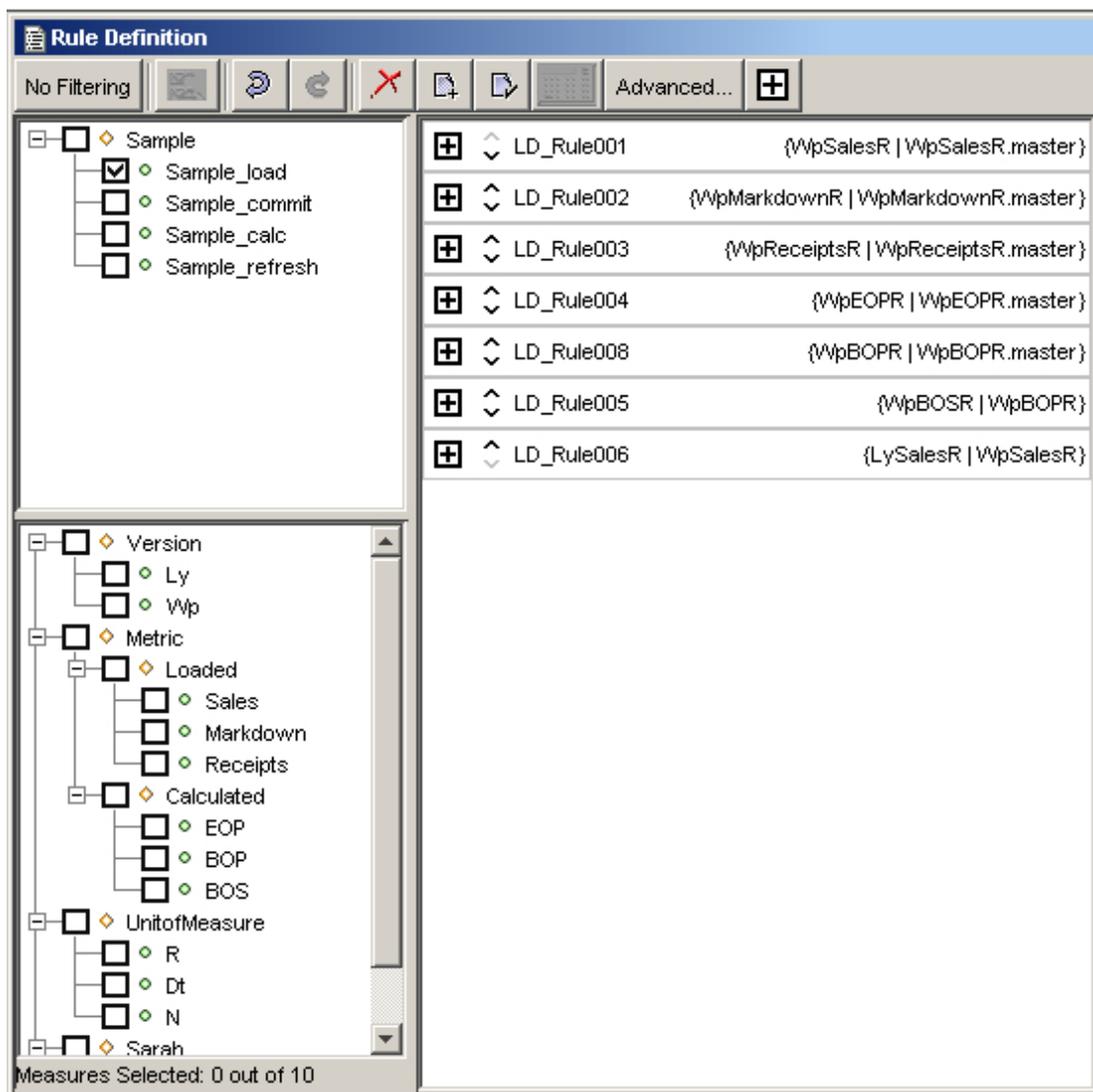
Rule Definitions window

1. In the Rule Definition window, select the rule set where you want to create a new rule group.

2. From the toolbar, click the new [image] button, and select Rule Group. The Add Rule Group window opens.

Add Rule Group window

1. In the Name field, enter the name of the rule group.

     Note: A rule group name can be a maximum of 16 characters.

2. In the Description field, enter a description of the rule.

3. Click **OK** to save and changes and close the window.

Add attributes to a rule group:

1. From the toolbar, click the edit [icon] button, and select Rule Group. The Rename Rule Group window opens.
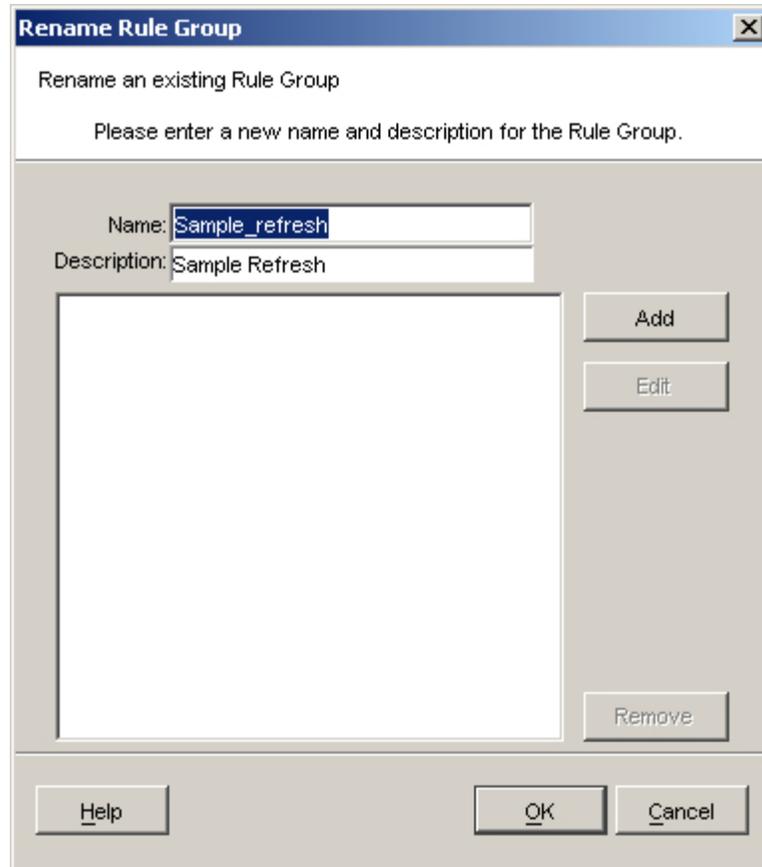
Rename Rule Group window

2. Click **Add**. The Edit Attribute window opens.

3. In the Attribute field, enter the appropriate information.

4. In the Value field, enter the appropriate information.

5. Click **OK** to save and changes and close the window.

Delete a rule group

⇒ **Navigate:** In the Configuration Manager, select ▢ Configuration > ▢ Project > ▢ Solution > 🖹 Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that you want to delete.

   📖   **Note**: You can only delete user-created rule groups. You cannot delete the default load, commit, calc, and refresh rule groups. If you select Delete for one of the default rule groups, all rules and expressions within the rule group will be deleted.

2. From the toolbar, click the delete [icon] button, and select Rule Group.

Edit a rule group

⇒ **Navigate:** In the Configuration Manager, select 🟩 Configuration > 🟦 Project > 🟦 Solution > 📄 Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that you want to edit.

2. From the toolbar, click the edit [icon] button and select Rule Group. The Rename Rule Group window opens.
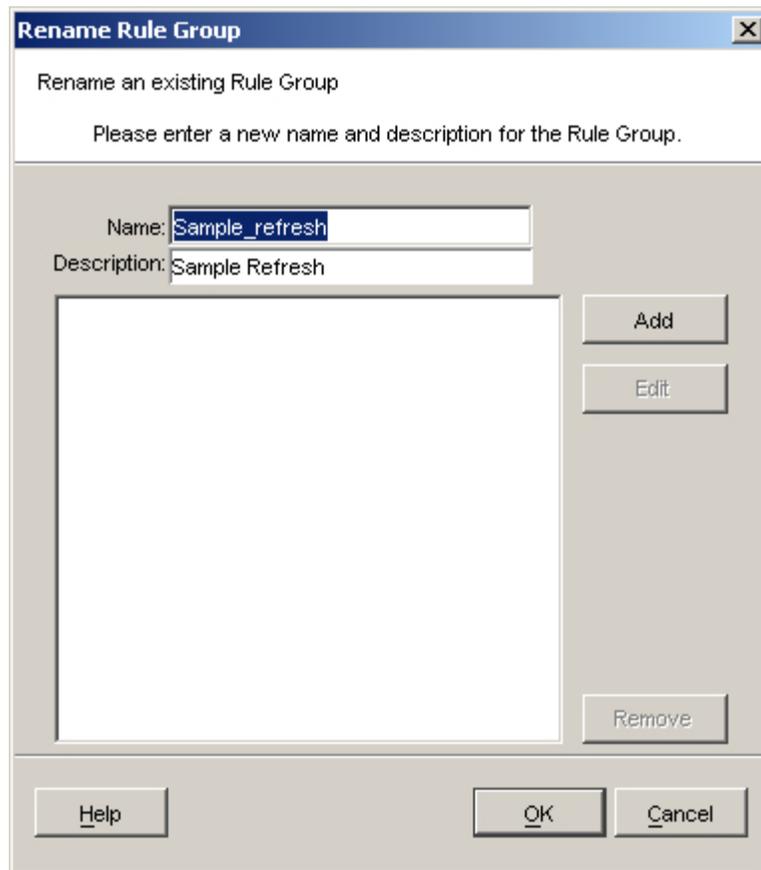
Rename Rule Group window

3. In the Name field, enter the name of the rule group.

     📖   **Note:** A rule group name can be a maximum of 16 characters.

4. In the Description field, enter a description of the rule.

5. To edit attributes for a rule group:

     a. Select the attribute that you want to edit.

     b. Click **Edit**. The Edit Attribute window opens.

     c. Update the information as necessary

   d. Click **OK** to save and changes and close the window.

6. To remove attributes from a rule group:

   a. Select the attribute that you want to delete.

   b. Click **Remove**. The attribute is removed from the display box.

7. Click **OK** to save and changes and close the window.

Copy a rule group

If two groups are similar, it may be beneficial to copy one rule group to prevent having to create a rule group from scratch.

⇒ **Navigate:** In the Configuration Manager, select ▣ Configuration > ▣ Project > ▣ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that you want to copy.

2. From the toolbar, click **Advanced**, and select Copy Rule Group. The Copy Rule Group window opens.
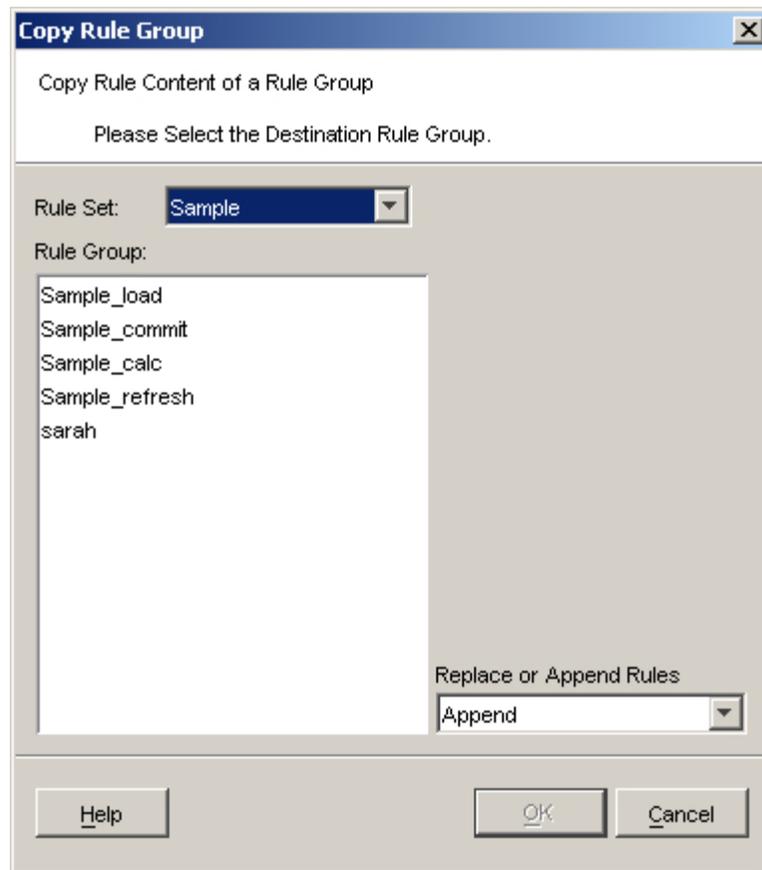
Copy Rule Group window

3. In the Rule Set field, select the destination rule set.

4. In the Rule Group area, select the desired destination Rule Group.

5. In the Replace or Append Rules field, select:

   ▪ **Replace** – to overwrite all rules that already exist in the destination rule group.

   ▪ **Append** – to add to the rules already in the destination rule group.

6. Click **OK**.

Measure validation in the rule tool

If measure content validation has been turned off in the Measure Tool, measure validation must be manually initiated in the rule tool. See "Measure validation within the measure tool."

Click the perform Measure Content Validation button on the toolbar in the Rule Tool.

## Work with rule sets

Overview

A rule set is created along with the following default rule groups: load, commit, calc, and refresh.

The rule group names are prefixed with the name of the rule set followed by an underscore. After the default rule groups are created, you can create additional rule groups as necessary. You can also rename the automatically generated rule groups, but you cannot delete them.
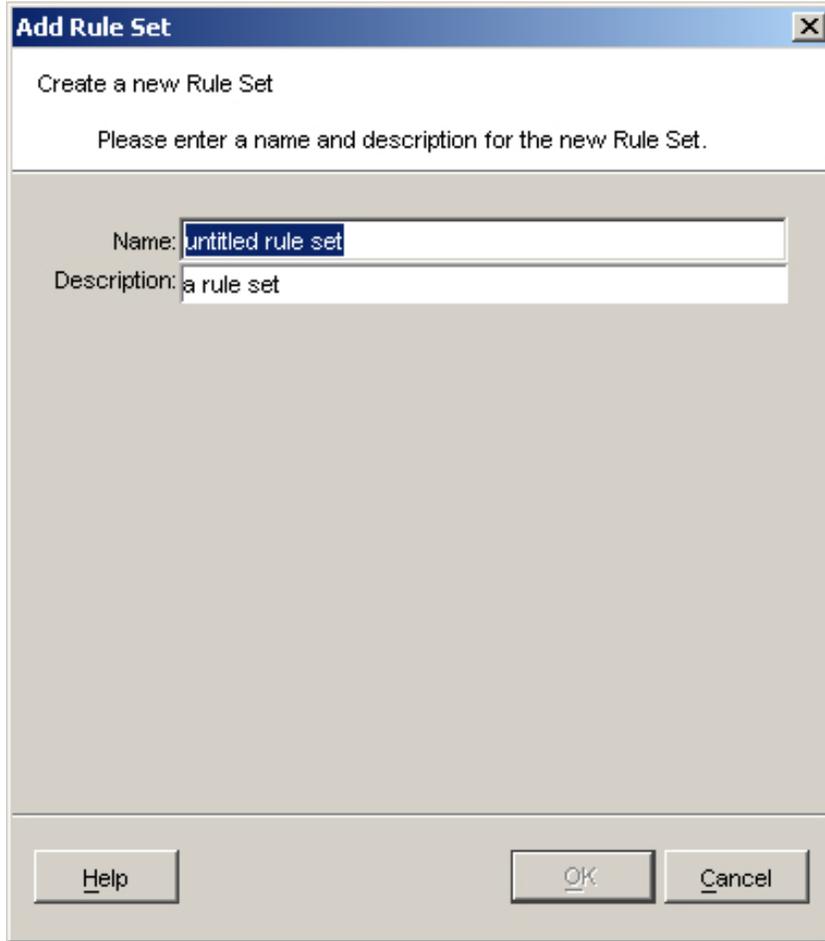
Procedures

Create a rule set

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.
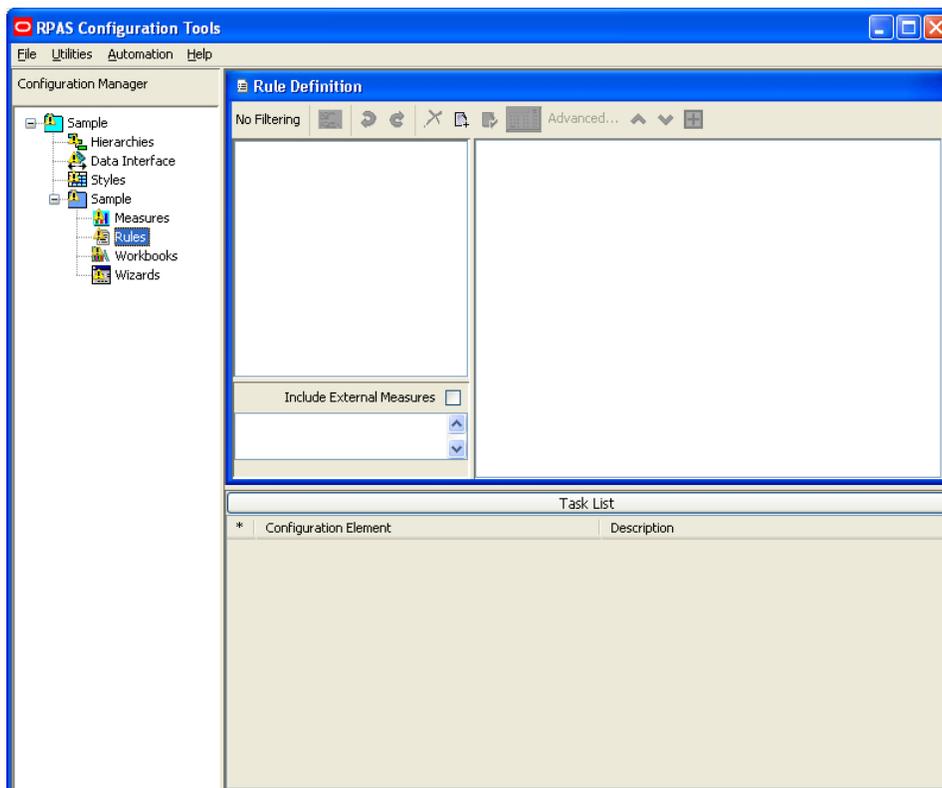
1. From the toolbar, click the new button, and select Rule Set. The Add Rule Set window is displayed.


Add Rule Set window

2. In the Name field, enter the name of the rule set.

    📖   **Note:** A rule set name can be a maximum of 10 characters.

3. In the Description field, enter a description of the rule set.

4. Click **OK** to save and changes and close the window.

OR

1. Right-click in the Rule Definitions window.

2. Select Add > Rule Set.

3. Again, enter the name and description of the rule set.

4. Click OK to save and changes and close the window.

Add attributes to a rule set

1. From the toolbar, click the edit [ ] button, and select Rule Set. The Rename Rule Set window opens.
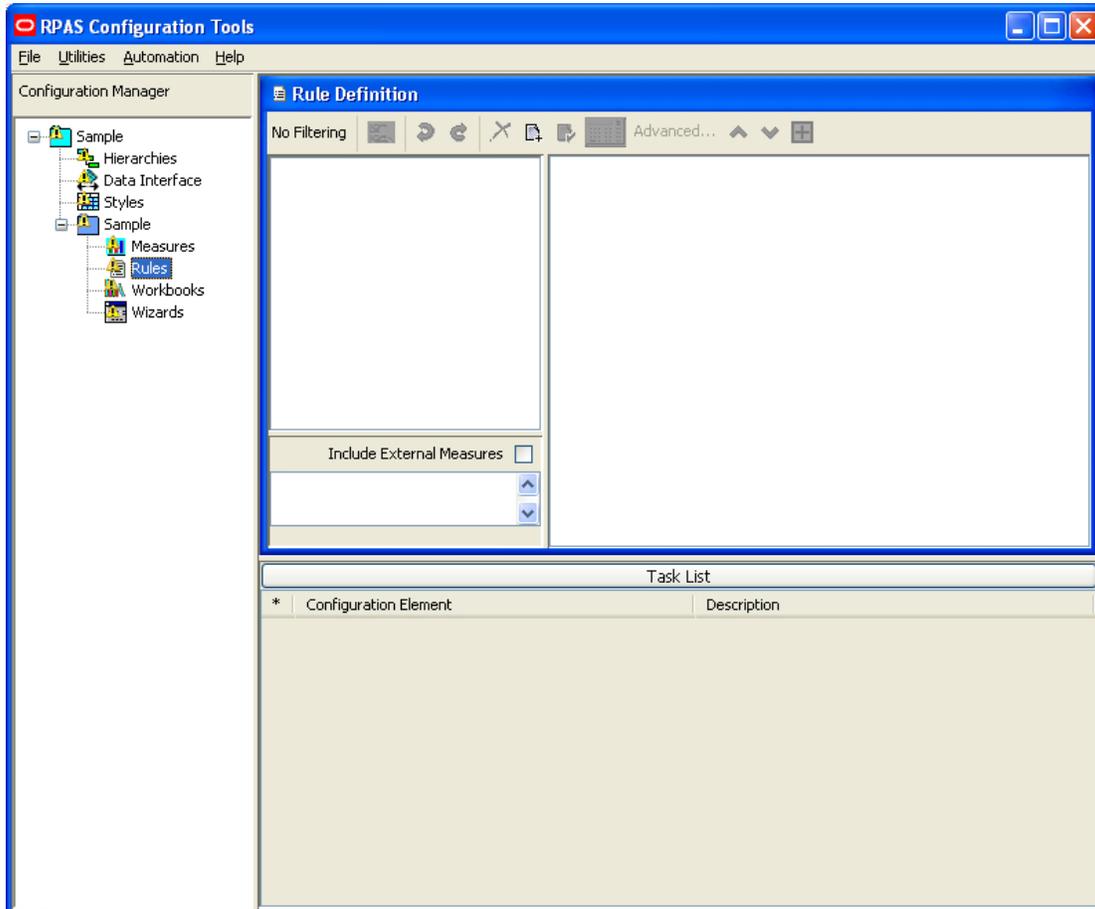

2. Click **Add**. The Edit Attribute window opens.
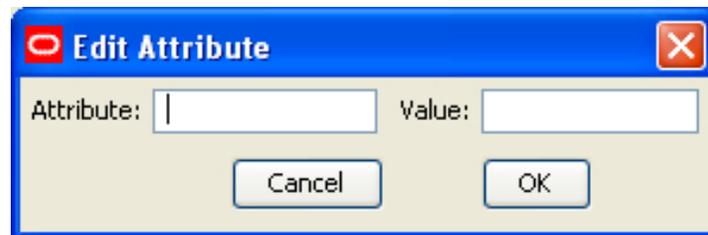
3. In the Attribute field, enter the appropriate information.

4. In the Value field, enter the appropriate information.

5. Click **OK** to save and changes and close the window.

Delete a rule set

⇒ **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > 🗐 Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule set that you want to delete.

2. From the toolbar, click the delete ![delete button] button, and select Rule Set.

   OR

   Right-click in the Rule Definition window, select Delete, and select Rule Set from the drop-down list.

Edit a rule set

⇒ **Navigate:** In the Configuration Manager, select ⬜ Project > ⬜ Solution > 🗒 Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that you want to edit.

2. From the toolbar, click the edit [ ] button, and select Rule Group. The Rename Rule Set window opens.
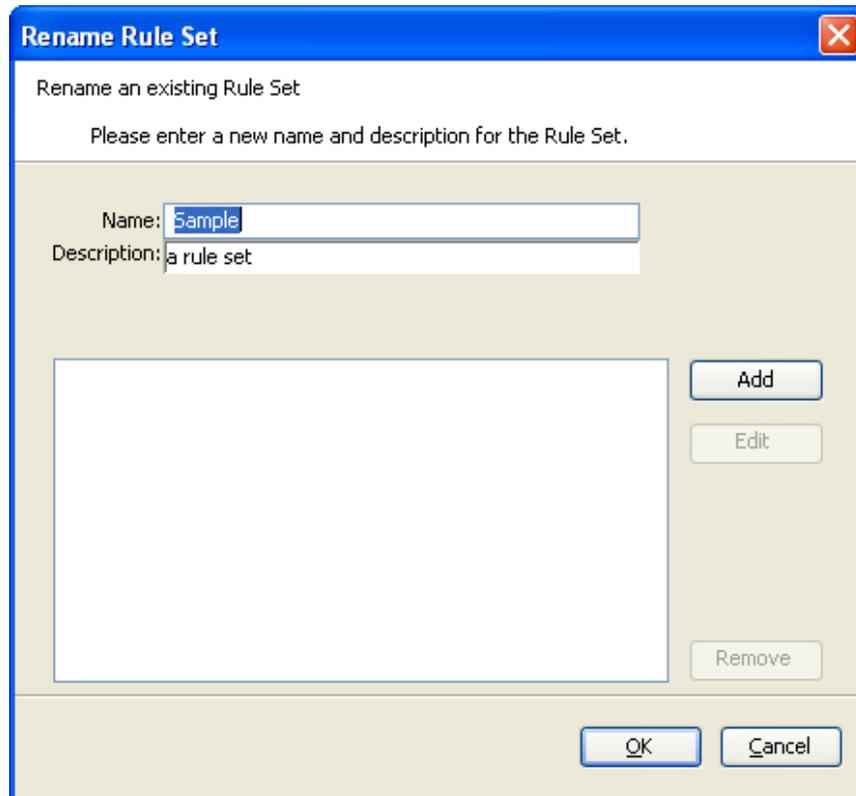
3. In the Name field, enter the name of the rule set.

   📖   **Note:** A rule set name can be a maximum of ten characters.

4. In the Description field, enter a description of the rule.

5. To edit attributes for a rule set:

   a. Select the attribute that you want to edit.

   b. Click **Edit**. The Edit Attribute window opens.

   c. Update the information as necessary

   d. Click **OK** to save and changes and close the window.

6. To remove attributes from a rule set:

   a. Select the attribute that you want to delete.

   b. Click **Remove**. The attribute is removed from the display box.

7. Click **OK** to save and changes and close the window.

Work with rules

Overview

When a rule has multiple expressions, those expressions are given priority sequence to help the calculation engine select a calculation path that follows business priorities. When given a choice, the calculation engine always selects the highest priority expression in the rule that is available to be selected. Considerable care should be taken in the design of models to ensure that appropriate expression priorities are established. The business priority may vary from implementation to implementation, and it may vary from one type of plan to another in the same implementation.

Use great care when you have expressions in a rule that use different measures, because the basic workings of the calculation engine assume that all expressions use the same measures. Also, it is possible to have an expression with multiple LHS measures, and this implies each expression only has a single LHS measure. See the Calculation Engine Users Guide.
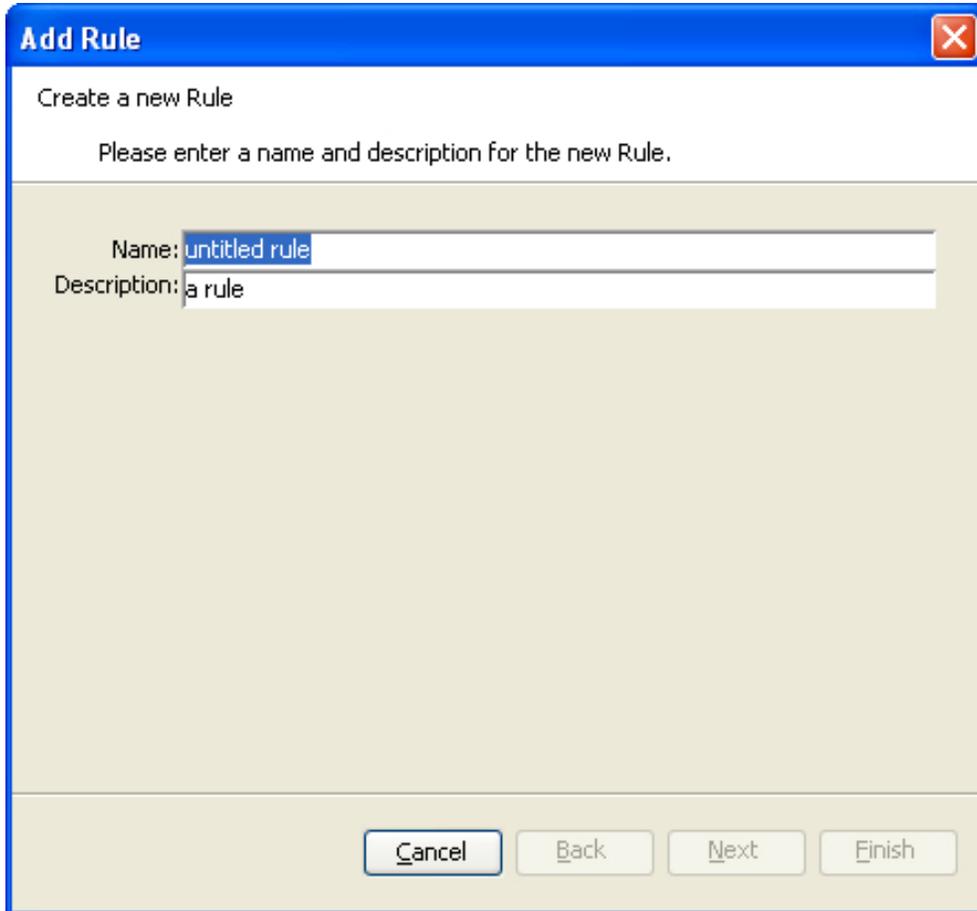
Procedures

Create a rule

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.
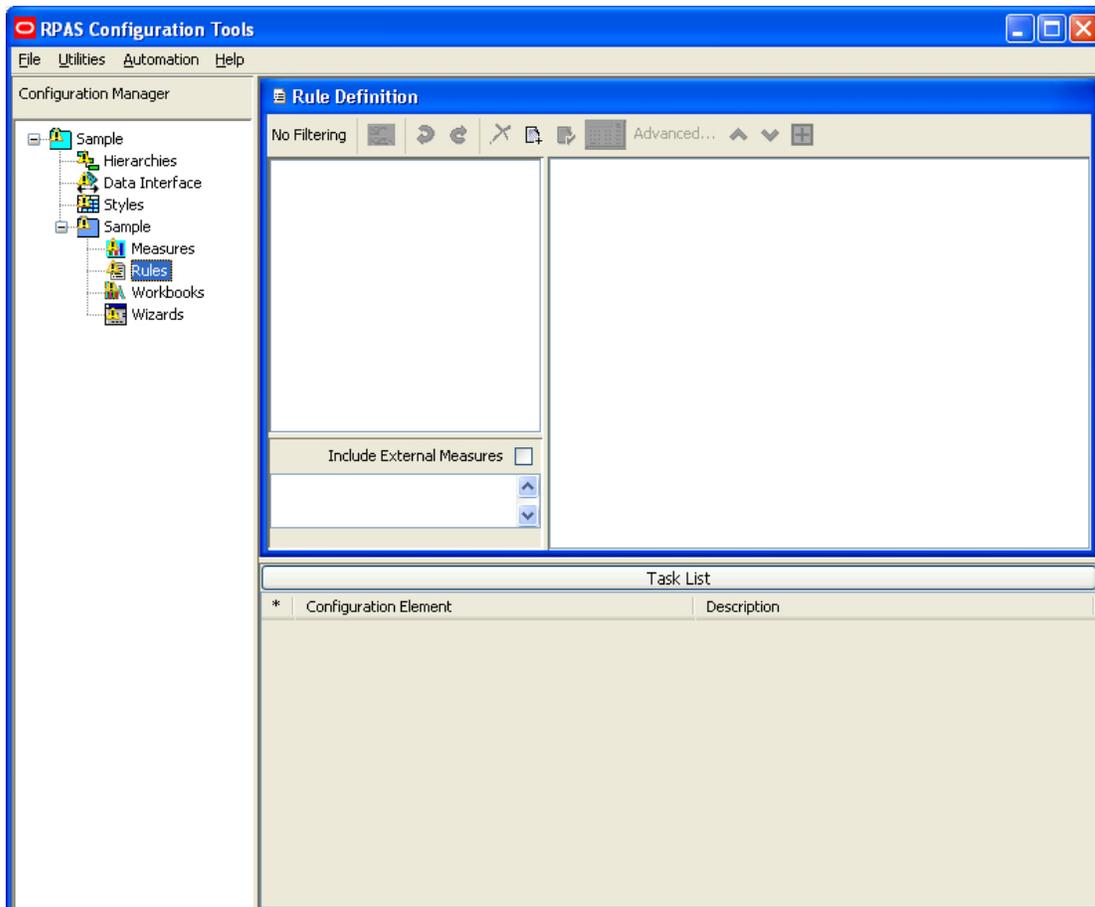
1. In the Rule Definition window, select the rule group where you want to create a rule.

2. From the toolbar, click the new ![button](new button icon) button.

3. Select Rule. The Add Rule window opens.
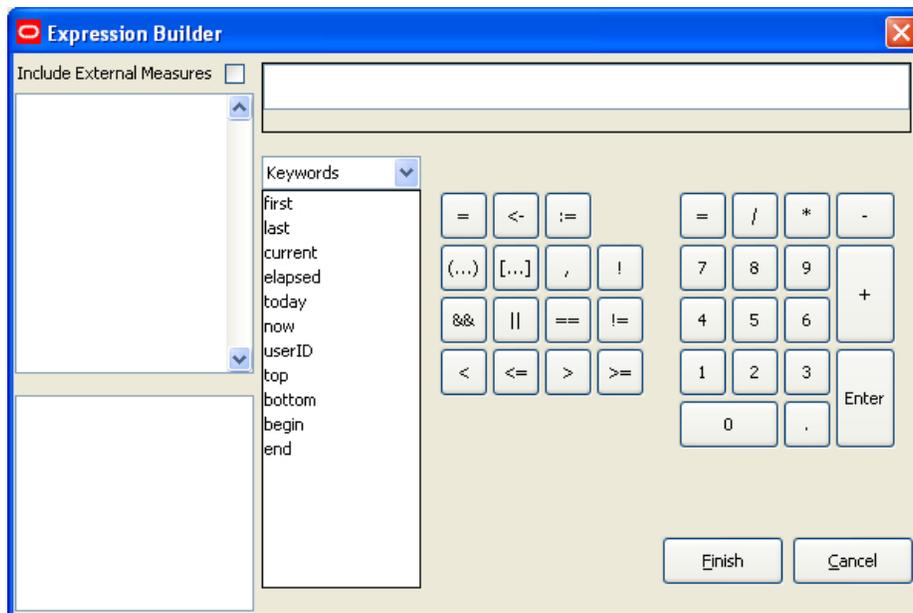

4. In the Name field, enter the name of the rule.

    📖   **Note:** A rule name can be a maximum of 14 characters.

5. In the Description field, enter a description of the rule.

6. Click **Next**. The Expression Builder window of the Add Rule window is displayed.
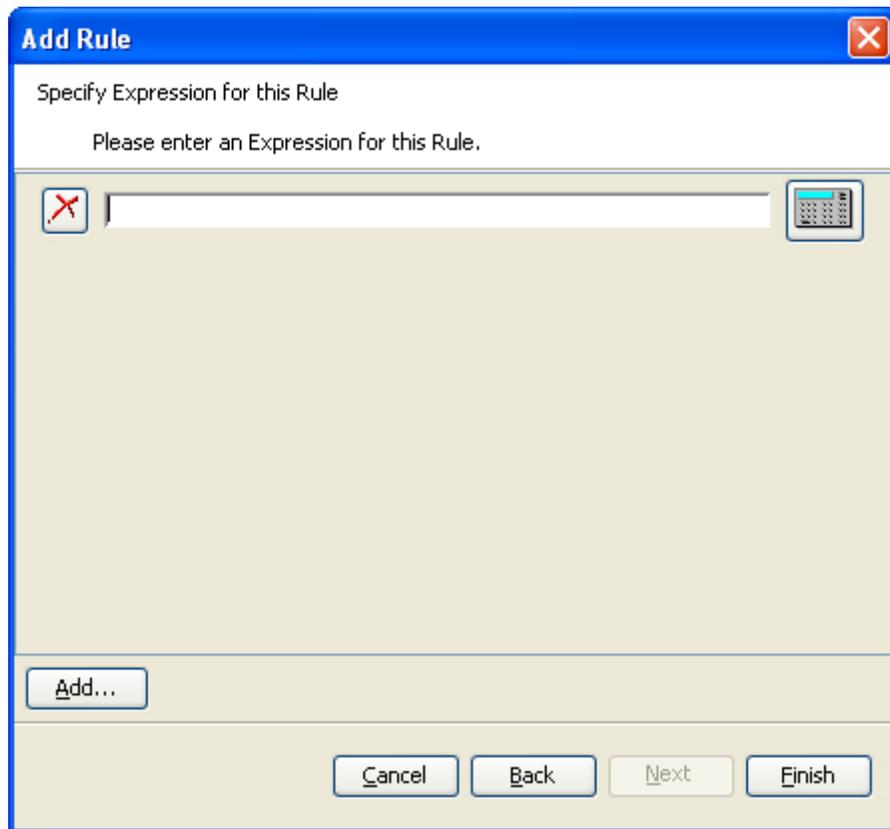
7. Click the expression builder button to the right of the text box. The Expression Builder window opens.

8. In the upper left hand pane, select the measure components. Realized measures are displayed in the lower left hand pane.

     Note: Based on the component selections made, realized measure will be displayed.

9. In the lower left hand pane, double click the measure that you want to use in the expression.

10. From the drop down list, select the functions that you want to use in the expression.

11. Add the appropriate mathematical operators and constants to construct an expression.
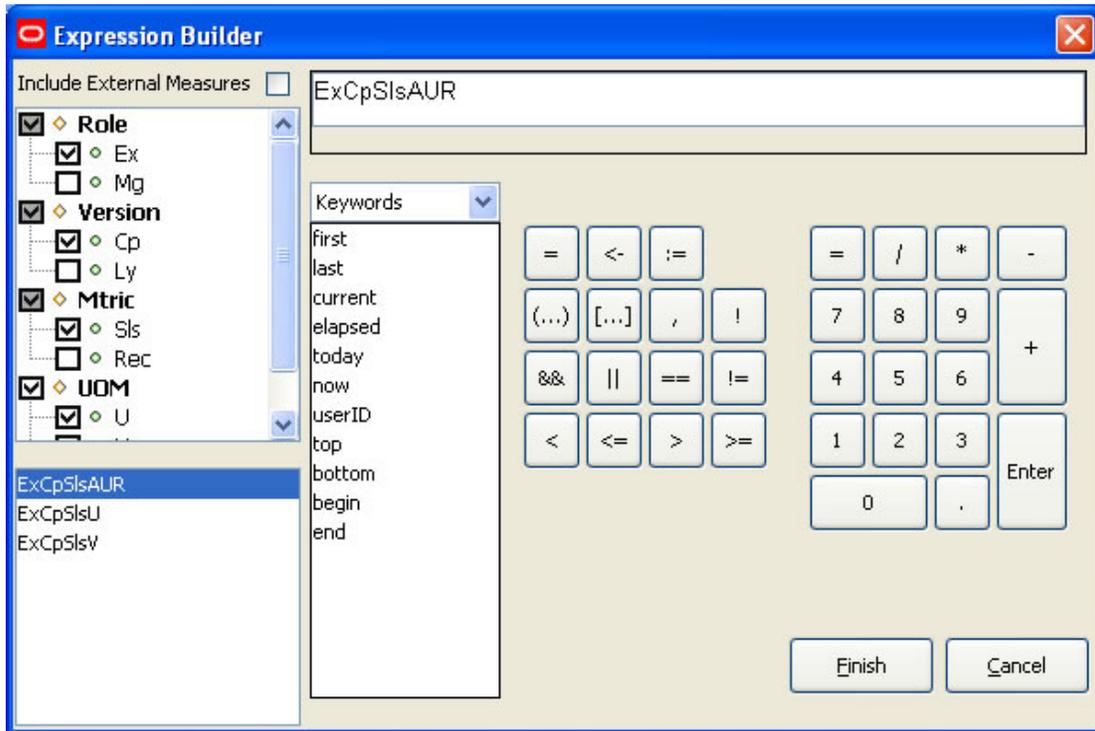
12. Click **Finish** to save any changes and close the window.

     Note: If you have created an invalid expression, a warning message is displayed and the font color is red.

13. To add multiple expressions to a rule, click **Add**.

14. Click **Finish** to save any changes and close the window.

Add attributes to a rule

1. Select the rule that you want to add attributes to.

2. From the toolbar, click the edit [ ] button and select Rule. The Rename Rule window opens.

3. Click **Add**. The Edit Attribute window opens.

4. In the Attribute field, enter the appropriate information.

5. In the Value field, enter the appropriate information.

6. Click **OK** to save and changes and close the window.

Apply a rule pattern

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The
   Rule Definitions window opens in the workspace.
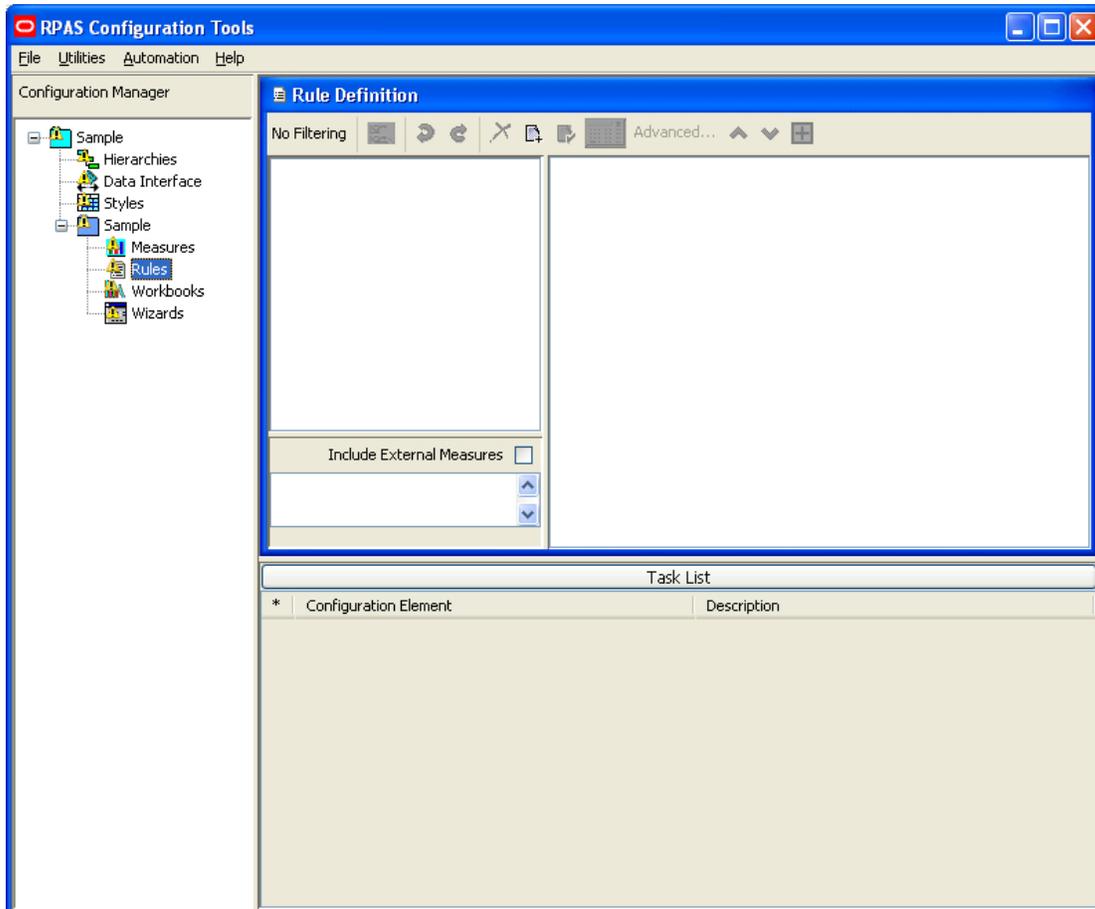
1. In the Rule Definition window, select the rule group that contains the rule to use as the
   pattern basis.

2. From the tool bar, click **Advanced**.

3. Select Apply Pattern.

4. Select the desired rules.

5. Click **OK**.

Delete a rule

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.

Rule Definitions window

1. In the Rule Definition window, select the rule group that contains the rule that you want to delete.

2. Select the rule that you want to delete.

     📖   **Note:** You can select multiple rules by holding the CTRL key as you click the rules to select. A selected rule is indicated by a bold rule name.

3. From the toolbar, click the delete ✗ button.

Edit a rule

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > 🖹 Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that contains the rule that you want to edit.
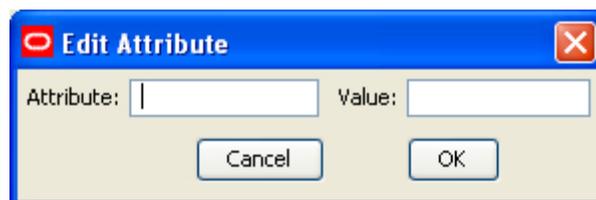
2. From the rule group, select the rule that you want to edit.

3. From the toolbar click the edit ▢ button.
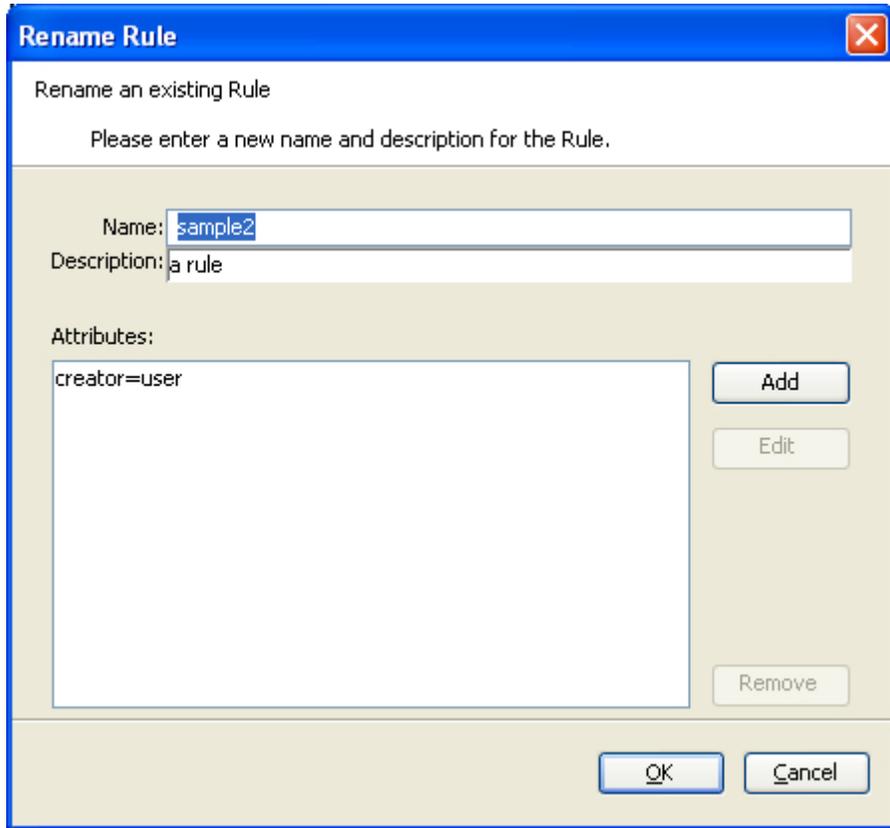
4. Select Rule. The Rename Rule window opens.

5. In the Name field, enter the name of the rule set.

 📖 **Note:** A rule name can be a maximum of 14 characters.

6. In the Description field, enter a description of the rule.

7. To edit attributes for a rule:

   a. Select the attribute that you want to edit.

   b. Click **Edit**. The Edit Attribute window opens.
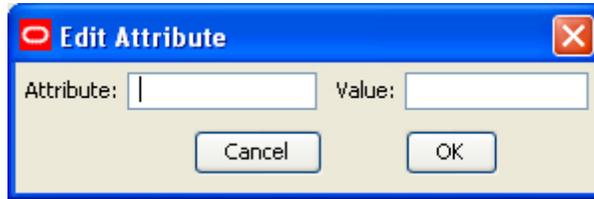
   c. Update the information as necessary

   d. Click **OK** to save and changes and close the window.

8. To remove attributes from a rule:

   a. Select the attribute that you want to delete.

   b. Click **Remove**. The attribute is removed from the display box.

9. Click **OK** to save and changes and close the window.

Rename all rules

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that contains the rules that you want to rename.

2. From the toolbar, click **Advanced**, and select Rename All Rules.

3. In the Prefix field, enter a prefix to be pre-pended to all rule names.

4. Click **OK**. You are prompted to confirm your decision.

   📖 **Note:** All rules in the rule group are renamed with the prefix followed by a 4 digit numeric identifier generated by the rule tool. The rule tool will maintain the order that the rules were in before they were renamed.
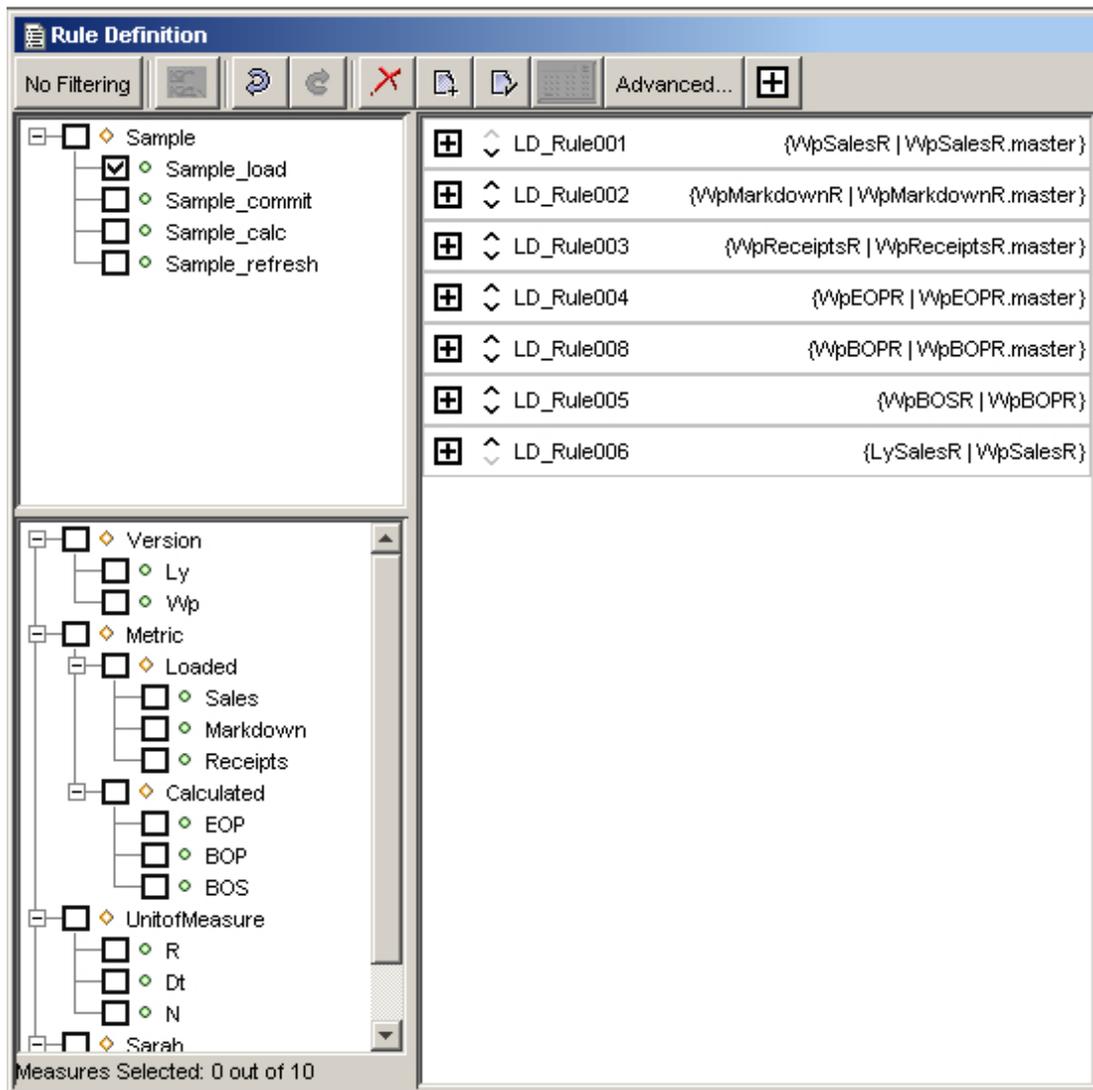
5. Click **Yes**.

Filter rules

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, click the Rule Filtering button.

   📖 **Note:** This is a dynamic button, and the text will change depending on the current filter mode.

2. Select one of the following options:

   ▪ Disable Filtering – All rules are displayed.

   ▪ Filter by Measure – Works in conjunction with the measure components box in the bottom left corner of the screen. Rules are filtered to show those whose measures conform to the selected component scheme.

   ▪ Filter by Size – Rules are filtered to show those with more than one expression.

   ▪ Filter by Validity – Only invalid rules are displayed.

Rule ordering

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.

1. In the Rule Definition window, select the rule group that contains the rule you want to move.

   a. Click the down arrow to move the expression down the list.

   b. Click the up arrow to move the expression up the list.

Auto generate load and commit rules

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▤ Rules. The Rule Definitions window opens in the workspace.

   📖 **Note:** Rules for the load and commit rule groups can be auto-generated based on the calc rule group. The measures referenced in the calc rule group are assumed to be all of the measures in a workbook. A load and commit rule is generated for all of those measures that have a database allocated, so they are physically stored.

1. Select the calc rule group to use as a basis.

2. From the toolbar, click **Advanced**.

3. Select Generate Load Rules or Generate Calc Rules. You are prompted to confirm your decision.

📖 **Note:** Measures to be used in the load rule group must have a database assigned in the Measure tool.

4. Click **Yes**. This will also clear out the undo stack.

Copy selected rules

⇒ **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > 📄 Rules. The Rule Definitions window opens in the workspace.

1. Select the Rule Group that contains the rules you want to copy.

   📖   **Note:** To select multiple rules, hold down the CTRL key and click the rules to select. A selected rule is indicated by a bold rule name.

2. From the toolbar, click **Advanced**.

3. Select Copy Selected Rules. The Copy Selected Rules window opens.
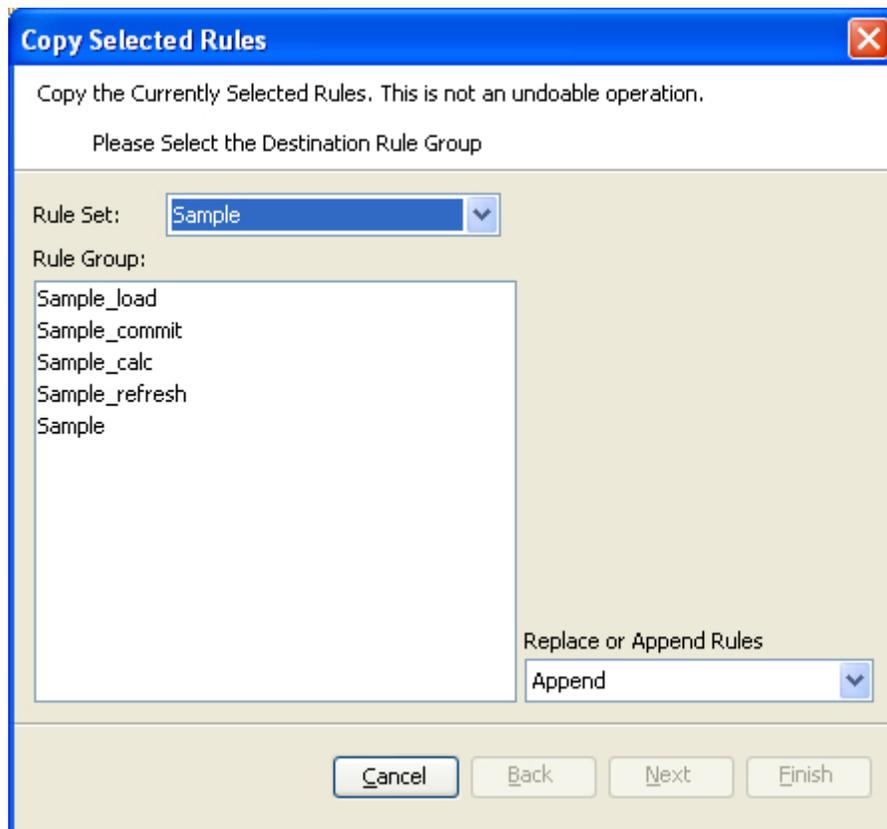
4. In the Rule Set field, select the destination rule set.

5. In the Rule Group area, select the desired destination Rule Group.

6. In the Replace or Append Rules field, select:

   ▪ **Replace** – to overwrite all rules that already exist in the destination rule group.

   ▪ **Append** – to add to the rules already in the destination rule group.

7. Click **Next**. The second window of the Copy Selected Rules window opens.
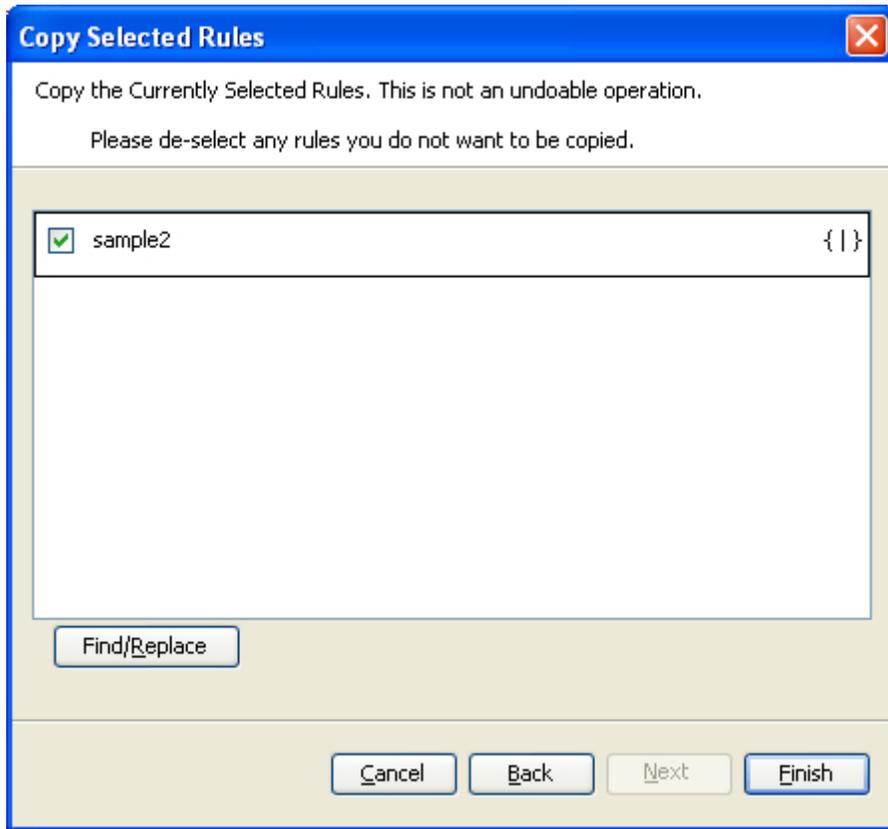
Find and replace measures in the copied rules:

    📖   **Note:** The ability to find/replace in the copy rules is a very powerful and useful feature. It allows the builder to build a collection of rules and "clone" them to a very similar collection of rules. For example, a collection of rules that calculate a series of variances with one version can be cloned to produce rules that calculate a series of variances with another version.

1. Click the Find/Replace button. The Find/Replace window is displayed.
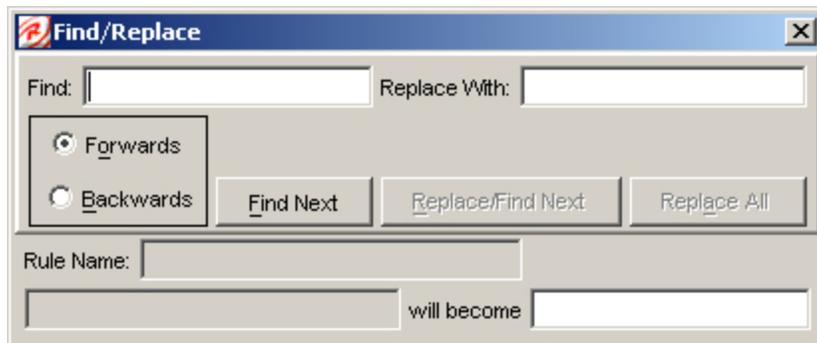
2. In the Find field enter the portion of the measure that you want to replace.

    📖   **Note:** The Find function is case sensitive.

3. In the Replace With field, enter the string to replace the portion of the measure name that you are finding.

4. Select Forwards to search the rules in order

   OR

   Select Backwards to search the rules in reverse order.

5. Click **Find Next**. The first candidate measure to be replaced will be displayed in the bottom left field.

6. Click **Replace/Find Next** to replace the current candidate measure and display the next candidate measure.

   OR

   Click **Replace All** to replace all instances of the measure.

   OR

   Click Find Next to skip over that occurrence of the portion of the measure name, and go onto the next one.

7. Click the close ⊠ button close the Find/Replace box.

8. Click **Finish**.

Workbooks

Work with workbook tabs

Overview

After you create a workbook, you must add tabs to the workbook.

Procedures

Create a workbook tab

⇒ **Navigate:** In the Configuration Manager, select 🟦 Project > 🟦 Solution > 📚 Workbooks. The Workbook Designer window opens in the workspace.
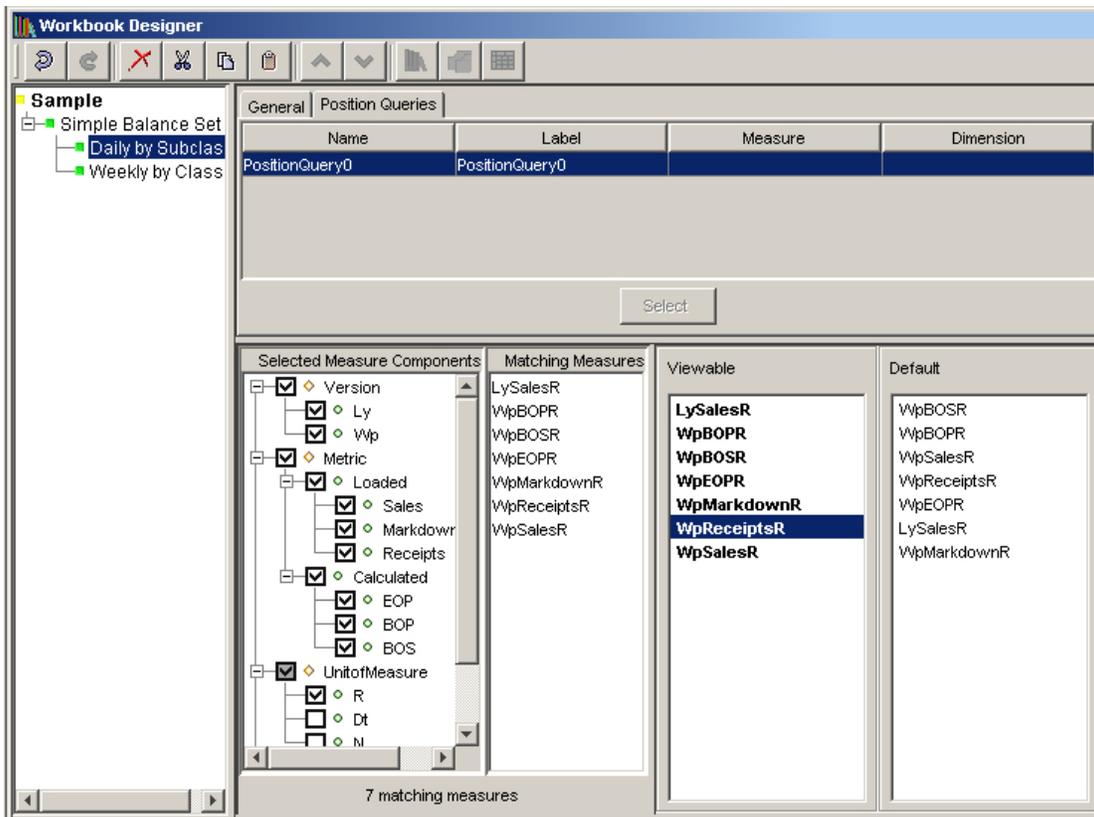
1. Select the workbook in which to create a new workbook tab.

2. Click the new workbook tab 🗂 button.

3. In the RPAS name field, enter RPAS internal name of the workbook tab.

4. In the User Label field, enter a description of the workbook tab that users will see.

Edit workbook tab properties

⇒ **Navigate:** In the Configuration Manager, select 📁 Project > 📁 Solution > 📚 Workbooks. The Workbook Designer window opens in the workspace.

1. Select the workbook tab whose properties you want to edit.

2. In the General tab, type the RPAS Name and the User Label.

3. To change the order of the tabs, select the up ⬆ button or down ⬇ button as necessary.

Remove a workbook tab

⇒ **Navigate:** In the Configuration Manager, select 📁 Project > 📁 Solution > 📚 Workbooks. The Workbook Designer window opens in the workspace.

1. Select the workbook tab that you want to remove.

2. From the toolbar, click the delete ❌ button. You are prompted to confirm your decision.

3. Click **Yes**. Associated worksheets are removed.

Work with workbooks

Overview

Take the time to design a well-planned workbook. Workbooks should be laid out in a logical format and should be easy to navigate. When you are building a workbook, think about how the workbook will be used. Understand what end users will need to access most and the business process flow. Most likely this information should be contained in the first workbook tab and worksheet.

Think about the most logical flow for the wizards. Names of workbook tabs and worksheets should be intuitive to an end user. RPAS names (not wizards) need to be unique across all workbook components in a domain. This includes workbook, tab, worksheet, wizard, and custom menu RPAS names. Workbook tabs can be used to organize similar worksheets together.

If you are setting properties for the first time for a workbook, you must create a workbook tab, create a worksheet, and assign measures to worksheets to complete the template.

- **Use Default Rules** – select the check box to use the default rules (Load, Commit, Calc, and Refresh)associated with the rule set:

    a. In the Load Rules, Commit Rules, and Calc Rules; select the rule group that you want to apply for each rule type.

    b. For the Refresh group, click once in the field. The Select and Order Multiple Rule Groups window opens.
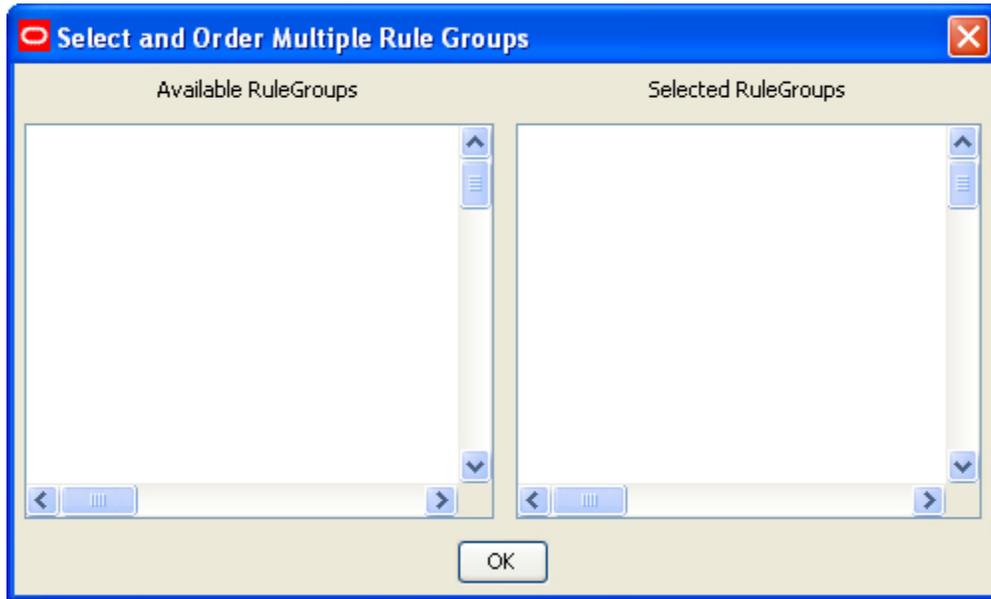
Within this window, you will specify which group can be used to refresh the workbook. You can refresh part of the data within a workbook without affecting the rest of the data. The order that they appear is the order that they will appear within the workbook.

    c. In the Available Rule Groups column, select the rule group to add; and drag it to the Selected Rule Groups column.

    d. Click **OK** to save any changes and close the window.

3. Define the axis layout of the worksheets in the workbook.

    a. Click the X-axis, Y-axis, Z-axis, or Unassigned field. The Axis window opens.
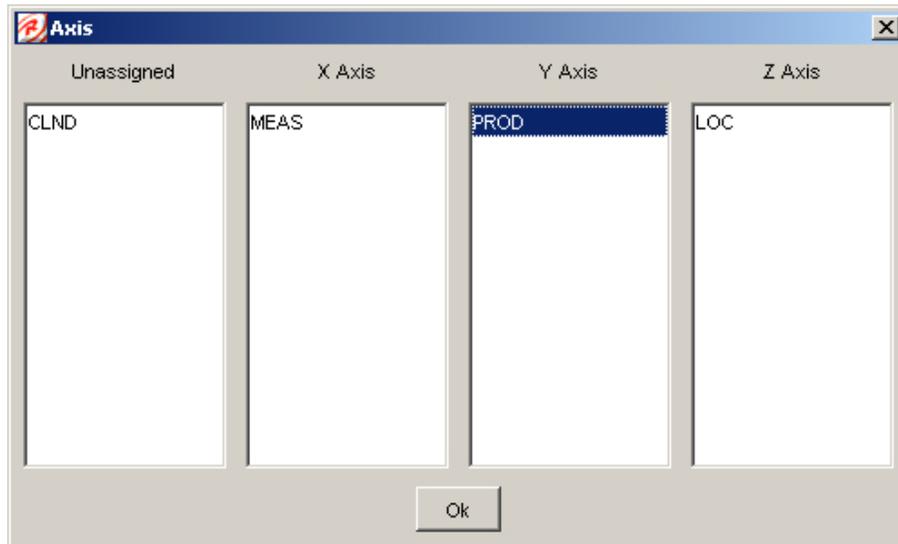
Axis window

    b. Drag the hierarchies to the appropriate axis column.

    c. Click **OK** to save any changes and close the window.

4. Determine the type of wizard to use for the workbook template.

Standard

    a. Click on the standard wizard field. The Standard Wizard window opens.
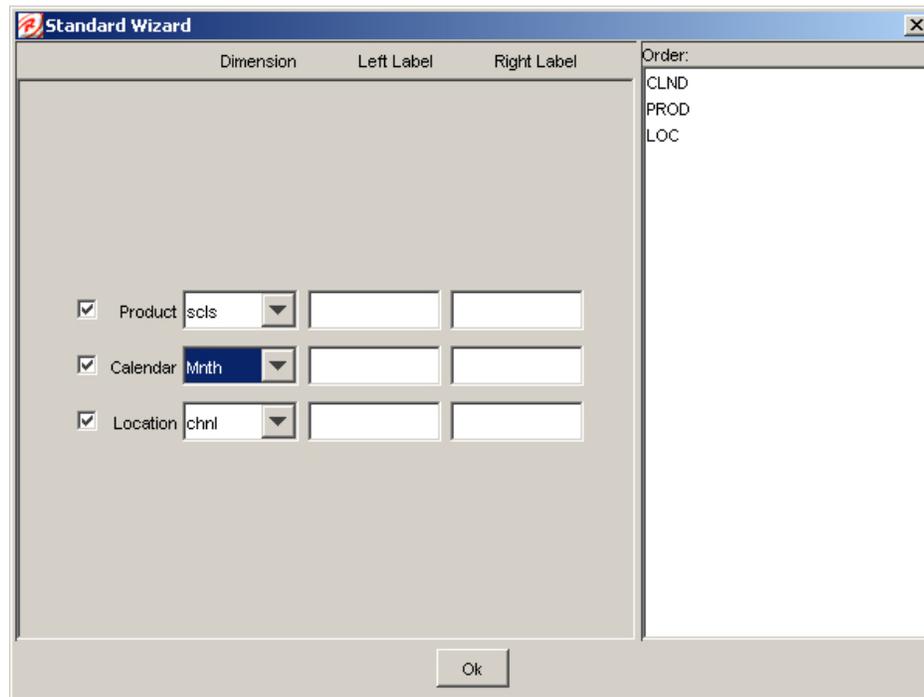
Standard Wizard window

    📖   **Note:** Hierarchies that are used in the base intersections for the measures used in the rule set assigned to the workbook will be displayed.

    b. Select the desired dimension from each hierarchy. This will specify the base intersection from which a workbook can be built in the RPAS client.

    c. Enter the left and right labels for each hierarchy. These labels will be displayed during the workbook wizard process.

    d. Adjust the order of the hierarchies as necessary by dragging them. This will be the order of the wizards during the workbook build process.

    e. Click **OK** to save any changes and close the window.

Custom

    a. In the User Custom Wizard field, select the check box.

    b. In the Custom Wizard field, select the custom wizard that you want use to build the workbook.

Custom Menus tab

1. Click on the Custom Menus tab.

    📖   **Note:** This tab is optional and is used to create a menu-driven process within the workbook using rule group transitioning. For example, custom menus are used to define an approval process.

2. In the Menu Label field, enter the name of the menu that will be displayed in the RPAS client.

3. Right click in the table area, and select Add.

4. Enter the following information:

    ▪ **Label** – The label that will be displayed in the menu

    📖   **Note:** Duplicate label names are not allowed.

    ▪ **Function** – Defaults to RuleGroupProcessor and cannot be changed

    ▪ **Arguments** – The rule groups to be used in the process, separated by commas

Hier Mods tab

1. Click on the Hier Mods tab.

    📖   **Note:** This tab is optional. The Hier Mods (Hierarchy Modifications) function is used to define the workbook hierarchy structure if different from what is defined using the hierarchy tool. For example, if the Product Hierarchy Dimensions are configured as Dept, Class, Style, SKU, but in the Workbook only Dept and Class need to be visible, Hier Mods would be used.

2. For each hierarchy displayed, enter the dimensions separated by commas.

Dynamic Hierarchies tab

1. Click on the Dynamic Hierarchies tab:

       📖   **Note:** This tab is optional and is used to configure a hierarchy whose member positions are dynamic and not constant. For example, if Cluster is an alternate hierarchy off of Store in the Location Hierarchy, and the stores that are included in the cluster can differ by product.

2. Right click in the table area and select Add.

   - **Name** – The name of the dynamic hierarchy. Duplicate names are not allowed.

   - **Measure** – Click **Select Measure** to get a list of the measures used in the workbook. Double click to select the desired measure.

   - **Label Measure** – Enter the label measure.

   - **Measure Hier** – Enter the measure hierarchy.

   - **Measure Dim** – Enter the measure dimension.

   - **Hier** – Enter the hierarchy.

   - **Dim** – Enter the dimension.

   - **Modified Dim** – Enter the modified dimension.

       📖   **Note:** Measures that have been unrealized, renamed, or realized are displayed in red.

Measures tab

1. Click on the Measures tab:

       📖   **Note:** This tab is optional and allows you to override the following measure information at the workbook level: Label, Description, Base State, Agg State, UI Type, Range, Loadrange, and Loadrange Measure.

2. In the Measures tab, right-click in the table area, and select Add.

   a. Click **Select Measure** to get a list of the measures used in the workbook.

   b. Double click to select the desired measure.

   c. Specify the desired property information.

   d. To apply values to the loadRange and LoadRange measure properties:

      i. Right click in the table area and select Set Dynamic Picklist.

      ii. Doubleclick in the LoadRange field. The LoadRange window opens.

      iii. Enter information in the appropriate fields as necessary.

      iv. Click **OK** to save any changes and close the window.

Extended measures tab

1. Click on the Extended Measures tab:

    📖   **Note:** This tab is optional and allows you to configure extended measures that represent different usages of the original base measure. These extended measures include using different aggregation methods for the same base measure, or participation measures, such as absolute and relative percent to parent measures. These extended measures can then be added onto worksheet profiles to be viewed in the RPAS client.

2. Right click in the table area, and select Add.

    📖   **Note:** If you are setting properties for the first time for a workbook, the next steps are to create a workbook tab, create a worksheet, and assign measures to worksheets.

Edit workbook properties

⇒ **Navigate:** In the Configuration Manager, select 📁 Project > 📁 Solution > 📚 Workbooks. The Workbook Designer window opens in the workspace.

Click on the tab you want to edit.

1. Update the information as appropriate.

2. To remove information from any of the tables:

    a.   Select the field.

    b.   Right click and select Remove.

Remove a workbook

1. Select the workbook that you want to remove.

2. From the toolbar, click the delete ✖ button. You are prompted to confirm your decision.

3. Click **Yes**. Associated worksheets and tabs are removed.

Work with worksheets

Overview

Measures and worksheets

There are four different categories of measures that are used on worksheets: default profile, viewable profile, hidden, and extended.

Default profile measures

The default profile contains the list of measures that will initially be displayed for this worksheet in the RPAS client. There must be at least one measure on the default profile or the worksheet will be invalid in the Tools.

Viewable profile measures

The viewable profile contains the list of measures that are not initially displayed, but that the RPAS client user can add to be displayed in their view of the worksheet by using the Show/Hide functionality within the RPAS client.

Hidden measures

Hidden measures are those that are used in the rule set assigned to a workbook, but that are not assigned to any of the profiles in any of the worksheets contained in that workbook. This might include measures that are used purely for calculation purposes and would have no usefulness to the RPAS client user.

Extended measures

Extended measures, which represent different usages of the original base measures, can be added to the default or viewable worksheet profile. You can add extended measures that are aggregated based on different aggregation methods. The aggregation methods available for selection are based on the Allowed Aggs of the base measure. The same base measure can have multiple extended measures based on different aggregation methods.

You can also add extended measures that represent the relative and absolute percent to parent contributions. The same base measure can have multiple extended measures based on different selections for relative and absolute percent to parent contributions.

Procedures

Create a worksheet

⇒ **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > ⓘ Workbooks. The Workbook Designer window opens in the workspace.
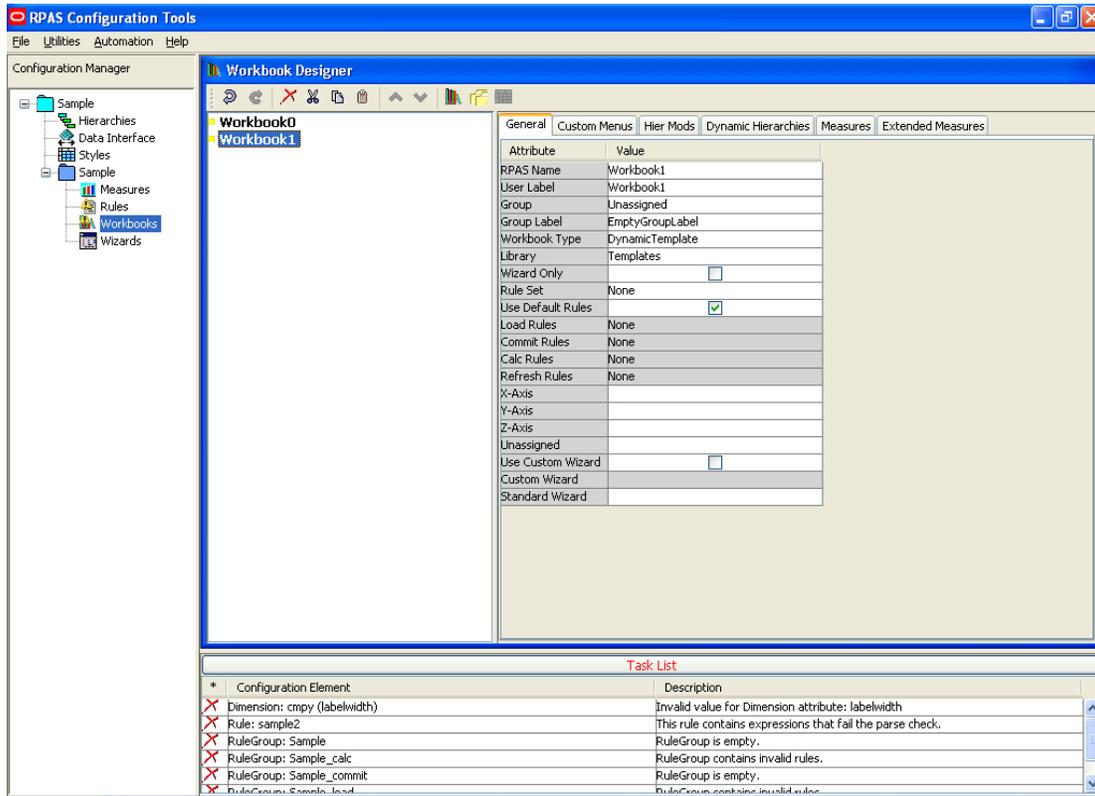
1. Right-click on the workbook title, and select new workbook tab.

2. Click the new worksheet ⊞ button.

General tab

1. In the RPAS Name field, enter the RPAS internal name of the workbook.

2. In the User Label field, enter a description of the workbook that users will see.

3. Define the axis layout of the worksheets in the workbook.

    a. Click the X-axis, Y-axis, Z-axis, or Unassigned field. The Axis window opens.
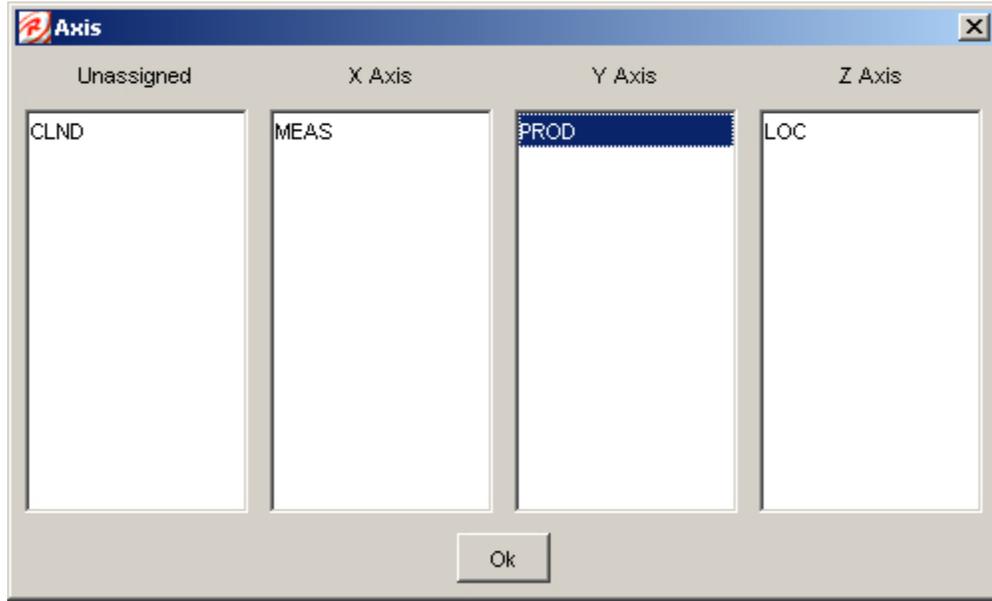
Axis window

    b. Drag the hierarchies to the appropriate axis column.

    c. Click **OK** to save any changes and close the window.

4. Click in the Base Intersection field. The Base Intersection window opens.

5. Select the dimension for each hierarchy.

6. Click **OK**.

Position Queries tab

1. Click on the Position Queries tab.

    📖   **Note:** This tab is optional, and it allows you to view positions in one dimension based on the current position of the dimension that is in the slice area of the worksheet. The position query is a Boolean measure based on the dimension being queried.

2. Right click in the table area, and select Add.

3. Enter the following information:

   ▪ **Name** – The name of that will be displayed in the RPAS client.

   ▪ **Label** – The label that will be displayed in the menu.

       📖   **Note:** Duplicate label names are not allowed.

   ▪ **Measure** – Click Select Measure to get a list of the measures used in the workbook. Double click to select the desired measure.

   ▪ **Dimension** – Select a dimension from the list of dimensions for the measure you selected.

Add measures to a worksheet

⇒ **Navigate:** In the Configuration Manager, select 🟦 Project > 🟦 Solution > 📚 Workbooks. The Workbook Designer window opens in the workspace.

1. In the Select Measure components column, select the check boxes next to the Measure components. The matching measures will appear in the Matching Measures column as the components are selected. The only realized measures that will be displayed are those used in the rule set that is assigned to the selected workbook.

2. Select the measures to add from the Matching Measures column.

3. Drag the measure to the Viewable or Default column.

       📖   **Note:** If you add a measure to the Default column, it will also be added to the Viewable column.

Edit worksheet properties

⇒ **Navigate:** In the Configuration Manager, select 🟦 Project > 🟦 Solution > 📚 Workbooks. The Workbook Designer window opens in the workspace.

1. Click on the tab that you want to edit.

2. Update the information as appropriate.

3. To remove information from the Position Queries table:

   a. Select the field.

   b. Right click and select Remove.

Remove a worksheet

1. Select the worksheet that you want to remove.

2. From the toolbar, click the delete ❎ button. You are prompted to confirm your decision.

3. Click **Yes**.

Workbooks

Overview

A workbook is the easily viewed, easily manipulated multidimensional framework used to perform specific business functions in the configured solution. To present data, a workbook can contain any number of multidimensional spreadsheets, called worksheets, as well as graphical charts and related reports. All of these components work together to facilitate viewing and analysis of business functions.

The workbook designer allows you to create, select, and integrate the various components of a workbook template, which is a pre-designed workbook that is formatted for RPAS users to view and manipulate date. It contains workbook tabs, worksheets, rule groups, wizards, and workflow processes.

Workbook tabs

A workbook tab is a major subdivision of a workbook. Each workbook contain at least one workbook tab by default, but additional tabs can be added for organizing workbooks to support business needs. The workbook designer allows users to define and name tabs and to specify their order in the workbook.

Worksheets

Worksheets are multidimensional spreadsheets that are used to display workbook-specific information. Workbooks can include one or many worksheets. Worksheets can present data in the form of numbers in a grid, or the numeric data values can be converted to a graphical chart.

Rule Groups

Rule groups are a collection of rules that are treated as a unit by the calculation engine. The rules in the rule group must be considered together to satisfy the calculation requirements for a specific business process. The sequence of rules in a rule group determines the calculation sequence unless the sequence is forced.

Wizards

RPAS uses a series of wizards to obtain information for you to build a workbook that contains a subset of the entire data available in the system. The main purpose of a wizard is to allow you to make choices regarding the shape of the workbook. The workbook designer allows you to specify which wizards will be used to build the workbooks.

Wizards

Overview

This section describes the tasks that you can perform by using the wizard designer. The wizard designer allows you to define the graphical layout for custom wizards.

Procedures

Create a wizard group

⇒ **Navigate:** In the Configuration Manager, select ▢ Project > ▢ Solution > ▦ Wizards. The Wizard Designer window opens in the workspace.
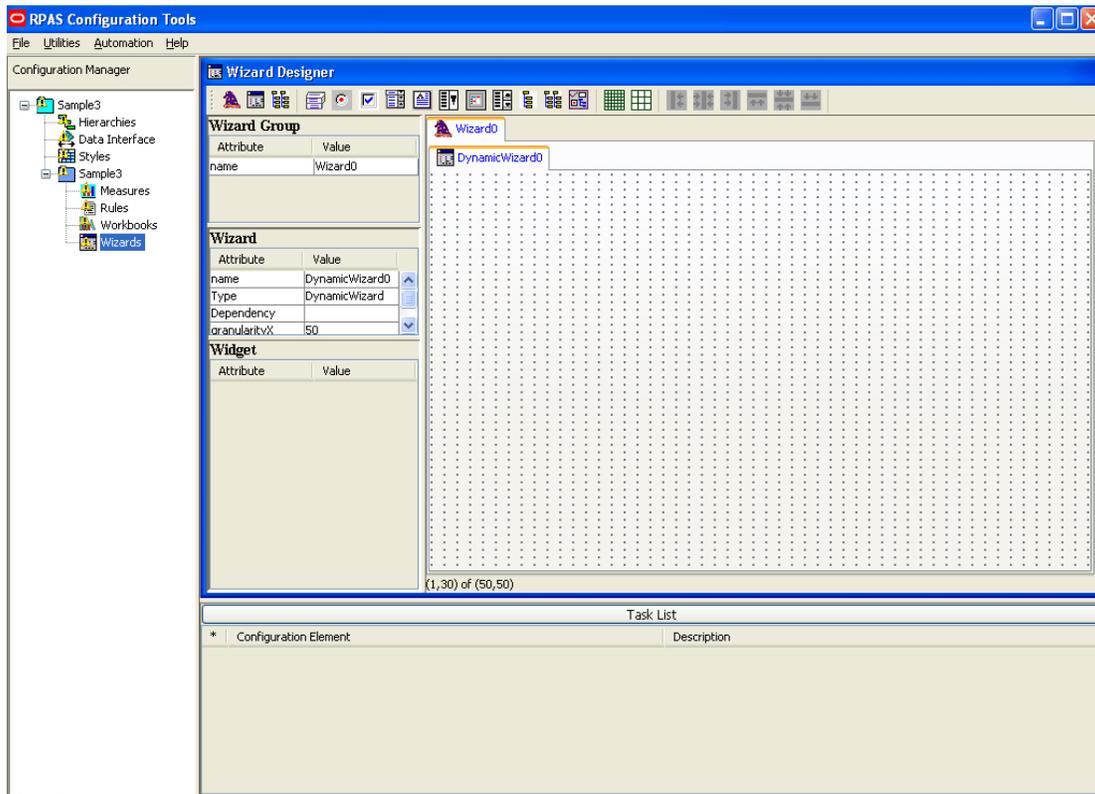
1. In the Wizard Designer window, click the new ▦ wizard button.

2. In the Wizard Group area, click in the Value field, and enter the name of the Wizard Group.

Create a wizard page

1. Click the button to add the appropriate control:

Button	Function
	Allows you to create a new dynamic wizard.
	Allows to create a two tree page.
	Allows you to create a text field that cannot be edited.
	Allows you to create a field from which you can make one choice from several options.
	Allows you to create a field from which you can make multiple choices from several options.
	Allows you to insert a field from which you can select the entered choices. The choice you select is displayed in the text box.
	Allows you to insert a field in which you can enter free-form text.
	Allows you to insert a list box that contains a list of items from which you can select.
	Allows you to create a labeled area in the wizard where you can group other wizard elements.
	Allows you to insert a hierarchy into the wizard.
	Allows you to insert a control that consists of two related tree controls. The left tree control is typically used to display all of the possible selections that can be made from a hierarchy structure. The right tree only displays only those elements of the left tree that are selected.
	Allows you to insert a generic object in the wizard.

2. Click on the wizard page grid to place the selected control on the page.
3. If necessary, drag the control to the appropriate place on the grid to reposition it.

4. In the Widget area, enter the following information:

    ▪   **name** – The RPAS internal name of the control.

    ▪   **type** – The control type.

    ▪   **text** – The text to be displayed on the control label.

    ▪   **align** – The same as the style attribute. Valid values are left, right, center, multiline, and flip depending on the type of widget you are creating.

    ▪   **func** – Indicates whether the widget will be dynamic or static. These are the only valid values for this attribute.

    ▪   **locx** – The x coordinate of the control on the wizard page. The value of this field is automatically changed when you move the control using your mouse.

    ▪   **locy** – The y coordinate of the control on the wizard page. The value of this field is automatically changed when you move the control using your mouse.

    ▪   **width** – The width of the control in pixels.

    ▪   **height** – The height of the control in pixels.

Edit control properties

⇒   **Navigate:** In the Configuration Manager, select ☐ Project > ☐ Solution > ▦ Wizards. The Wizard Designer window opens in the workspace.

1. Select the wizard group tab that contains the wizard you want to edit.

2. Select the wizard tab that contains the widget you want to edit.

3. Select the widget.

4. In the Widget area, update the information as necessary.

Work with solutions

Overview

A solution corresponds to an application configuration (for example, Merchandise Financial Planning or Item Planning). Each project can contain one or more solutions. For each solution, you define measures, rules, workbooks, and wizards. Once you create a solution, you can move it from one project to another.

📖   **Note:** Some solutions; such as Curve, Forecast, and Promote have configuration steps that are specific to those solutions. For more information, see the RPAS Solution Extension Guide.

Procedures

Create a solution

⇒ **Navigate:** From the File menu, select New > Solution. The New window opens.
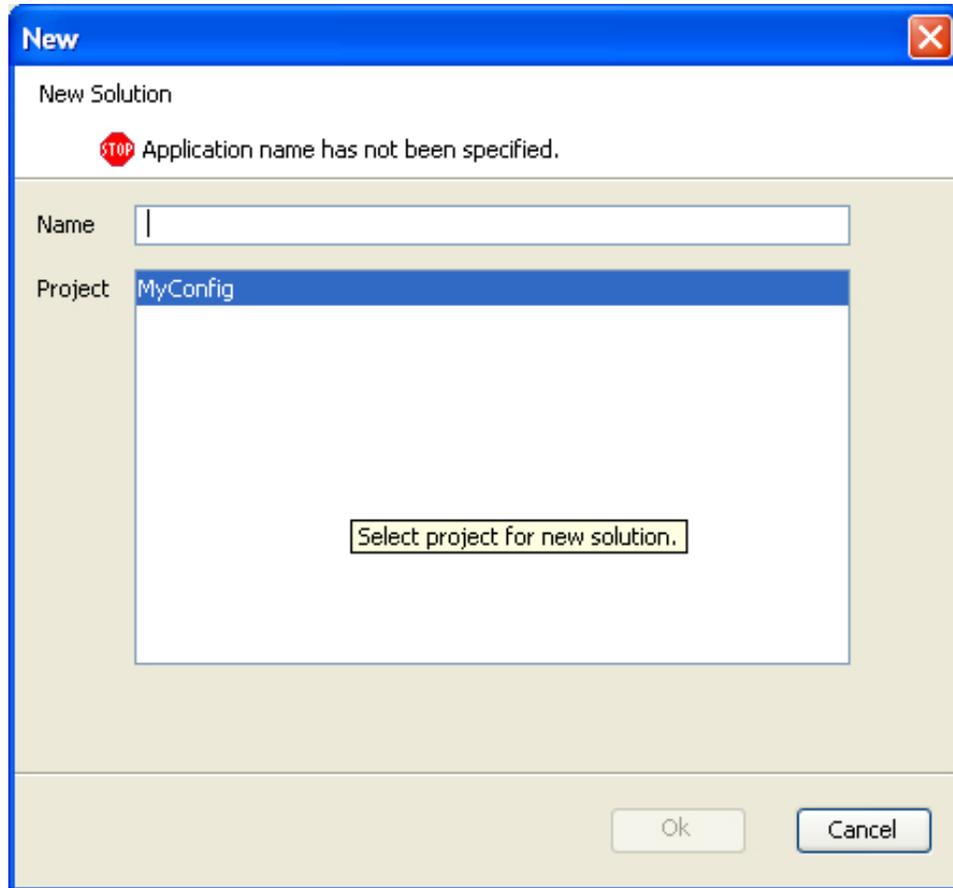
1. In the Name field, enter the name of the solution.

2. In the Project field, select the project that you want associated with the solution.

   📖 **Note:** The tools only support one project, which will be selected in the dialog box that follows.

3. Click **OK** to save any changes and close the window.

Copy a solution

1. Select the solution that you want to copy.

2. Right-click in the Configuration Manager, and select Copy.

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

4. Select the project where you want to create the copied solution.

5. Click **Finish** to save the copied solution in the specified project.

Move a solution

1. Select the solution that you want to move.

2. Right-click in the Configuration Manager, and select Move.

3. Type the name of the solution in the text box.

    Note: This is what the solution is named after it has been moved.

4. Select the project where you want to move the solution.

5. Click **Finish** to move the solution to the specified project.

Chapter 5 – System preferences

Overview

You can set general preferences for the configuration tools at the workbench level, which refers to the entire tool. This includes the domain type and general preferences.

Global Domain

Overview

The Global Domain is a type of structure that provides the ability to view data from multiple domains and to administer common activities of an RPAS domain and solution.

Domains can be built in one of two methods:

1. Simple domain – This is the traditional, stand-alone domain that has no visibility to other domains.

2. Global Domain – This is a domain environment that contains two or more local domains (or sub-domains) and a master domain that has visibility to all local domains that are part of that environment.

There are two primary functional benefits in using a Global Domain environment:

1. The ability to have a global view of data in workbooks

   You can build workbooks with data from local domains, refresh global workbook data from local domains, save global workbooks, and commit the data from global workbooks to the individual local domains.

   "Local" domains are typically organized (partitioned) along organizational structures that reflect user responsibilities and roles. Most users will only work within the local domain(s) that contain their area of responsibilities, and they may not need to be aware of the Global Domain environment. For performance and user contention reasons, Global Domain usage should be limited to relatively infrequent processes that require data from multiple local domains.

2. Configuration and administration

   Most of the mechanisms that are required to build and administer a domain have been centralized, and they need only be run in the "master" domain, which either propagates data to the local domains or stores it centrally so that the local domains reference it in the master.

     Note: For a Global Domain environment to function properly, all local domains must be structurally identical.

     Note: The check boxes are domain properties, which will not change the configuration. The Configuration Language box holds the content of the language that was specified during the initial configuration. You can view the configuration language field that was specified when the configuration was initially created, but you cannot change it.

Measure Data

In a global domain environment, measure data can be physically stored across the local domains or in the master domain.

Measure data that is stored in local domains is split across the domains based on a pre-determined level of a given hierarchy. This level is defined during the configuration process, and it is referred to as the "partition" level.

The base intersection of a measure (for instance, what dimensions a measure contains) determines whether data is stored in the local domains or in the master domain. The data will be stored in the master domain if the base intersection of a measure is above the "partition" level or if it does not contain the hierarchy on which the Global Domain environment is partitioned. This type of measure is referred to as a "Global Domain measure," or a "Higher Base Intersection measure."

Consider a global domain environment where the partition-level is based on the Department dimension in the Product hierarchy. In this scenario, data for measures that have a base intersection in the Product hierarchy at or below Department (other hierarchies are irrelevant for this discussion) is stored in the local domain. This is based on which Department the underlying position in the Product hierarchy belongs to.

Measures that have a higher base intersection in the Product hierarchy than Department (for instance, Division) or measures that do not contain the Product hierarchy (such as a measure based at Store-Week, no Product Hierarchy) cannot be split across the local domains. These measures will reside in the master domain, and they will be accessed from there when these measures are required in workbooks.

All measures will be registered in the master domain, and they will be automatically registered in all local domains. RPAS automatically determines where the measure needs to be stored by comparing the base intersection of the measure against the designated partition-level of the Global Domain environment.

The physical location of the measure data will be invisible to the user after the measure has been registered; however, administrators must know where data for each measure is to be stored (master vs. local) as the data must be loaded in the proper location.

Hypersparse

Sparsity is a term that is used to describe the density of measure data, which indicates the percentage of data points that have populated values. A populated value is a value that is not equal to the NA value, which is most commonly zero for numeric measures. Data is either dense, sparse, or hypersparse. By default, domains are not hypersparse, which means that they only contain dense or sparse data. However, if a domain is a certain size (based on the number of positions) and the data reaches a specific density, the domain must be a hypersparse domain. This is most common for very large environments (organizations with thousands of products and hundreds of stores).

Whether a domain should be hypersparse is a technical decision that should be made with the consultation of Oracle Services.

    📖   **Note:** It is possible to convert a domain to and from a hypersparse setting after the domain has been built by using an RPAS utility. This is described in the RPAS Administration guide.

Multi-Language

RPAS domains are built to be used in English only or in English and other languages. Multi-lingual domains allow for most data elements in an RPAS domain to be translated into another language. The translation process is managed by the Oracle Translation group and is handled as a separate agreement with Oracle.

📖   **Note:** An existing domain cannot be converted to multi-lingual after it has been built.

Procedures

Set Tools (Workbench) preferences

⇒ **Navigate:** From the File menu, select Tools Preferences. The Tools Workbench Preferences window opens.
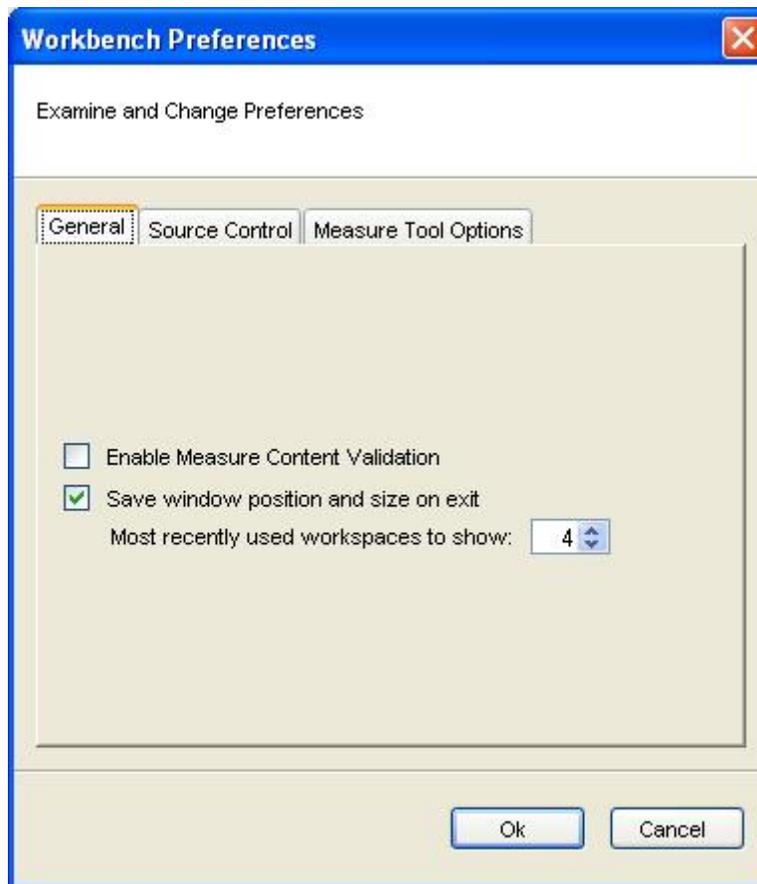
1. Select the General tab.

    ▪ In the Most Recently Used Entries field, use the up and down arrows to specify the number of configurations to be displayed in the Most Recently Used (MRU) list.

    ▪ In the Measure Content Validation field, select the check box. This check box enables validation of measure properties when configuration measure information.

2. Select the Source Control tab. Enter the following information:

    ▪ User – Your CVS user ID

    ▪ Password – Your CVS password

    ▪ Server – The name of the machine hosting the CVS server

    ▪ Repository – The path to the CVS repository

3. Select the Measure Tool Options tab.

    ▪ Number of Measures/Page – select the number of measures to display per page in the Measure Tool.

    ▪ Display Measures by – Display measures by either their name or label in various locations of the Tools.

    ▪ Display Measure components by – Display measure components (in the Measure tool) by either name or label.

4. Click **OK** to save any changes, and close the window.

Set configuration properties

⇒ **Navigate:** From the File menu, select Configuration Properties. The Configuration Properties window opens.
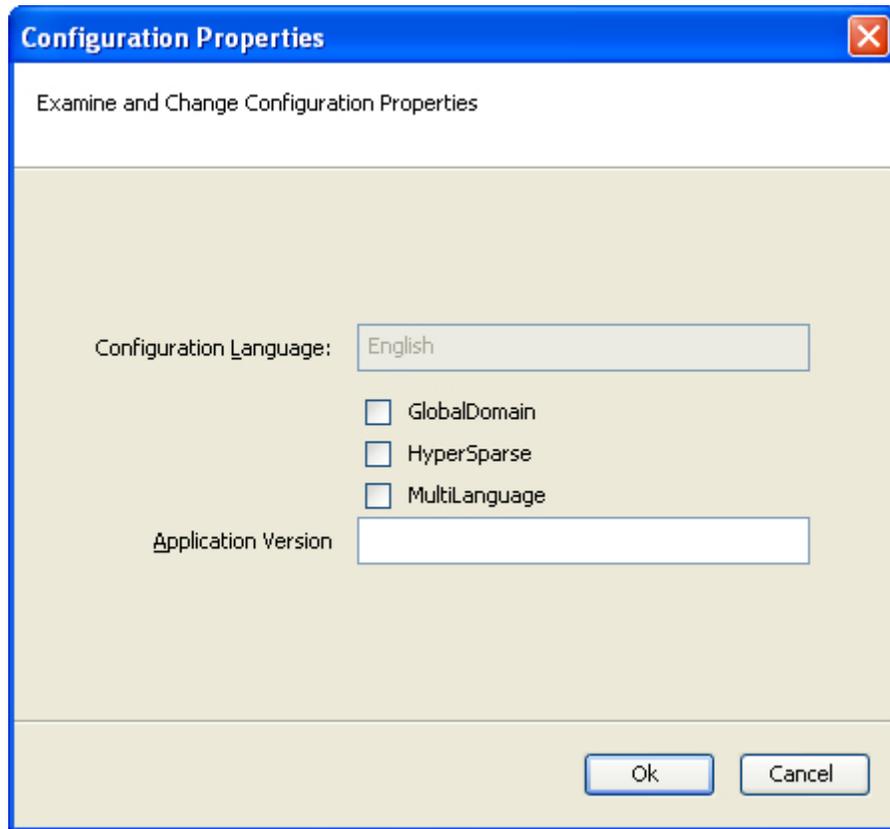
1. **Configuration Language** – Currently configurations can only be in English.

2. Select the following check boxes as necessary:

   ▪ **Global Domain check box** – The configuration uses a global domain environment. This environment enables you to create workbooks from multiple domains and to administer and update multiple domains from a single master domain.

   ▪ **Hypersparse check box** – The configuration uses a hypersparse domain. Hypersparse is a format for storing data where up to 5 quintillion cells can be handled. This format is characteristic of global organizations with thousands of products at hundreds of locations.

   ▪ **MultiLanguage check box** – The configuration supports multiple languages in the domain. This does not effect the configuration.

3. Click **OK** to save any changes, and close the window.

Chapter 6 – Configuration utilities

Overview

The following utilities are stand-alone utilities that can be run externally or from within the Configuration Tools. The utilities provided include the Configuration Converter, Measure Extractor, Rule Extractor, and the Function Library Manager; which are described in detail below. These utilities can be accessed from within the Configuration Tools in the Utilities pull down menu or from the Configuration Tools Utilities directory in Windows Explorer.

Configuration converter

The Configuration Converter allows you to convert a configuration that was originally created and saved in a prior release of the Configuration Tools.

In the Configuration Tools:

From the Utilities menu, select Configuration Converter.
In Windows Explorer:

1. Navigate to the configuration tools repository.

2. Open the Utilities subdirectory.

3. Open Converter.bat.

4. Specify a new directory to which the newly converted configuration will be saved. The converter automatically updates the configuration to a more recent release of the Tools.

Measure extractor

The Measure Extractor is used to extract all measure information within the configuration to a flat file. The file is created in the root directory of the configuration called measout.txt. This file contains all of the measures and their properties within that solution. The first line of the file is the names of all of the measure properties. Thereafter, each measure is listed one to a line, with the properties separated by commas.

In the Configuration Tools:

⇒ **Navigate:** From the Utilities menu, select Measure Extractor.

In Windows Explorer:

1. Navigate to the configuration tools repository.

2. Open the Utilities subdirectory.

3. Open Measure Extractor.bat.

4. You are prompted to load a configuration and select the project and solution within the configuration.

Rule extractor

The Rule Extractor can be used to extract the rules in the configuration, for all rule sets and rule groups within a solution, to a flat file. A file contains a listing of the rules within that solution. Each line contains a one expression or one rule. The format of each line within the rule file is the name of the rule set, the rule group, the rule, and the expression. Each component is separated by commas.

In the Configuration Tools:

⇒ **Navigate:** From the Utilities menu, select Measure Extractor.

In Windows Explorer:

1. Navigate to the configuration tools repository.

2. Open the Utilities subdirectory.

3. Open RuleExtractor.bat.

4. You are prompted to load a configuration.

Function library manager

The Function Library Manager is used to allow custom functions to be validated within the Configuration Tools. Any custom functions are validated when you use them in a rule inside the Rule Tool. This custom function must exist in the /applib directory of your RPAS_HOME variable settings. If necessary, you can also use this utility to remove custom function libraries.

In the Configuration Tools:

⇒ **Navigate:** From the Utilities menu, select Measure Extractor. The Function Library Manager window opens.
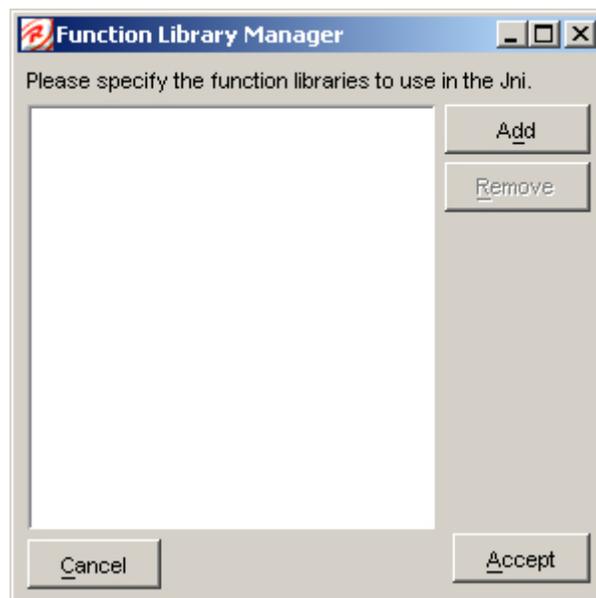
Function Library Manager window

In Windows Explorer:

1. Navigate to the configuration tools repository.

2. Open the Utilities subdirectory.

3. Open Function Library Manager.bat. The Function Library Manager window opens.

4. Click **Add**. The Input window opens.

5. Enter the name of the library to include.

    📖   **Note:**   Enter the name without the *.dll or *.so extension.

6. Click **OK**.

7. Click **Accept** to save any changes and close the window.

Chapter 7 – Building and patching domains

Build a domain

Overview

After you create a fully defined configuration, you can build an RPAS domain. Since building an RPAS domain is a manual process, you should be skilled in administering UNIX or NT servers and should have scripting skills.

Prerequisites

The following prerequisites are required before beginning the domain build process:

- Installation of RPAS on your server. Refer to the RPAS Installation Guide for more information.

- Installation of the Configuration Tools on your server. Refer to the RPAS Installation Guide for more information.

- Configuration built using the Configuration Tools.

- A collection of hierarchy input files that contain positions for the domain; a hierarchy data file (name.dat) is required for each defined hierarchy.

After building the domain

After building the domain and loading measure data:

- Use the RPAS EConfigure utility to specify the location of the domain on the server and designate a port number where the RPAS server (DomainDaemon) is running.

- Begin the RPAS server process (DomainDaemon) with the port number specified in the EConfigure utility. Refer to the RPAS Administrator's Guide for instructions on executing the DomainDaemon.

View the domain using the RPAS client. Set up the appropriate permissions on the measures in the domain using the Measure Administration workbook and then view the measures by building a Measure Analysis workbook. Refer to the RPAS 11.1 Administrator's Guide for instructions on using these workbooks.

Patch a domain

Overview

In order to patch your RPAS domain with changes to your configuration project, use the -patchinstall option when running the Installer. Prior to the beginning the patchinstall process, we recommend that you:

- save all workbooks

- create a backup copy of the domain

- verify that you are using the patchinstall option in the rpasInstall command instead of the -fullinstall option

    📖   **Note:** You should only change the fullinstall option to patchinstall inside your call to rpasInstall. You should not change any other options in the rpasInstall call when running a patchinstall.

Currently, the patchinstall operation does not update any hierarchy or dimension changes in your configuration with the exception of user defined dimensions. This includes being unable to patch changes to the hierarchy purge age, the security dimension, and the multi-language flag inside the Workbench.

    📖   **Note:** If you need to update the hierarchy purge age inside your existing domain, you can use the loadHier RPAS utility in batch mode to override the current settings. Refer to the RPAS Administrative Guide for further details regarding the usage of the loadHier utility.

The patchinstall operation allows for changes to only some measure properties. See the following section for additional information. The patchinstall operation also allows for changes to rule sets, rule groups, and workbook templates. During the process the rule sets, rule groups, rules, and workbook templates are completely rebuilt.

Applying a new tools build patch release

When there is a new release of the RPAS Configuration Tools, you will need to replace the entire Tools directory with the new release. You should make a backup copy of your existing configuration before using it with the new Tools release. You should also create a new directory for the new Tools release rather than overlaying the old release. Remember to update your RIDE_HOME environment variable as needed to point to the new Tools release. Refer to the RPAS Installation Guide for more details.

Patching measure properties

It is important to note that certain measure properties cannot be updated without unregistering and reregistering measures, which result in the loss of measure data. Measure properties that can be modified are referred to as updateable, while those that cannot are not updateable.

A complete list of measure properties and which ones are updateable are available in an appendix of this document.

Client-side procedures

Export the configuration project from the configuration tools

Overview

To begin the domain build process, you need to start with a configuration project that was built using the Configuration Tools. You can use a purchased configuration or create a new one with your specific hierarchies and measures. If you are creating a new configuration, be sure to note the path where your configuration is saved on your local disk.

    📖   **Note:** In your configuration, if you create a measure that requires a database, you must specify the database as **data/dbname**. If you do not specify the data part of the path, the measure database will be built directly under the domain root.

Procedures

Zip up the configuration project folder

1. Find the location where you saved your configuration project.

2. Using Windows Explorer, go to the path of the configuration project.

3. Right-click on the Configuration Project folder, and select Add to Zip. You should zip the entire contents of the project beginning with the configuration project root folder such that the zip file will include the Project(s) and Solution(s). It is important to zip the entire configuration project for the entire directory structure and not just the specific .xml files. Do not change the name of the configuration project folder or alter the contents of the folder in any way.

4. In the following example, you should click on TPGA to highlight the folder, and right-mouse click and select Add to TPGA.zip.

📁 Configuration Project		(directory for all configurations)
	📁 TPGA	(Configuration Name directory)
		📁 TP	(Project directory)
		TPGA.xml	(Configuration .xml file)

5. Using FTP, transfer the <Configuration Project>.zip file over to the server in binary mode. You can place it in your home directory for now.

Load measure data

Overview

    Note: The following is optional. If you have already purchased a solution template that contains a script to load data, you can skip this section and proceed with the installation process of your solution template.

During the initial domain build process, all of the hierarchy and measure data files in your input home directory will be copied to the input directory domain. If necessary, move or copy any new measure data files to the input directory of the domain. The measure data file must have a valid measure filename, such as xxx.ovr, xxx.rpl, xxx.inc, or xxx.clr, where xxx is the measure name.

Procedure

Use the *loadmeasure* utility to load your data. Loadmeasure should be in your path, which means that you can run it from any directory. The syntax for loadmeasure is as follows:

loadmeasure –d domainPath –m measName (–l | –a) {–u user:password}

where:

–d domainPath: Specifies the domain in which to load the measure.

–m measName: Specifies the name of the measure to load. The name should be lowercase.

–l: Use this argument to load the named measure.

–a: Use this argument to apply any staged loads for the named measure.

–u user:password: Specifies an Acumate user and password. If –u is not specified, adm is used.

The loadmeasure utility must be used for each measure for which you want to load data. It is recommended to redirect the output to a log file.

Example:

loadmeasure -d /domains/D01 -m rsal -l > /domains/logs/rsal.log.d01 &

Measure loading properties

Property	Description	Updateable
Load Intersection	Intersection at which measure is loaded (for ovr, inc loads). Defaults to the base intersection. The load intersection can only be specified at or below the base intersection.	Not Updateable
File Name	File used to load measure (suffix or ovr, inc., clr). Defaults to the measure name.	Updateable
Start Position	Start column of data (defaults to column after dimensions).	Updateable
Column Width	Width of data (defaults by type). **Note:** For Boolean types, width can be >=1, but only the first character is read.	Updateable
Load Aggregation Method	The aggregation method used when aggregating the data from the load intersection to the base intersection.	Updateable

Server-side procedures

Import the configuration to RPAS

Overview

After you have exported the configuration project, you can import it to the server and begin building the domain. You can build a new domain, or you can add fixes to an existing domain by applying patches. Once you have made all of the changes to your domain, you should validate the domain build results.

    📖   **Note:** The configuration tools are supported on all platforms (Sun, HPUX, AIX, and NT); however, they require Java v1.4 or higher. Make sure that the server you will be using has this version of Java.

Procedures

Unzip the configuration file

1. Find the location where you want to save your configuration project file.

    📖   **Note:** Always put a newly changed configuration project in a new directory path. Do not overwrite an existing configuration project.

2. Move the <Configuration Project>.zip file to this location.

3. Unzip the <Configuration Project>.zip file using the UNIX command:

        unzip a <Configuration Project>.zip

4. Do not change the directory name for the configuration project or alter the contents in any way.

Verify the environment variable settings

Prior to this step, you should have installed RPAS and the Configuration Tools on your server. During that process, you should have set up the necessary environment variables for RPAS and the Configuration Tools. Refer to the RPAS Installation Guide if the environment variables below have not been set up.

Log into the server. Use the commands below to verify your environment settings:

echo $RPAS_HOME

echo $RIDE_HOME

echo $JAVA_HOME

echo $PATH

    📖   **Note:** The path for the RPAS_HOME variable changes with each new RPAS release.

If you make any changes to the environment variable settings, remember to exit and restart your UNIX session in order to execute your .profile and make the changes effective. This step is very important before you continue to the remaining steps.

Get the input files ready

1. Designate a directory for the location of your input files, and move your files into this directory.

   📖 **Note:** As a recommendation, use the directory name **to_rpas** as a standard for the location of input files. At a minimum, the hierarchy files (product, location, and calendar hierarchy files) are needed to build the domain. At this time, a calendar file must be loaded.

2. If necessary, rename the hierarchy files to **prod.dat**, **loc.dat**, and **clnd.dat**. These file names are required. The format of these files must match the hierarchy configuration specified using the configuration tools.

Build the domain

1. Run the **rpasInstall** script to build the domain. This executable is located in the path to $RIDE_HOME/bin. The syntax for the rpasInstall script is as follows:

   ```
   rpasInstall [-version | -testinstall | -fullinstall | -
   patchinstall] cn <config_name> ch <config_home> dh
   <domain_home> in <input_directory> [-rf <function_name>]
   log <logfile_name>
   ```

   📖 **Note:** A full explanation of the RPAS Install options are available.

Required:

   [-fullinstall | -patchinstall | -testinstall]: the type of installation to be performed.

   where:

   -fullinstall – Builds a full domain and loads the hierarchy data files.

   -patch install – Patches an existing domain. Updates or unregisters/registers measures as necessary that have changed.

   -testinstall – Used for testing only. Only generates configuration files.

   ch <config_home> – Where <config_home> is path to the directory that contains the configuration.

   -cn <config_name> – Where <config_name> is the name of the configuration.

   -in <input_home> – Where <input_home> is the directory that contains the input files for the domain to be created.

   -log <log_name> – Where <log_name> is the name of the log file to be created.

Optional:

   -v-version – Provides the version number of the Tool build as it relates to the release or patch number.

   -rf <function_name> – Where <function_name> is the filename of the function to be registered. This pairing may be repeated for multiple functions.

Conditional:

> **-configdir <config_directory>** – Where <config_directory> is the path to the directory that contains the xml files that are used by RPAS. This is a required argument if the user wants to supply globaldomainconfig.xml or calendar.xml.

> **-dh <domain_home>** – Where <domain_home> is the path to the directory where the domain will be created. This does not include the root of the domain. This is a required option if globaldomainconfig.xml is not used.

> **-p <dim_name>** – Where <dim_name> is the partitioning dimension. Use this if the global domain is being implemented without the use of globaldomainconfig.xml.

Note: When building a domain for the first time, in addition to the directory for the domain, an installs directory gets created at the same level as the domain. The installs directory is essential for the patch process, so you should not move or rename this directory.

When sending in an issue to Retek Customer Care, if asked to provide a domain, be sure to provide the installs directory with it. For completeness, you should provide the following items in order to help Customer Care better diagnose the issue:
The configuration

- The script used to run the rpasInstall script

- The domain

- The install directory

- The log output file

Validate domain build results

Once the domain build process finishes, you should check the logfile to verify that the process executed successfully. Search for the words **ERROR, FAILURE,** and **exception** inside the logfile. The end of the logfile should look similar to the output below:

```
Time: 58.451
```

```
COMPLETE
```

Appendix A – Global Domain technical information

The following points should be noted when configuring and setting up a global domain environment:

- When creating a new global domain configuration file, globaldomainconfig.xml, reference the RPAS example of the configuration file, globaldomainconfig_example.xml, which is located in the %RPAS_HOME%/domain/config_examples directory of the RPAS installation. Reference this to use as a guide in building this file. Once this file is created, put it in the configdir directory path as specified by using the –configdir option in the Installer.

- In the globaldomainconfig.xml file, in the path the user must specify the entire path to the master domain and the sub-domains including the root name of the domains. The path leading up to the root of the master domain must exist, but the master domain root directory must not exist at the beginning of the domain build process. The path to the sub-domains must also exist unless the sub-domains are located inside the root of the master domain.

- The master global domain will contain a directory called config, which will house the globaldomainconfig.xml file. You should not delete this directory or file.

- You should not delete the tmp directory under the domain home directory while the domain build process is taking place.

- To configure the global domain functionality, the user can provide a globaldomainconfig.xml file that specifies how each of the local domains should be partitioned based on position groupings. If a globaldomainconfig.xml file is not provided and the partitioning dimension is provided, a local domain will be built for each position within the partitioning dimension. For example, if a company has five departments, and the partitioning dimension is department, there will be five local domains and one master global domain.

- Conditional parameters that are used by the Installer for configuring and setting up a global domain environment are noted below:

    -dh <domain_home> : where <domain_home> is the path to the directory in which the domain will be created

    > Non-Global Domain – required

    > Global Domain (w/ globaldomainconfig.xml) - ignored

    -configdir <config_directory> : where <config_name> is the path to the configuration xml files, including globaldomainconfig.xml, hierarchy.xml, and calendar.xml

    > Non-Global Domain – optional, but required if using a calendar.xml file

    > Global Domain (w/ globaldomainconfig.xml) - required

    > Global Domain (w/o globaldomainconfig.xml) - optional, but required if using a calendar.xml file

–p <dim_name>: where <dim_name> is the partitioning dimension. Only valid if the configuration has been marked as a global domain configuration with the Properties dialog box. If the configdir option is specified and a globaldomainconfig.xml file is found there, the –p option will be ignored and the partitioning dimension that is specified within the globaldomainconfig.xml file will be used instead.

Non-Global Domain – ignored

Global Domain (w/ globaldomainconfig.xml) - ignored

Global Domain (w/o globaldomainconfig.xml) - required

- For patching a Global Domain implementation; you can change measures, rules, and workbook templates, and the master domain and local domains will be patched accordingly. You cannot turn a non-global domain implementation into a global-domain implementation and vice versa. The Global Domain flag in the configuration is ignored during the patch process, so the construction of the implementation will not change even if you change its status.

- When patching either a Global Domain implementation or a regular, stand-alone, or single domain; in the call to the Installer (rpasInstall); it is imperative that you do not change any of the parameters that were used during the original domain build. The only parameter that you should change is fullinstall to patchinstall. The other parameters need to remain unchanged. We recommend using a script for the rpasInstall call so that it is easier to change the fullinstall parameter to patchinstall while leaving the other parameters in their original state.

Appendix B – Measure Properties

The following table indicates those measure properties that can be updated via the patch process; those that cannot be updated require that the measure to be unregistered and reregistered, which results in a loss of measure data. See the section,

Specify component information, for additional information about these measure properties.

Property	Description	Updateable
Label	Label of measure. Defaults to measure name. Unique name of measure for display in client (in grid and measure dialogs). In the past, labels were sometimes non-unique. There is no maximum size limit, but you should keep grid display limitations in mind when creating a measure label. History: displayed in wb, but not in measure dialogs.	Updateable
Description	Potentially long, descriptive summary of measure.	Updateable
Type	Data type of the measure. Available types: integer (int), real, string, date, Boolean.	Not updateable
NA Value	Suggested value for unpopulated (na) cells. Defaults according to type. Initially, the base array of a measure is defined with this naval; however, this is subject to change in order to optimize the compression factor.	Not updateable
Base Intersection	Base intersection for measure. Concatenation of all dimension names (4 characters, suffixed by ' ' as necessary). Dimensions are ordered according to the hierarchy to which they belong. In [Measure] SFX array, baseint is prefixed by "I" Simply 'I' for scalar measures.	Not updateable
Default Agg	Default aggregation method (see Rule Functions Reference Guide for additional information). 📖 **Note:** Changes to the Default Agg will only be reflected in new or rebuilt workbooks.	Updateable unless set to recalc

Property	Description	Updateable
Default Spread	Default spread method (see Rule Functions Reference Guide for additional information). 📖 **Note:** Changes to Default Spread will only be reflected in new or rebuilt workbooks.	Updateable unless set to recalc
Base State	Determines if a measure can be edited at the measure's base intersection. Values are read (default) and write.	Updateable
Agg State	Determines if a measure can be edited above the measure's base intersection. Values are read (default) and write.	Updateable
Database	Database where measure is stored in a domain (in the future, a partitioned database which comprises many partitions). If the database is NA, the measure is not stored. It specifies the relative path from domain root.Syntax is as "data/test."	Not Updateable

Property	Description	Updateable
View Type	Indicates if the measure is calculated when viewed. If the type is not 'none,' the measure must not be materialized (not stored in workbook or domain), and it must appear on the LHS of only one expression in a rule (the view expression). Synchronized view types are maintained immediately at cell edit time. View types (and their corresponding integer values) include: • none (0) - measure is stored • view only (1) - measure is calculated when viewed (aggstate and basestate must be read only) • sync_first_lag (2)-period 1 from first measure (no calendar), periods 2..N fromsecond measure 1..N-1(lag)[such as.bop ←→ os &eop] • syn_lead_last (3)- periods 1..N-1 from first measure 2..N (lead), period N second measure (no calendar) {such as.eop ←→ bop & cs} • sync-first (4) - get period 1 from measure (similar to first agg type along calendar dimension) [such as os ←→ bop] • sync-last (5) - get period N from measure (equivalent to last agg type along calendar dimension) [such as. cs ←→ eop]	Updateable
Sync With	Comma separated list of measures used for synchronization. Depends on view type.	Updateable
Insertable	Indicates whether measure is visible to user for inclusion in workbooks (used in addition to measure security). Default to true.	Updateable
UI Type	Indicates whether a measure is a picklist or not.	Updateable

Property	Description	Updateable
Range	Specify suitable range for the measure at edit time. For picklist measures (indicated with ui-type property) these are the displayed/possible values. For numeric values, the syntac is as Lower Bound : Upper Bound. For strings, it is the maximum length. For picklist measures (both numeric and string) values can be explicitly listed as "a,b,c,d...", additionally labels can be specified for each value such as "valuea(labela),, valueb(labelb),..."where the labels will be displayed in the user interface.	Updateable
Purge Age	Limits the history of data in a measure along the time dimension (purge occurs during load measure process). Default to na.	Updateable
Refreshable	No longer used in the 11.x RPAS architecture. The refresh rule group is used to determine which measures are refreshed.	Updateable
Lower Bound	A measure that provides minimum bounding values for this measure's data cells.	Not updateable
Upper Bound	A measure that provides maximum bounding values for this measure's data cells.	Not updateable
Special Value Type	Indicates the different types that are allowed for the special value functionality.	Updateable
Special Value	The value for the special value when the type is specified as user-entered.	Updateable
Dimension Attribute	Indicates that the measure is a dimension attribute measure of which certain qualifications must be met.	Updateable
Dimension Attribute Name	Indicates the name of the dimension attribute measure.	Updateable

Property	Description	Updateable
Dimension Attribute Label	Indicates the label of the dimension attribute measure.	Updateable
Allowed Aggs	The allowed aggregation methods depending on the data type of the measure. This property displays a multi-select checkbox dialog box in the Tools, which contains values; such as total, average, min, and max.	Updateable

Glossary

C

configuration: A configuration represents the overall workspace for an RPAS implementation. For example, a configuration could represent a company or a division.

D

dimension: Levels of a hierarchy, such as day, week, month, year for a calendar hierarchy.

E

expression: Expressions are the basis of all calculations of the relationships between measures,and are evaluated by the calculation engine during a calculation. Expressions arewritten in a syntax that allows for the calculation of a single measure from othermeasures, constants and parameters, using standard arithmetical functions and a richset of mathematical, technical and business functions. Expressions are thereforean algorithmic statement of a relationship between measures.

H

hierarchy: Hierarchies are the structures used by an organization to describe the relationships that exist between the many dimensions. For each project, the following hierarchies are automatically created: Calendar, Product, Location, and Measure.

hypersparse: Hypersparse is a format for storing data where up to 5 quintillion cells can be handled. This format is characteristic of global organizations with thousands of products at hundreds of locations.

M

major component: A major component specifies the default measure property information that is inherited by all minor components. Ultimately, this property information is inherited by the measures that are generated from the component structure.

measure: Any item of data that can be represented on a grid in a worksheet.

measure profile: The worksheet-specific set of viewable measures.

minor component: A minor component inherits measure property information from the major component to which it belongs. Property information can also be modified at this level. Measures generated from these components inherit the property information defined for the minor component.

P

project: A project within a configuration represents an RPAS domain. For example, configuration XYZ could contain a Planning project or Forecasting projects.

R

realize: When a measure is auto-generated by the Measure Tool, it is considered a prototype measure and cannot be edited or used in any other configuration component. To enable the measure to be edited and/or used elsewhere in the configuration, it must be realized first.

rule: Rules are collections of expressions that describe the same relationship between measures, but that solve for different measures. All the expressions in a rule must use the same measures, and must have a different target measure (the measure on the left hand side of the expression that is calculated by the expression).See expression.

rule group: A rule group is a collection of rules that are treated as a unit by the calculation engine, with the integrity of all the rules in the rule group being maintained together. The calculation engine always has one (and only one) active rule group.

rule set: A collection of rule groups, each of which is used by the calculation engine for a specific purpose. By default, each rule set contains the following rule groups: load, calculate, refresh and commit.

S

solution: A solution corresponds to an application configuration within an RPAS domain. For example, project XYZ could contain Retek Item Planning and Retek Merchandise Financial Planning solutions.

W

wizard: A wizard is a feature that steps the user through the process of building a new workbook from a template. A wizard displays successive dialogs that require the user to answer a sequence of questions or enter selections regarding the content of the workbook. Responses to these questions are used to automatically format and populate the workbook.

workbook: A workbook is the easily viewed, easily manipulated multidimensional framework used to perform specific business functions such as generating trading information and reviewing availability data.

workbook tab: A workbook tab is a major subdivision of a workbook. A workbook contains one or more workbook tabs.

worksheet: Worksheets are multidimensional spreadsheets used to display workbook-specific information. Workbooks can include one or many worksheets. Worksheets can present data in the form of numbers in a grid, or these numeric data values can be easily converted to a graphical chart.