Retek® Predictive Application Server™ 11.0.4

Configuration Guide

Retek

Corporate Headquarters:

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

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

Fax: +1 612.587.5100

European Headquarters:

Retek
110 Wigmore Street
London
W1U 3RW
United Kingdom

Switchboard:
+44 (0)20 7563 4600

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

Fax: +44 (0)20 7563 46 10

Printed in the United States of America.

Customer Support

Customer Support hours

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

Depending on the Support option chosen by a particular client (Standard, Plus, or Premium), the times that certain services are delivered may be restricted. Severity 1 (Critical) issues are addressed on a 7x24 basis and receive continuous attention until resolved, for all clients on active maintenance.

Contact Method	Contact Information
Internet (ROCS)	www.retek.com/support Retek's secure client Web site to update and view issues
E-mail	support@retek.com
Phone	US & Canada: 1-800-61-RETEK (1-800-617-3835) World: +1 612-587-5800 EMEA: 011 44 1223 703 444 Asia Pacific: 61 425 792 927
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 RPAS configuration tools provide a flexible means to configure and build RPAS-based applications with customer-specific business parameters.

The configuration tools consist of an integrated set of task-specific configuration “aids” that can be used to configure a solution from scratch, or to use an existing solution configuration template as a basis from which to make customer-specific changes. RPAS functionality is exposed via APIs to the configuration tools, which provide a streamlined and user-friendly interface to utilize RPAS functionality. Once configured, a set of installer scripts is used to build an RPAS domain and the configured solution.

This document describes how to use the RPAS configuration tools and how to build an RPAS domain.

Concepts

The core of the RPAS configuration tools is the configuration manager. The configuration manager provides an overall view of the configuration components (projects and solutions, which contain hierarchies, measures, rules, workbook templates, and wizards).

An RPAS configuration represents the overall workspace. For example, you could create a workspace called XYZ for customer XYZ. Within configuration XYZ, you create one or more projects. Each project represents an RPAS domain. For example, within configuration XYZ, you might create a separate project called Planning.

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 that comprise a hierarchy. For each project, the following hierarchies are automatically created: Calendar, Product, Location, and Measure. You create and define dimensions for these hierarchies, and for any additional hierarchies that you create.

For each project, you can also specify data interfaces for measures (that is, the data files from which measure data will be imported).

Each project contains one or more solutions, which corresponds to an application configuration (for example, Financial Planning or Item Planning). For each solution, you define measures, rules, workbooks, and, optionally, wizards.

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

Rules are collections of expressions that describe the relationships between measures. Expressions are the basis of all calculations of the relationships between measures, and are evaluated by the calculation engine during a calculation. Expressions are written in a syntax that allows for the calculation of a single measure from other measures, constants and parameters, using standard arithmetical functions and a rich set of mathematical, technical and business functions. Expressions are therefore an algorithmic statement of a relationship between measures. All the expressions in a rule must use the same measures; that is, all inverses of a rule must be defined in expressions for that rule They must also have a different target measure (that is, the measure on the left hand side of the expression that is calculated by the expression).

A workbook is the easily viewed, easily manipulated multidimensional framework used to perform specific business functions such as creating a merchandise plan and reviewing availability data. To present data, a workbook can contain any number of multidimensional spreadsheets, called worksheets, as well as measures that are used on the worksheets and the rules that are used to generate the measure data.. All of these components work together to facilitate viewing and analysis of business functions. The configuration tools allow you to configure workbook templates incorporating these various components.

A wizard is a feature that steps the user through the process of building a new workbook. 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. The configuration tools allow you to create custom wizard formats.

Configuration Steps

Follow these steps to create an RPAS configuration.

- Set up system properties and source control. See Chapter 3 – System properties and source control.
- Create a configuration. See Working with configurations.
- Create projects. See Working with projects.
- Create solutions. See Working with solutions.
- Create hierarchies and dimensions. See Working with hierarchies and Working with dimensions.
- Create measure classes and measures. See Working with classes and Working with measures.
- Create rule sets, rule groups, and rules. See Working with rule sets, Working with rule groups, and Working with rules.
- Create workbooks, workbook tabs, and worksheets. See Working with workbooks, Working with workbook tabs, and Working with worksheets.
- Create wizards. See Working with wizards.
- Define data import information. See Working with the Data Interface Manager.

- If applicable, create solution extensions. See the RPAS 11.0.4 Solution Extensions Guide for more information**Error! Reference source not found.**.
- Build an RPAS domain. See Chapter 12 – Building an RPAS domain.

Chapter 2 – Using menus and toolbars

Application main menu bar

The menu bar and the items contained in each menu are context-sensitive, and change according to what you are working on and how you are working. Menu items that are grayed out are not available to you in your current work mode.

The menu bar displays the following options:

- File
- Window
- Help

Menu shortcuts

You can also access menus by pressing the Alt key followed by the underlined letter for the menu item. For example, <Alt+F> accesses the File menu. Some menu items have shortcut key combinations; those that do have the key combination displayed next to the menu item name. For example, <Ctrl+N> accesses the New file dialog box. In addition to key combinations, some menu items can be directly accessed from toolbar buttons.

In addition, standard Windows shortcuts are included in the tools, although they may not appear on a menu. Examples are Ctrl+V (copy) and Ctrl+C (paste).

File

File > New

Allows you to select from the following:

- Configuration
- Project
- Solution
- Curve Solution
- Forecast Solution
- Promote Solution

Note: Depending on what already exists in your configuration, not all of the above options may be selectable. For example, you cannot select Solution until you have created a project for your configuration. Also, once you have created Curve, Forecast, and Promote solutions for a project, they are no longer available for that project.

File > Open

Displays a dialog box that allows you to access all the configurations that you have created and saved. To open a configuration, highlight the selection you want to view and click Open.

File > Source Control

The Allows you to check configurations into and out of a CVS repository, so that more than one person can work on a configuration at the same time. You can select from the following options:

- Import Configuration
- Synchronize Configuration
- Add Configuration

File > Close

The Close menu item is only available when a configuration is open. If the configuration has been previously saved and no new changes have been made, selecting Close closes the workbook.

File > Save

Stores all the information in the configuration. If the configuration you are working on has previously been saved, then selection of this menu item will update the stored information. If the configuration has not been previously saved, then selection of this menu item will open the Save As dialog box, which will prompt you for a configuration name. The Save menu item is only enabled when a workbook has been created or opened.

File > Save As

Opens a dialog box in which you can specify the name of the configuration to be saved. The Save As menu item is only enabled when a configuration is open.

In the Configuration text field of the Save As dialog, type a name for the configuration.

Selecting the “Use Defaults” check box saves the configuration in the default location. This location defaults to the last location you saved to. If the “Use Defaults” check box is not checked, you can specify a directory in which to save the configuration. If the directory does not exist, it will be created. Alternately, you can click Browse to select a directory in which to save the configuration.

Click OK to save the configuration, or click Cancel to return to the configuration without saving.

File > Properties

The General tab allows you to specify how many items to display in the Most Recently Used (MRU) list, to view language configuration information, and to set up source control.

File > MRU (Most Recently Used) List

The MRU (Most Recently Used) List is located just above the Exit option on the File menu. The MRU List is a list of the most recently opened configurations. When a configuration is displayed on the MRU List, it can be accessed quickly by selecting that configuration directly from the File menu.

If you open a configuration that already exists and save it under a new name by selecting Save As from the File menu, then both the old and the new configuration names will appear on the MRU List.

File > Exit

The Exit option logs you off of the application and exits the system completely.

If you have an unsaved configuration open at the time you select Exit, the Close dialog box will be opened, informing you that modifications have been made to the configuration.

Window

The Window menu allows you to navigate quickly through your open configurations. This provides an alternative to using the tree structure in the Configuration Manager window.

Window > Configuration > Hierarchies

Opens the Hierarchy Definition window for the configuration you select. See Chapter 5 – Hierarchies for more information.

Window > Configuration > Data Interface

Opens the Data Interface window for the configuration you select. See …

Window > Configuration > Solution > Measures

Opens the Measure Definition window for the solution you select. See Chapter 6 – Measures for more information.

Window > Configuration > Solution > Rules

Opens the Rule Definition window for the solution you select. See Chapter 7 – Rules for more information.

Window > Configuration > Solution > Workbooks

Opens the Workbook Designer window for the solution you select. See Chapter 8 – Workbooks for more information.

Window > Configuration > Solution > Wizard Designer

Opens the Wizard Designer window for the solution you select. See Chapter 9 – for more information.

Help

Help > Contents

Displays the Table of Contents for the online help system.

Help > Index

Displays an index of the terms found in the online help system.

Help > Search

Use this function to search the online help system for a word or phrase.

Quick (right-click) menus

Overview

In various areas throughout the application, context-sensitive quick menus, also called right-click menus, are available to access certain commands. To access a quick menu, place the cursor over an appropriate screen area and click the right mouse button. These quick menus are context-sensitive; that is, their availability, appearance, and the options they offer differ depending on your current mode of work.

There are other uses for quick menus, as well, depending on your mode of work. The following is a description of the various right-click menus you may encounter:

Configuration Manager quick menu – Obtained when you right-click over the left frame of the Configuration Manager window. Use this menu to create new configurations, projects, and solutions. You can also use it to open, save, rename, clone, and close existing configurations, projects and solutions.

Hierarchy Definition quick menu – Obtained when you right-click over the Hierarchy window. Use this menu to create, delete, and rename hierarchies and dimensions.

Measure quick menus include the Measure Definition menu, and menus for the Classes tab, Measures tab, and Invalid Measures tab. Use these menus to create, rename, and delete classes and measures; validate and invalidate measures; realize and unrealized measures; specify measure properties; push measures down and pull measures up; and show and hide measures.

Rule quick menus include the Rule definition menu and the Rules menu. Use these menus to add, rename, and delete rules, rule groups, and rule sets; add expressions to rules; edit expressions; and generate load and commit rules.

Wizard Designer menu – Obtained when you right-click over the Wizard Designer window. Use this menu to delete and copy wizards and wizard pages.

Configuration Manager

New > Configuration

Creates a new configuration.

New > Project

Creates a new project within the selected configuration.

New > Solution

Creates a new solution within the selected project.

Open

Displays the Open dialog box, from which you can select a configuration to open.

Close

Closes the current configuration. If you have made any changes and have not yet saved them, the Save dialog box is displayed, allowing you to save your changes before closing the configuration.

Close All

Closes all open configurations. For each configuration you have changed and have not yet saved, the Save dialog box is displayed, allowing you to save your changes before closing the configuration.

Save

Saves any changes you have made to the selected configuration.

Save As

Opens the Save As dialog box, allowing you to save the selected configuration under another name.

Save All

Saves changes you have made to all open configurations.

Rename

Opens the Rename dialog box, allowing you to specify a new name for the selected object.

Clone

Opens the Clone dialog box, allowing you to create an exact duplicate of the selected object.

Move

Opens the Move dialog box, allowing you to move the selected object from one location to another.

Delete

Allows you to delete a project or solution, with a delete confirmation message. The user cannot delete a configuration, however.

Hierarchy Definition

The following options are available on the Hierarchy Definition quick menu:

New Hierarchy

Creates a new hierarchy in the Hierarchy Definition window. By default, the hierarchy is assigned a name starting with H followed by the next available number (for example, H59).

New Dimension

Creates a new dimension within the selected hierarchy. By default, the dimension is assigned a name starting with D followed by the next available number (for example, D23).

Set User Maintained

Allows a dimension to be set as a user-defined dimension.

Delete

Opens the Confirm Delete dialog, allowing you to delete the selected hierarchy object.

Rename

Makes the name field editable, allowing you to change the name of the selected hierarchy object.

Undo

Undoes the last action.

Redo

Redoes the last undone action.

Measure

Measure Definition

The following options are available on the Measure Definition quick menu:

Add Major Class

Adds a major class to the Measure Definition window. By default, the class is assigned a name starting with “class” followed by the next available number (for example, class1004).

Add Minor Class

Adds a minor class underneath the selected class. By default, the class is assigned a name starting with “class” followed by the next available number (for example, class1009).

Pull Up

Pulls the selected class up one level in the measure classification structure.

Note: You cannot pull up a major class. Nor can you pull up a minor class directly beneath a major class.

Push Down

Pushes the selected class down one level in the measure classification structure. A new class is created to take the place of the class you are pushing down. By default, the new class is assigned a name starting with “class” followed by the next available number (for example, class1014).

Show

Shows the classes and measures associated with the selected class on the Classes and Measures tabs. A check in the box next to the class indicates that the measures are shown.

Hide

Hides the classes and measures associated with the selected class. The classes and measures are not displayed on the Classes or Measures tabs. The box next to the class is unchecked.

Show All

Shows all classes and measures for the solution on the Classes and Measures tabs. The boxes next to all classes are checked.

Hide All

Hides all classes and measures for the solution. No classes or measures are listed on the Classes or Measures tabs, and the boxes next to all classes are unchecked.

Realize

Realizes the selected measure, and checks the Realized box on the Measures tab. This option is available only if the measure is not already realized.

Unrealize

Unrealizes the selected measure, and unchecks the Realized box on the Measures tab. This option is available only if the measure is already realized.

Rename

Makes the name field editable, allowing you to change the name of the selected class.

Edit Properties

Highlights the properties for the selected measure so you can edit them.

Delete

Displays the Confirm Delete dialog, allowing you to delete the selected class.

Undo

Undoes the last action.

Redo

Redoes the last undone action.

Classes tab

The following options are available on the Classes tab quick menu:

Undo

Undoes the last action.

Redo

Redoes the last action.

Measures tab

The following options are available on the Measures tab quick menu:

Realize All

Realizes all of the displayed measures. The box in the Realized column is checked for each measure.

Unrealize All

Unrealizes all of the displayed measures. The box in the Realized column is unchecked for each measure.

Undo

Undoes the last action.

Redo

Redoes the last undone action.

Realized Measures Tab

Undo

Undoes the last action.

Redo

Redoes the last undone action.

Rule

Rule Definition

The following options are available on the Rule Definition quick menu:

Add > Rule Set

Displays the Add Rule Set dialog, allowing you to add a rule set to the solution.

Add > Rule Group

Displays the Add Rule Group dialog, allowing you to add a rule group to the selected rule set.

Add > Rule

Displays the Add Rule dialog, allowing you to add a rule to the selected rule group.

Rename > Rule Set

Displays the Rename Rule Set dialog, allowing you to change the name and description of the selected rule set.

Rename > Rule Group

Displays the Rename Rule Group dialog, allowing you to change the name and description of the selected rule group.

Delete > Rule Set

Displays the Confirm Delete dialog, allowing you to delete the selected rule set.

Note: All rule groups within the rule set are also deleted.

Delete > Rule Group

Displays the Confirm Delete dialog, allowing you to delete the selected rule group. You cannot delete the load, commit, calc, and refresh rule groups. If you attempt to delete them, the Confirm Delete dialog is displayed, allowing you to empty the rule groups instead.

Note: All rules within the rule group are also deleted.

Undo

Undoes the last action.

Redo

Redoes the last undone action.

Advanced > Generate Load Rules

Auto generates rules in the load rule group, based on the measures contained in the calc rule group that also have a database defined in the measure tool.

Advanced > Generate Commit Rules

Auto generates rules in the commit rule group, based on the measures contained in the calc rule group.

Rules

The following options are available on the Rules quick menu:

Add > Rule

Displays the Add Rule dialog, allowing you to add a rule to the selected rule group.

Add > Expression

Displays the Add Expression dialog, allowing you to add an expression to the selected rule.

Rename Rule

Displays the Rename Rule dialog, allowing you to change the name and description of the existing rule.

Apply Pattern

Displays the Apply Pattern dialog, allowing you to auto generate rules using the same rule pattern as the selected rule.

Edit Expressions

Displays the Edit Expression dialog, allowing you to add expressions to the rule and edit existing expressions.

Delete > Rule

Displays the Confirm Delete dialog, allowing you to delete the rule from the selected rule group.

Undo

Undoes the last action.

Redo

Redoes the last undone action.

Wizard Designer

The following options are available on the Wizard Designer quick menu:

Copy Wizard

Copies the selected wizard page.

Delete Wizard

Deletes the selected wizard page.

Copy Wizard Group

Copies the selected wizard and all associated wizard pages.

Delete Wizard Group

Deletes the selected wizard and all associated wizard pages.

Paste

Pastes whatever you most recently copied using either Copy Wizard or Copy Wizard Group.

Buttons

Hierarchy Definition

Hierarchy frame

Move hierarchy up

Moves the selected hierarchy up one position in the hierarchy table. Note that this indicates the hierarchy's position in the data file only, and not the hierarchy.

Move hierarchy down

Moves the selected hierarchy down one position in the hierarchy table. Note that this indicates the hierarchy's position in the data file only, and not the hierarchy.

Dimension frame

Move dimension up

Moves the selected dimension up one position in the hierarchy table. Note that this indicates the dimension's position in the data file only, and not the hierarchy.

Move dimension down

Moves the selected dimension down one position in the hierarchy table. Note that this indicates the dimension's position in the data file only, and not the hierarchy.

Data Interface Manager

New Meas

Displays the New Measure Specification dialog, allowing you to specify the data interface for a measure.

Delete Meas

Displays the Confirm Delete dialog, allowing you to remove the measure from the list.

Rule Definition

Rule filtering

Allows you to select one of four filters for displaying rules:

- Disable Filtering – all rules are displayed.
- Filter by Measure – filters by measures contained in the rule.
- Filter by Size
- Filter by Validity – only valid rules are displayed.

Toggle the display of rule descriptions

Allows you to display or hide rule descriptions. If you hide rule descriptions, only the rule name and associated measures are displayed in a shorthand format.

Workbook Designer

Move selected Workbook Tab to the left

Moves the selected workbook tab one position to the left.

Move selected Workbook Tab to the right

Moves the selected workbook tab one position to the right.

Create a new workbook template

Creates a new workbook template in the selected solution.

Create a new workbook tab

Creates a new workbook tab in the selected workbook.

Create a new worksheet

Creates a new worksheet in the selected workbook tab.

Edit properties

Displays the Properties dialog for the selected workbook, workbook tab, or worksheet.

Delete workbook or workbook component

Displays the Confirm Delete dialog, allowing you to delete the selected workbook, workbook tab, or worksheet.

Undo last action

Undoes the last action.

Redo last action

Redoes the last undone action.

Cut to clipboard

Cuts the selected object and places it on the clipboard.

Copy to clipboard

Copies the selected object and places it on the clipboard.

Paste from clipboard

Pastes the object from the clipboard into the workbook.

Wizard Designer

Create a new wizard

Creates a new wizard in the selected solution.

Create a new wizard page

Creates a new wizard page in the selected wizard.

Add a Label

Adds a label to the selected wizard page.

Add a Button

Adds a button to the selected wizard page.

Add a radio button

Adds a radio button to the selected wizard page.

Add a check box

Adds a check box to the selected wizard page.

Add a combo box

Adds a combo box to the selected wizard page.

Add a text box

Adds a text box to the selected wizard page.

Toggle gridlines

Displays or hides the gridlines on the selected wizard page.

Finer gridlines

Displays more closely spaced gridlines on the wizard page.

Coarser gridlines

Displays less closely spaced gridlines on the wizard page.

Align left

Aligns the control to the left.

Center horizontally

Centers the control horizontally.

Align right

Aligns the control to the right.

Align top

Aligns the control at the top.

Center vertically

Centers the control vertically.

Align bottom

Aligns the control at the bottom.

Space horizontally

Spaces the control horizontally.

Space vertically

Spaces the control vertically.

Chapter 3 – System properties and source control

Overview

At the system level, you can set general properties for the configuration tools.

General properties

Displaying language information

To display language information for a configuration:

1. From the File menu, select Properties.
2. Click on the General tab.

   The language specified for the configuration is displayed in the Configuration Language field.

   If the configuration supports multiple languages, the MultiLanguage checkbox is checked.

Modifying language information

To modify language information for a configuration:

1. From the File menu, select Properties.
2. Click on the General tab.

   The language specified for the configuration is displayed in the Configuration Language field.

3. Check or uncheck the MultiLanguage checkbox to enable or disable multiple language support.

   Note: Although you can enable or disable multiple language support for a configuration at any time, this information is only used when the RPAS domain is first built. Changes you make to this field are ignored if you perform a patch installation. For more information, see Chapter 12 – Building an RPAS domain.

Adjusting the MRU list

To specify how many entries should be displayed in the Most Recently Used (MRU) list:

1. From the File menu, select Properties.
2. Click on the General tab.
3. In the Most Recently Used Entries field, use the up and down arrows to specify the number of configurations to be displayed in the MRU list.

Source control

Overview

The configuration tools are integrated with the CVS source control system, allowing multiple users to work on a configuration simultaneously, while maintaining data integrity. You must own a CVS license and have a CVS repository set up in order to use this feature. Refer to the CVS documentation for more information on setting up a repository and adding users.

To use this feature, a "master" configuration is stored on a server in a CVS repository. At a minimum, the master configuration should contain a shell configuration consisting of hierarchies, measure class structure, rule sets, and workbook templates. Configuration users can then pull a copy of the configuration from the repository to their desktops. Changes can then made to this local copy. Users have the ability to synchronize their local copy of the configuration with the master copy that is stored on the server, by selecting changes to be updated on the server, or changes from the server to update their local copy.

IMPORTANT: While CVS features allow multiple users to work on a configuration simultaneously, external communication and coordination between configuration tools users must occur in order to avoid configuration data loss or corruption. The last change that is synchronized with the server is what wins. For example, suppose a user checks in a configuration after making changes to hierarchies. Shortly thereafter, another user checks in the same configuration after also making changes to hierarchies. The changes the second user made will overwrite the changes the first user made.

Setting up source control

To set up source control:

1. From the File menu, select Properties.
2. Click on the Source Control tab.
3. Type 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.
4. Click **OK** to save your changes

Adding a configuration

To add a configuration to CVS:

1 From the File menu, select Source Control, and then select Add Configuration.

2 In the Workspace Name field, type the name of the configuration that you want to add to the repository.

3 In the Workspace Directory field, type the name of the directory containing the configuration you want to add to the repository, or click the Browse button to select the directory.

Note: If you type or select the wrong directory, the message “Workspace does not exist” is displayed, and the OK button is deactivated.

4 Click **OK** to add the configuration to the CVS repository.

Importing a configuration

To import a configuration from CVS:

1 From the File menu, select Source Control, and then select Import Configuration.

2 In the Workspace Name field, type the name of the configuration that you want to import from the CVS repository.

Note: You must type the name *exactly* as it was entered when it was added to the repository.

3 In the Workspace Directory field, type the name of the directory on your local machine where you want to save the configuration, or click the Browse button to select the directory.

4 Click **OK** to import the configuration from the CVS repository.

Committing changes to a configuration

To commit changes to a configuration in CVS:

1 In the Configuration Manager window, select the configuration whose changes you want to commit.

2 Save your configuration. See Saving a configuration in Chapter 4 for more information.

3 From the File menu, select Source Control, and then select Synchronize Configuration.

4 In the Version Control Update window, select the changes that you want to commit to the repository. Eligible changes are identified by a right arrow (➔). To select more then one change at a time, press and hold the Ctrl key while selecting the changes.

Note: A double arrow (←→) indicates that a change has been made in the repository and on your local machine. You must decide how to proceed. You can commit your change, in which case the version in the repository will be overwritten. Alternately, you can update your local configuration with the latest changes from the repository, in which case the changes you made on your local machine will be overwritten. See Updating your local configuration for more information.

5 Press the Commit button to save your changes to the repository.

Updating your local configuration

To update your local configuration with the latest changes from the repository:

1 In the Configuration Manager window, select the configuration whose changes you want to commit.

2 Save your configuration. See Saving a configuration in Chapter 4 for more information.

3 From the File menu, select Source Control, and then select Synchronize Configuration.

4 In the Version Control Update window, select the changes that you want to save to your local configuration. Eligible changes are identified by a left arrow (←). To select more then one change at a time, press and hold the Ctrl key while selecting the changes.

Note: A double arrow (←→) indicates that a change has been made in the repository and on your local machine. You must decide how to proceed. You can update your local configuration with the latest changes from the repository, in which case the changes you made on your local machine will be overwritten. Alternately, you can commit your change, in which case the version in the repository will be overwritten. See Committing changes to a configuration for more information.

5 Press the Update button to update your local configuration with the changes you selected.

6 To see the updates from the repository, save your configuration, close it, and reopen it.

Chapter 4 – Configuration Manager

Overview

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

Working with configurations

A configuration represents the overall workspace for the RPAS implementation. Each configuration contains one or more projects, which in turn contain one or more solutions.

Creating a new configuration

To create a new configuration:

1. From the File menu (or via the right click menu), select New, and then select Configuration.
2. Type the name of your new configuration in the text box.
3. If you want to save the new configuration in the default directory, go to Step 5. Otherwise, uncheck the Use Defaults check box.
4. Type the name of the directory in which to save the new configuration. Alternately, click Browse to choose a directory.
5. Select the language that the configuration will be configured in from the drop-down list. This is the standard list that RPAS supports for translation purposes.

   Note: The default language is English.

6. If you want the configuration to support multiple languages, check the MultiLanguage check box.

   Note: You can enable or disable multiple language support at any time. For details, see Modifying language information in Chapter 3. Language support information is only used when an RPAS domain is first built, however. Changes made to MultiLanguage field are ignored if you do a patch installation. For more information, see Chapter 12 – Building an RPAS domain.

7. Click OK to save the new configuration in the specified directory.

Opening an existing configuration

To open an existing configuration:

1. From the File menu (or via the right click menu), select Open.
2. Browse the directory structure until you locate the configuration you want to open.
3. Click **OK** to open the selected configuration.
   OR
   From the File menu, select a configuration from the displayed list of recently used configurations.

Saving a configuration

To save a configuration:

1. From the File menu, select Save. Any changes you have made to the configuration are saved.

Cloning a configuration

You can create an exact copy of an existing configuration by cloning it. To clone a configuration:

1. From the Configuration Manager right-click menu, select Clone.
2. Type the new name for the configuration in the text box.
3. Click OK to save the configuration under the new name.

Renaming a configuration

To rename a configuration:

1. From the Configuration Manager right-click menu, select Rename.
2. Type the new name for the configuration in the text box.
3. Click OK to save the configuration under the new name.

Working with projects

A project corresponds to an RPAS domain. Each configuration can have one or more projects. For example, you can create separate project (and, therefore, separate domains) for Planning and Forecasting.

Creating a new project

To create a new project:

1. From the File Menu or Configuration Manager right-click menu, select New, and then select Project.
2. Type the name of the new project in the text box.
3. Select the workspace (configuration) in which to create the new project.
4. Click OK to create the new project in the specified workspace.

Cloning a project

You can create an exact copy of an existing project by cloning it. To clone a project:

1. Select the project that you want to clone.
2. From the Configuration Manager right-click menu, select Clone.
3. Type the new name for the project in the text box.
4. Select the workspace (configuration) in which to create the cloned project.
5. Click OK to save the cloned project in the specified workspace.

Renaming a project

To rename a project:

1. Select the project that you want to rename.
2. From the Configuration Manager right-click menu, select Rename.
3. Type the new name for the project in the text box.
4. Click OK to save the project under the new name.

Moving a project

You can move a project from one configuration to another. To move a project:

1. Select the project that you want to move.
2. From the Configuration Manager right-click menu, select Move.
3. Type the name of the project in the text box.

   Note: This is the name the project is called after it has been moved.
4. Select the workspace (configuration) to which you want to move the project.
5. Click OK to move the project to the specified workspace.

Working with solutions

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.

NOTE: Curve, Forecast, and Promote solutions have configuration steps which are specific to those solutions. For more information, see the RPAS 11.0.4 Solution Extensions Guide.

Creating a new solution

To create a new solution:

1 From the File Menu or Configuration Manager right-click menu, select New, and then select Solution.

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

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

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

Renaming a solution

To rename a solution:

1 Select the solution that you want to rename.

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

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

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

Cloning a solution

You can create an exact copy of an existing configuration by cloning it. To clone a solution:

1 Select the solution that you want to clone.

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

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

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

5 Click Finish to save the cloned solution in the specified project.

Moving a solution

You can move a solution from one project to another. To 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.

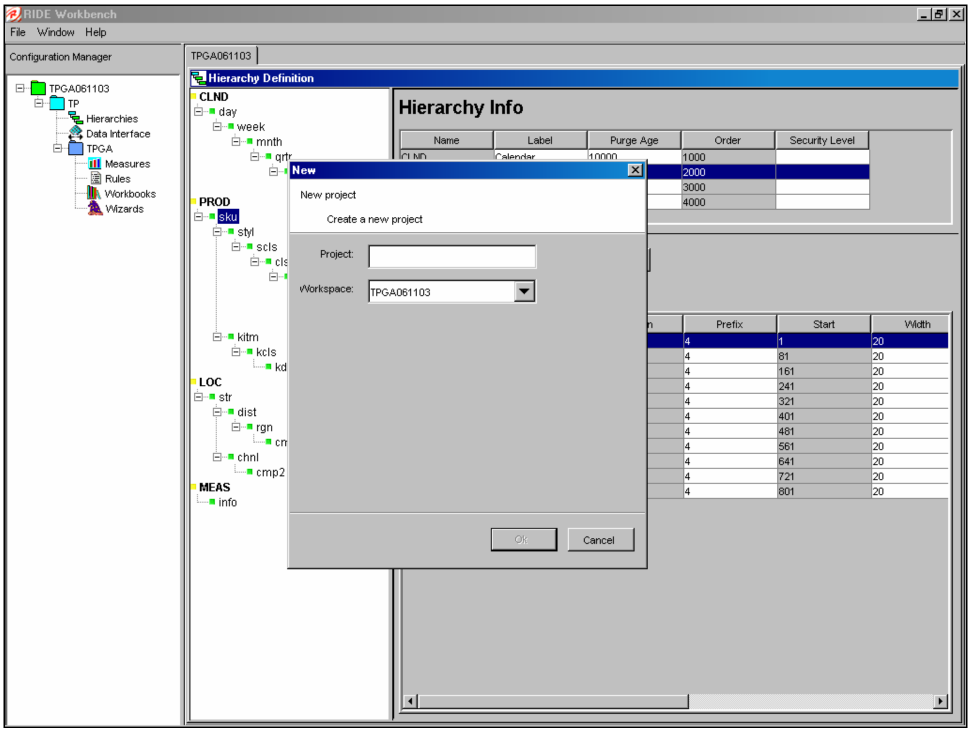
Note: This is the name the solution is called after it has been moved.

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

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

Example

The following example shows the dialog box for adding a new project to the TPGA061103 configuration.



Chapter 5 – Hierarchies

Overview

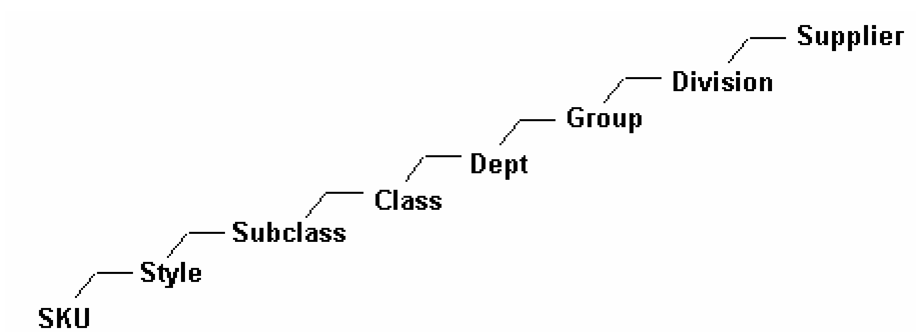
The hierarchy definition tool allows you to define and construct hierarchies, dimensions within those hierarchies, and relationships between dimensions.

When you create a project, the calendar, product, location, and measure hierarchies are automatically created by default. These hierarchies are required by RPAS and cannot be deleted, nor their hierarchy order changed. The order indicates the order in which the hierarchies are expected in the file used for hierarchy data loading. Additional hierarchies can be created as well. Dimensions for each of the hierarchies can then be defined. These hierarchies and dimensions are used for all solutions within the project. Therefore, two or more solutions can only be in the same project if they use the same hierarchy structure.

Understanding hierarchies and dimensions

Hierarchies are the structures that an organization uses to describe the relationships between and among the many dimensions.

The following illustrates an example of a product hierarchy and the dimensions within that hierarchy.

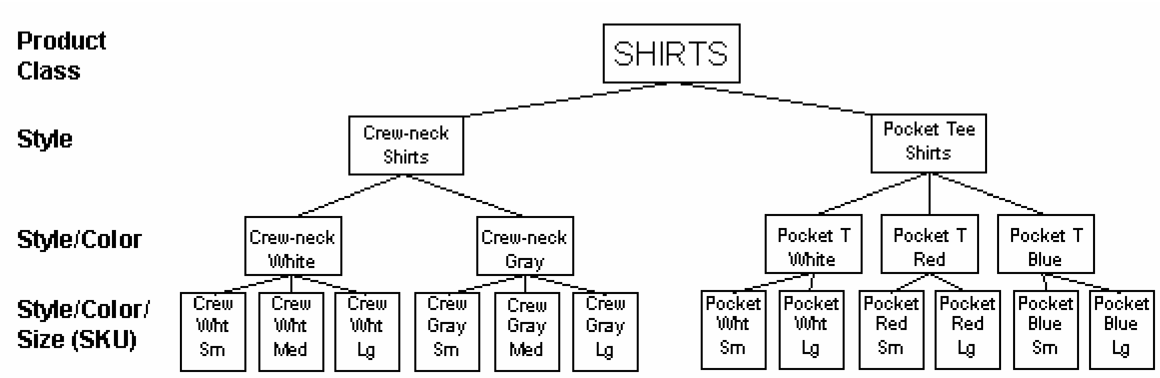


Merchandising system product hierarchy structure

Dimensional level relationships in a product hierarchy

SKU, style, and class are examples of specific dimensions that define the relationship between individual products. For this reason, SKU, style, and class are said to be dimensions of the product hierarchy.

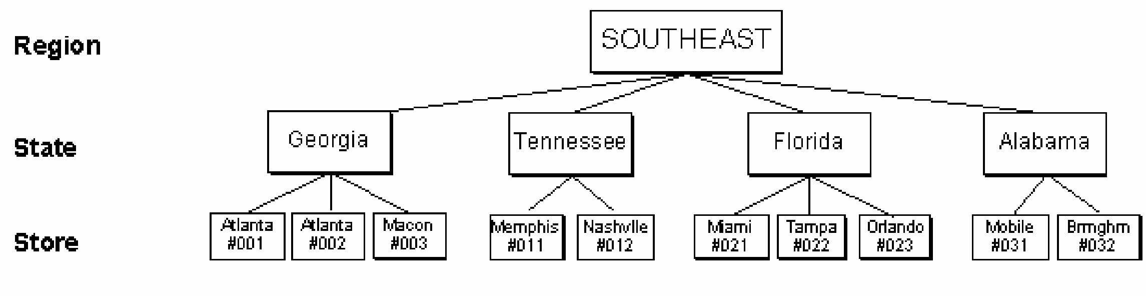
For example, consider an apparel manufacturer where shirts are one of the company's many product classes. The following figure shows a possible relationship between dimensional levels in the company's product hierarchy.



The items at the bottom of the hierarchy can be defined at their lowest level by SKU. Items at each successively higher level in the hierarchy contain the elements directly beneath; that is, lower level components in the hierarchy roll up (or aggregate) into higher ones. In this product hierarchy, shirts at the SKU level roll up into aggregate style/color groupings (for example, white crew-necks). Shirts at the style/color level can then be rolled up to the style level (for example, crew-necks); and styles can be rolled up into the product class level (for example, shirts).

Dimensional level relationships in a location hierarchy

This company can represent its hierarchy of locations in a similar fashion. In the following location hierarchy, stores are identified at their lowest level (the Store dimension) by city/store number. This hierarchical level rolls up into the state dimension, and the state level can be further aggregated into the region.



Hierarchy rules (alternate hierarchies)

Certainly, hierarchies can be much more complex than those in the examples. Note that an item at a particular level in a hierarchy can be rolled up along more than one hierarchical path (that is, an alternate hierarchy). However, for any given path of rollup, that item can only belong to one parent at any higher dimensional level. For example, if a given store location is rolled up to the state level, it can only belong to one state position (for example, Georgia or Florida, but not both). You can view data at any level of detail you want by drilling down or rolling up through levels in the hierarchy. Hierarchies define the path of data aggregation and spreading.

Working with hierarchies

Creating a new hierarchy

To create a new hierarchy:

1 From the Hierarchy Definition right-click menu, select New Hierarchy.

2 Use the up and down arrows to move the hierarchy to the desired location.

Note: The Order column indicates the order in which the hierarchy information is expected in the file used for hierarchy data loading.

The order of the default hierarchies of Calendar, Product, Location, and Measure cannot be changed.

3 To change the name of the newly created hierarchy, click the hierarchy name twice and type the new name (or change the name in the hierarchy info table).

Note: Hierarchy names can contain a maximum of four characters.

Specifying hierarchy information

To specify hierarchy information:

1 In the Hierarchy Definition window, click the name of the hierarchy for which you want to specify information or select it in the hierarchy info table.

2 In the Hierarchy Info frame, type the following:

- Name – The RPAS internal name of the hierarchy. The hierarchy name can contain a maximum of four characters.
- Label – The hierarchy label that is displayed to RPAS users.
- Purge Age – The number of days (without a load) before a position is purged from a dimension in this hierarchy.
- Security Level – If position-level security is enabled, specifies the name of the dimension for which security is enabled.

Note: The Order column indicates the order in which the hierarchy information is expected in the file used for hierarchy data loading.

Deleting a hierarchy

To delete a hierarchy:

1 In the Hierarchy Definition window, select the hierarchy that you want to delete.

2 From the Hierarchy Definition right-click menu, select Delete.

3 Click Yes in the Confirm Delete dialog box to delete the hierarchy.

Renaming a hierarchy

To rename a hierarchy:

1 In the Hierarchy Definition window, select the hierarchy that you want to rename.

2 From the Hierarchy Definition right-click menu, select Rename (or just change it in the hierarchy info table).

3 Type the new name for the hierarchy.

Note: Hierarchy names can contain a maximum of four characters.

OR

Click the hierarchy name twice, and then type the new name.

Note: Hierarchy names can contain a maximum of four characters.

Working with dimensions

Creating a new dimension

To create a new dimension:

1 From the Hierarchy Definition window, select the hierarchy or dimension under which to create the new dimension.

Note: You can create only one top-level dimension per hierarchy.

2 From the Hierarchy Definition right-click menu, select New Dimension (you can also use the insert key).

Specifying dimension information

To specify dimension information:

1 In the Hierarchy Definition window, select the dimension for which you want to specify information (or select the dimension in the Dimension Info table).

2 In the Dimension Info frame, specify the following information, which is used during the data loading process:

- Name – The RPAS internal name for the dimension. The dimension name can contain a maximum of four characters.
- Label – The dimension label that is displayed to RPAS users.
- Prefix – Number of characters (0 to 4) for an automatic prefix 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 then the left-most characters from the dimension name are used.

  Note: The Day dimension must have a prefix of 3.
- Width – Width of position names for this dimension in the hierarchy load file.
- Label Width – Width of position labels for this dimension in the hierarchy load file.
- Database – The name of the database in which the dimension information is stored.

These fields are non-editable and provide the following information:

Attribute	Description
Column	Identifies the column in the hierarchy load file.
Start	Identifies the start position of the dimension name in the hierarchy load file.
Label Start	Identifies the start position for the label in the hierarchy load file.
Aggs	Identifies the dimension that the current dimension aggregates.
User Dim	Indicates whether or not the dimension is user-defined.

Deleting a dimension

To delete a dimension:

1. In the Hierarchy Definition window, select the dimension that you want to delete.
2. From the Hierarchy Definition right-click menu, select Delete.
3. Click **Yes** in the Confirm Delete dialog box to delete the dimension.

Renaming a dimension

To rename a dimension:

1. In the Hierarchy Definition window, select the dimension that you want to rename.
2. From the Hierarchy Definition right-click menu, select Rename.
3. Type the new name for the dimension.

   Note: Dimension names can contain a maximum of four characters.

   OR

   Click the dimension name twice, and then type the new name.

   Note: Dimension names can contain a maximum of four characters.

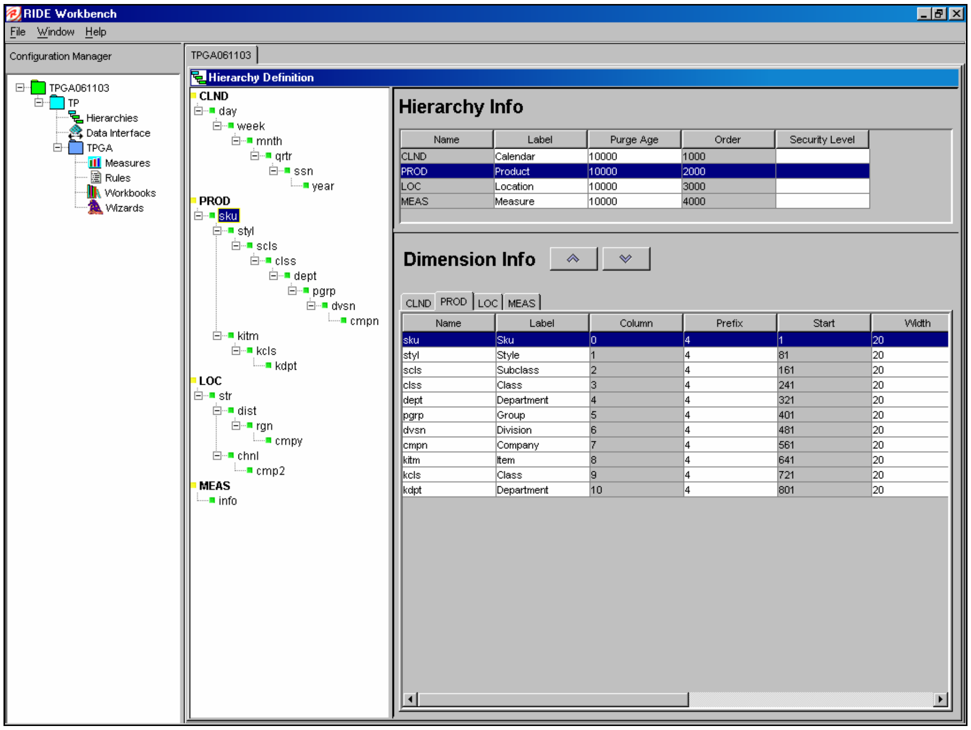
Creating an alternate hierarchy

To create an alternate hierarchy:

1. In the Hierarchy Definition window, select the dimension from which you want to create an alternate hierarchy.
2. From the Hierarchy Definition right-click menu, select New Dimension (you can also use the insert key).

Example

In the following example, the Calendar, Product, Location, Hardware, and Measure hierarchies are displayed. Dimension information for the Calendar hierarchy is shown in the Dimension Info window.



Name	Label	Purge Age	Order	Security Level
CLND	Calendar	10000	1000	
PROD	Product	10000	2000	
LOC	Location	10000	3000	
MEAS	Measure	10000	4000	

Name	Label	Column	Prefix	Start	Width
sku	Sku	0	4	1	20
styl	Style	1	4	81	20
scls	Subclass	2	4	161	20
clss	Class	3	4	241	20
dept	Department	4	4	321	20
pgrp	Group	5	4	401	20
dvsn	Division	6	4	481	20
cmpn	Company	7	4	561	20
kitm	Item	8	4	641	20
kcls	Class	9	4	721	20
kdpt	Department	10	4	801	20

Chapter 6 – Measures

Overview

The measure definition tool allows you to define classes of measures and specify default measure properties for each class. After you have defined the class structure, the measure definition tool auto-generates prototype measures by combining the classes you created. You can also specify properties for individual measures, thereby overriding the class default properties for those measures.

Concepts

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

Within each major class, you create one or more minor classes. You can also create minor classes underneath minor classes. You can also modify properties at the minor class level. Measures are auto-generated by combining the minor classes at the lowest level (that is, those minor classes with no further minor classes beneath them).

After you have defined your major and minor classes, the measure definition tool auto-generates prototype measures based on the combination of classes you have created. These prototype measures cannot be used elsewhere in the configuration tools until you have realized them. See Realizing and unrealizing measures.

Class process

1. Create major classes, which are measure components.
2. Create minor classes, which are sub-groupings or specific items in a major class.
3. Define measure properties at the major class level. The minor classes will inherit the properties associated with the major class they belong within.
4. If necessary, modify the measure properties at the minor class level. After you create major and minor classes, the configuration tool creates prototype measures by creating all the possible combinations between the classes. The auto-generation of measures saves you from having to create all the measures manually.
5. Because all possible measures are created from the combinations of classes, you must identify those that are either invalid or will not be used in a solution. Measures that you invalidate are moved to the Invalid Measures tab. These measures cannot be used in the configuration, but are not lost in case you later decide to validate them so they can be used.
6. The last step is to realize measures, thus allowing them to be used in other parts of the configuration.

Measure classification design

There are two basic principles to keep in mind to make the measure tool as powerful as possible.

First, major and minor classes should be designed with the idea of *maximizing default properties*. For example, if a configuration has 2000 measures, and 1500 of them are of data type *real*, then every effort should be made to see that the property *real* is only explicitly set a very small handful of times at the minor class level, and not 1500 times at the actual measure level. Measures are the composition of minor classes, and it is in those minor classes that defaults should be set.

The other major principle is to use minor classes to make measure definition manageable. For example, in the case in the previous paragraph, we established that we want to define the data type of *real* only a few times. What we want to avoid, though, is grouping all 1500 of these measures into a single minor class such that we cannot easily edit smaller subgroups of those 1500 measures. Remember that minor classes can also have minor classes, so within those 1500 measures, we could break them out further based on aggregation method, such as total, max, recalc, etc. Ideally, the “checking” of a lowest level minor class should allow the user to easily view and manage every resulting measure within that minor class.

Measure Property Inheritance/over-writing defaults

Measure properties are inherited at the class level. Properties defined for a class are spread down to all of the minor classes that belong to that class, and also to the measures associated with that class. Properties can then be over-written either at the minor class or measure level. Note that once an property is changed at the measure level, changes made at the class level will no longer be inherited by that measure.

Working with classes

Creating a major class

To create a major class:

1 From the Measure Definition right-click menu, select Add Major Class.

Creating a minor class

To create a minor class:

1 In the Measure Definition window, select the class under which to create the minor class.

2 From the Measure Definition right-click menu, select Add Minor Class.

Displaying classes

By checking or unchecking the check boxes next to the class names, you determine which classes are displayed in the Class Defaults tab.

If you want to display information about a class in the Class Defaults tab, check the box next to the class name.

If you do not want to display information about a class in the Class Defaults tab, uncheck the box next to the class name.

Note: Checking or unchecking a box causes the boxes for all subclasses to be checked or unchecked, accordingly.

Specifying class information

To specify class information:

1. In the Measure Definition tool, select the class for which you want to specify information.
2. On the Classes tab, specify the following information applicable to the measures inheriting from the class.

   Note: The defaults entered here will be inherited by the child minor classes, and will also be inherited by the auto-generated measures. Not all properties need to be entered for all classes. Also, properties defined for measure classes listed fist will override properties for classes further down the list.

   - Identifier – The identifier of the class. The identifier is used to create measure names.
   - Label – The label that is used to generate measure labels, which are displayed to RPAS users.
   - Description – A description of the class.
   - Data Type – Must be one of the following types:
     - Real – Floating point numeric values. Most measures are of this type.
     - Integer – 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’.
     - Date – Date and time. Can easily be converted to position names by standard functions.
     - String – Variable length strings, typically used for notes and names.
     - Boolean – True or false values, typically used for flags and indicators.
   - NA Value – The suggested value for unpopulated (na) cells. Initially, the base array of a measure is defined with this value; however, the value is subject to change in order to optimize the compression factor.

- Base Intersection – Lowest level at which data is stored for a measure. Double clicking on this field will bring up a dialog box that contains the hierarchies that were defined using the Hierarchy tool. One dimension from each hierarchy can be selected.
- Agg Method – The default aggregation method for the class:
  - recalc – recalculate measure at each level, via recalc expression
  - none – measure cannot be aggregated
  - total – for numeric types, sum of all values
  - average – for numeric types, average of all values (requires total)
  - average_pop – for numeric types, average of populated values (requires total and popcount)min – minimum of all values
  - max – maximum of all values
  - min_pop – minimum of populated values
  - max_pop – maximum of populated values
  - pst – for numeric types, period start total (first along time dimension and total along other dimensions)
  - pet – for numeric types, period end total
  - first – first of all values in base (relevant for ordered dimensions)
  - last – last of all values in base (relevant for ordered dimensions)
  - first_pop – first populated value in base
  - last_pop – last populated value in base
  - and – for Boolean types, and of all values
  - or – for Boolean types, or of all values
  - ambig – ambig of all values (all values equal, otherwise ambig)
  - ambig_pop – ambig of all populated values
  - popcount – count of populated values in base
  - median – for numeric types, median of all values
  - median_pop – for numeric types, median of populated values (requires popcount)

- Spread Method – The default spread method for the measure:
  - prop – for numeric types, spread value proportionally (non-zero total) or evenly (zero total)
  - even – for numeric types, spread value evenly
  - even_pop – for numeric types, spread value evenly across populated cells
  - delta – for numeric types, increment each cell evenly
  - delta_pop – for numeric types, increment each cell evenly across populated cells
  - repl – for all types, replicate the value to each cell
  - repl_pop – for all types, replicate the value to each populated cell
  - ps_delta – for numeric types and pst agg methods, apply delta to starting period with decreasing effect towards ending period
  - pe_delta – for numeric types and pet agg methods, apply delta to ending period with decreasing effect towards starting period
- 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).
- Database – The database where the measure is stored in the domain. If the database is NA then the measure is not stored. Specifies the relative path from domain root. You should specify the database as "data/dbname" so the database is not built directly in the root directory of the domain.
- View Type – Indicates if the measure is calculated when viewed. If the view type is not *none*, then the measure must not be materialized (that is, they are not stored in workbook or domain) and must appear on the left hand side of one and only one expression in a rule group. Synchronized view types are maintained immediately at cell edit time.
  View types include:
  - none – measure is stored
  - view_only – measure is calculated when viewed (aggstate and basestate must be read-only)
  - sync_first_lag – period 1 from first measure (no calendar), periods 2..N from second measure 1..N-1 (lag) [eg. bop ⇔ os & eop]
  - sync_lead_last – periods 1..N-1 from first measure 2..N (lead), period N from second measure (no calendar) [eg. eop ⇔ bop & cs]
  - sync_first – get period 1 from measure (similar to first agg type along calendar dimension) [eg. os ⇔ bop]
  - sync_last – get period N from measure (equivalent to last agg type along calendar dimension) [eg. cs ⇔ eop]

- Sync With – A comma-separated list of measures used for synchronization. Depends on view type.
- Insertable – Indicates whether the measure is visible to user for inclusion in workbooks (used in addition to measure security). Possible values are true and false.
- 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.
- 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 – Not currently used.
-
- Lowerbound – A valid, realized measure name indicating the lowest value that a user can enter for this measure.
- Upperbound – A valid, realized measure name indicating the highest value that a user can enter for this measure. The Upperbound measure and the measure containing the Upperbound property must be of either type integer or real.
- Materialized – A materialized measure is one in which it is stored in the domain and calculated by the RPAS calculation engine. A non-materialized measure is one in which the measure is calculated “on-the-fly” during the fetch process of the RPAS client and thus not stored in the domain. The user does *NOT* have the ability to configure this measure property. The value of this property is *automatically* determined by the Tools Installer during the domain build process. The Installer evaluates all rule groups across all rule sets in order to determine the non-materialized candidates, such that they are globally in effect throughout the entire domain. In order for a measure to be non-materialized, it must meet the following requirements:
    - It cannot have a database. (The database field must not be populated.)
    - The Agg Method must be set to recalc.
    - The Base State and Agg State must be set to read, meaning it cannot be edited.
    - It cannot be the source measure in any rule calculation, meaning it cannot be on the RHS (right-hand-side) of a rule.

- It cannot be used as a parameter in a procedure or function.
- It must not be a sanity check bounding measure, meaning it cannot be defined as an Upperbound or Lowerbound measure for another measure.

Renaming a class

To rename a class:

1. In the Measure Definition window, select the class that you want to rename.
2. From the Measure Definition right-click menu, select Rename.
3. Type the new name for the class.

   Note: Changing the class name changes the class label, not the class identifier.

   OR

   Click the class name twice, and then type the new name.

   Note: Changing the class name changes the class label, not the class identifier.

Moving classes

To move a measure from one class to another:

1. In the Measure Definition window, select the class that you want to move.
2. Drag the class to its new location and release it.

Note: You cannot move a major class.

Working with measures

Based on the major and minor classes that were defined, the Measure Tool will auto-create and name measures based on the class combinations. The properties for these measures will be inherited from the class properties. Once realized, the properties can then be over-written for individual measures by entering them the same way as on the classes tab. Note that once over-written at the measure level, changes made at the class level will no longer spread down to that measure as it will retain the over-written value.

Realizing and unrealizing measures

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.

Realize a measure

1. In the Measure Definition window, select the classes containing the measures you want to realize.
2. In the Measures tab, for each measure that you want to realize, select the check box in the Realized column.

   OR

   If you want to realize all measures in the classes you selected, from the Measures tab right-click menu, select Realize All.

Unrealize a measure

1. In the Measure Definition window, select the classes containing the measures you want to unrealize.
2. In the Measures tab, for each measure that you want to unrealize, deselect the check box in the Realized column.

   OR

   If you want to unrealize all measures in the classes you selected, from the Measures tab right-click menu, select Unrealize All.

Renaming a measure

To rename a measure:

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

Note: The measure must be realized before you can rename it.2 Type the new name for the measure.

Showing measures

Show a measure

1 In the Measure Definition window, check the check boxes next to the classes whose measures you want to show.
OR
Select the classes containing the measures you want to show.

2 From the Measure Definition right-click menu, select Show.

Show all measures

1 From the Measure Definition right-click menu, select Show All.

Hiding measures

Hide a measure:

1 In the Measure Definition window, uncheck the check boxes next to the classes whose measures you want to hide.
OR
Select the classes containing the measures you want to hide.

2 From the Measure Definition right-click menu, select Hide.

Hide all measures

1 From the Measure Definition right-click menu, select Hide All.

Sorting measures

In the Measures and Invalid Measure tabs, you can sort measures by property value. To sort measures:

1 In the Measures or Invalid Measures tab, check the check box next to the property that you want the measures sorted by.

The measures are sorted according to the value of the property selected. The sort order is either alphabetic or numeric, depending on the property selected.

Pushing measures down

To push a measure down:

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

2 From the Measure Definition right-click menu, select Push Down.

The class is pushed down one level in the class hierarchy, and a new class is created to take the place of the pushed-down class.

Note: You cannot push down a major class.

Pulling measures up

To pull a measure up:

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

2 From the Measure Definition right-click menu, select Pull Up.

The class is pulled up one level in the class hierarchy.

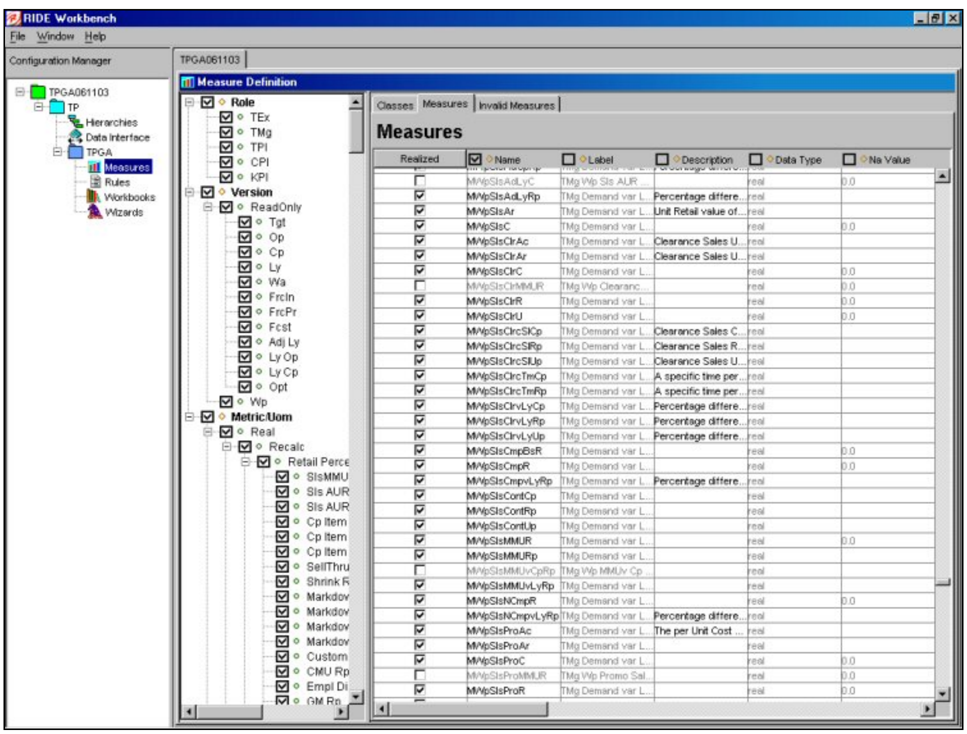
Note: You cannot pull a minor class up to become a major class.

Sorting measures by property

The checkboxes at the top of the Measures table allow you to sort the measure view. You can sort measures based on any of the properties in the table.

Example

In the following example, the measures associated with the checked classes are displayed. Realized measures are identified by a checkmark in the Realized column.



Chapter 7 – Rules

Overview

The rule definition tool allows you to define, organize, and manage rules, rule groups, and rule sets. It also allows you to create expressions and add expressions to rules.

Concepts

Understanding 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, and, at first sight, very complex. However, when the 'building blocks' of the calculation engine are properly understood, much of this apparent complexity goes away. This overview of the calculation engine processes will therefore start by describing the three fundamental processes of aggregation, spreading and expression evaluation, before explaining how the various processes integrate into a comprehensive whole.

RPAS supports an OLAP type model. In this model, individual pieces of data, named 'cells', apply to a single position in one or more hierarchies or dimensions. These will typically include a 'measures' dimension, a calendar or time hierarchy, and other hierarchies such as for products and locations. The measures dimension is fundamentally different to the other hierarchies, since measures (which in other systems may be referred to by terms such as facts, performance indicators or variables) represent the fundamental events or measurements that are being recorded, whereas the positions in the other hierarchies provide a context for the measurement: where, when, what, etc. Measures relate to one another through rules and expressions: positions in all the other hierarchies relate to each other through hierarchical relationships.

RPAS supports two different forms of relationships between cells: hierarchical relationships that require 'aggregation' and 'spreading', and measure relationships that require rules and expressions. 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. 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.

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.

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, where possible, all known ‘relationships’ between cells, for whatever reason, are always enforced. Much of the logic of the processing of rules and rule groups depends on this basic principal.

Expressions

Expressions are the basis of all calculations of the relationships between measures, and are evaluated by the calculation engine during a calculation. Expressions are written in a syntax that allows for the calculation of a single measure from other measures, constants and parameters, using standard arithmetical functions and a rich set of mathematical, technical and business functions. Expressions are therefore an algorithmic statement of a relationship between measures.

Rules

An expression describes the relationship between measures in a way that causes a measure to be calculated through the expression. An expression may be said to ‘solve’ the relationship for the measure that is calculated through the expression. In some cases, there may be business methodology reasons for wanting more than one of the measures in a relationship to be calculable or solvable through that relationship. To support this requirement, RPAS has the concept of a rule, which consists of one or more 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 (i.e. the measure on the left hand side of the expression that is calculated by the expression).

Where a rule has multiple expressions, those expressions are given a priority sequence, to help the calculation engine to select a calculation path that follows business priorities. Consider the rule that relates together sales value, sales units and sales price. Let us assume that there are three expressions in this rule, that is, that each of the measures involved in the rule may be ‘solved’ through the rule. If a user makes a change to, say, sales value, it should be clear that the calculation engine could enforce the mutual integrity of all the cells by holding the sales price constant and recalculating a new sales units. It could also achieve the same end by keeping the sales units constant, and recalculating the sales price. Both approaches are mathematically valid, and produce a consistent result, with complete data integrity. However, it is likely that one approach makes more ‘business sense’ than the other (in this case, most businesses in most circumstances would want the price to remain constant and have the units recalculated), and the prioritization of the expressions in the rule provides this information to the calculation engine. Considerable care should be taken in the design of models to ensure that appropriate expression priorities are established.

In essence, the calculation engine, when given a choice, will always select the highest priority expression in the rule that is available to be selected. In this example, therefore, the expression that calculates sales units would have a higher priority than the expression that calculates sales price. Similar consideration of the desired effect of a change to sales units will probably lead to a conclusion that the expression that calculates sales value would also have a higher priority than the expression that calculates sales price. But what of the relative priority of the expressions to calculate sales value and sales units? What is the ‘business priority’ for those expressions? That may vary from implementation to implementation. Indeed, it may even vary from one type of plan to another in the same implementation: for a financial merchandise plan, such as Merchandise Financial Planning, the preferred behavior may be that a change to just sales price causes a recalculation of the sales units, whereas in a unit oriented lower level plan such as a Item Planning, the preferred behavior may be that a change to just sales price causes a recalculation of sales value.

The same measure may appear in multiple rules. Indeed, this will frequently be necessary, since the same measure can be involved in a number of different relationships with other measures. For example, there may be a relationship between sales value, sales units and sales price. Sales value may also be involved in another relationship with closing stock and a cover value, and yet another with opening stock, receipts, markdowns and closing stock.

Rule groups

It is most unusual for a model to only require a single rule. In most cases, there will be a collection of relationships between measures that must be maintained. 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. The calculation engine always has one (and 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.

Rules within a rule group are given a priority. The calculation engine uses this to select a calculation path that follows business priorities, by using rule priorities to determine which rule to enforce when there is a choice to be made

Within a system as a whole, there may be many rules defined. The validation of rules is performed in isolation, but rules within a rule group are also validated in the context of all the other rules in the rule group. This can mean that a rule that is perfectly valid syntactically, is not valid within a particular rule group. Rule group validations include:

- Each rule in a rule group must represent a completely different measure relationship. Therefore no two rules in a rule group may use exactly the same collection of measures, and neither may one rule group 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 LHS of the only expression in a rule may not be on the LHS of any other expression.

Although there may only be one active rule group at any time, RPAS allows for the definition of multiple rule groups to satisfy different calculation requirements. Rule groups may be one of four different types: load, calculate, refresh and commit. The RPAS application automatically uses the load rule group when loading data into the workbook, the refresh rule group to refresh data and the commit rule group when committing data to the domain. These rule groups are perfectly ‘normal’, so although they will typically include many rules that use the master modifier to load or commit data, they may also have other rules. It is perfectly possible, for example, to commit data to the domain for a measure that does not exist in the workbook, merely 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, 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 calc rule group to another.

Rule group transitions

Although only a single rule group may be active at any time, RPAS supports the transition from one rule group to another. The calculation engine ensures the integrity of measure relationships at all times, so this process is not merely a case of switching from one rule group to another, as there is no guarantee that the integrity of the rules in the rule group being transitioned into have been maintained.

RPAS makes a worse case assumption when transitioning rule groups. Any rule that is in both the old and new rule groups is assumed to have its integrity maintained. Any other rule is assumed to be potentially wrong, and so is flagged as ‘affected’. A normal calculation is then initiated, with expressions to be evaluated determined by the usual process. All affected rules will therefore have their integrity imposed by the evaluation of an expression, and ‘knock-on’ effects may cause some rules that occur in both the old and new rule groups to also be evaluated. Since all base intersections must be calculated during rule group transition, a large or complex rule group transition is likely to take longer than a normal calculate.

Note that there are circumstances when 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. The measures to be refreshed will have been selected by the user. 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 calculate 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 will typically commit 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.

Understanding 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.

Refer to the *RPAS 11 Rule Functions* document for details about RPAS functions.

Working with rule sets

Adding a rule set

To add a rule set:

1. From the Rule Definition right-click menu, select Add and then select Rule Set.
2. Type the Name and Description of the rule set.

   Note: The name can contain a maximum of 16 characters, consisting of letters, numbers, and spaces only.

3. Click **OK** to create the rule set.

   The 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.

Deleting a rule set

To delete a rule set:

1. In the Rule Definition window, select the rule set that you want to delete.
2. From the Rule Definition right-click menu, select Delete and then select Rule Set.
3. Click **Yes** in the Confirm Delete dialog box to delete the rule set.

Renaming a rule set

To rename a rule set:

1. In the Rule Definition window, select the rule set that you want to rename.
2. From the Rule Definition right-click menu, select Rename and then select Rule Set.
3. Type the new Name and Description for the rule set.
4. Click **OK** to accept the new name and description.

Working with rule groups

Creating a rule group

To create a rule group:

1 In the Rule Definition window, select the rule set in which to create a new rule group.

2 From the Rule Definition right-click menu, select New and then select Rule Group.

3 Type the Name and Description for the rule group.

4 Click OK to create the rule group.

Deleting a rule group

To delete a rule group:

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 load, commit, calc, and refresh rule groups.

2 From the Rule Definition right-click menu, select Delete and then select Rule Group.

3 Click **Yes** in the Confirm Delete dialog box to delete the rule group.

Renaming a rule group

To rename a rule group:

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

2 From the Rule Definition right-click menu, select Rename and then select Rule Group.

3 Type the new Name and Description for the rule group.

4 Click **OK** to accept the new name and description.

Copying a rule group

To copy a rule group:

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

2 From the Rule Definition right-click menu, select Advanced-Copy Rule Group.

3 In the Rule Set field select the desired destination Rule Set.

4 In the Rule Group frame select the desired destination Rule Group.

5 Select Replace or Append. Replace will overwrite all rules that already exist in the destination rule group. Append will add to the rules already in the destination rule group.

6 Click **OK**.

Working with rules

Creating a rule

To create a rule:

1 In the Rule Definition window, select the rule group in which to create a rule.

2 From the Rule Definition right-click menu, select New and then select Rule.

3 Type the Name and Description for the rule.

4 If you want to add an expression to the rule, click Next. Otherwise, go to step 6.

5 To add an expression to the rule, do one of the following:

a. Enter the expression directly into the expression text box.

b. Click the Expression Builder button to the right of the text box.

See “Building an expression” for more information.

6 Click Finish to create the rule.

Building an expression

To build an expression:

1 Click the button to the right of the expression text box to open the Expression Builder.

2 In the Expression Builder window, select the desired measure classes to display measures. Based on the class selections made, realized measure will be displayed.

3 Select the measures you want to use in the expression by double clicking on them.

4 Using the selected measures, the functions available from the dropdown list, and the mathematical operators, construct an expression.

5 Click Finish to return to the previous window.

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

Rule/expression ordering

To order a rule or expression:

1 In the Rule Definition window, select the rule or expression that you want to move.

2 Click the down arrow to move the rule or expression down the list, or click the up arrow to move the rule or expression up the list.

Editing an expression

To edit an expression:

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

2 Select the rule containing the expression you want to edit.

3 From the Rule right-click menu, select Edit Expressions.

4 Either edit the expression in the text box, or click the button to the right of the text box to edit the expression using the Expression Builder.

See “Building an expression” for information on using the Expression Builder.

5 Click **OK** to accept the changes you made to the expression.

Adding an expression

To add an expression to a rule:

1 In the Rule Definition window, select the rule group containing the rule to which you want to add an expression.

2 Select the rule to which you want to add an expression.

3 From the Rule right-click menu, select Edit Expression.

4 In the Edit Expressions window, click Add.

5 Either type the expression in the text box, or click the Expression Builder button to create an expression using the Expression Builder.

   See “Building an expression” for information on using the Expression Builder.

6 Click **OK** to add the expression to the rule.

Deleting an expression

To delete an expression:

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

2 Select the rule containing the expression you want to delete.

3 From the Rule right-click menu, select Edit Expressions.

4 Click the Delete button next to the expression you want to delete.

5 Click **OK** to accept the changes you made.

Applying a rule pattern

To apply a rule pattern:

1 Select the rule to use as the pattern basis.

2 Select Apply Pattern from the right click menu.

3 A dialog box will be displayed listing candidate rules patterned after the selected rule. Select the desired rules.

4 Click **OK**.

Deleting a rule

To delete a rule:

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

2 Select the rule you want to delete.

3 From the Rule right-click menu, select Delete and then select Rule.

4 Click **Yes** in the Confirm Delete dialog box to delete the rule.

Renaming a rule

To rename a rule:

1 In the Rule Definition window, select the rule group containing the rule you want to rename.

2 Select the rule you want to rename.

3 From the Rule right-click menu, select Rename Rule.

4 Type the new Name and Description for the rule.

5 Click **OK** to accept the new name and description.

Filtering rules

To filter rules:

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

2 Select one of the following options:

- Disable Filtering – All rules are displayed.
- Filter by Measure – Works in conjunction with the measure classes box in the bottom left corner of the screen. Rules are filtered to show those whose measures conform to the selected class scheme.
- Filter by Size – Rules are filtered to show those with more than one expression.
- Filter by Validity – Only invalid rules are displayed.

Note: Only those rules satisfying the filtering criteria are displayed.

Auto generating load and commit rules

Rules for the load and commit rule groups can be auto-generated based on the calc rule group. To auto-generate load or commit rules:

1. Select the calc rule group to use as a basis.
2. Right mouse click to and select Advanced.
3. Select Generate Load Rules or Generate Calc Rules
4. Click **Yes** on the Non-Undoable dialog box. This will also clear out the undo stack.

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

Copying rules

Copy rules

1 Right-click in the Rule/Expression frame in the Rule Definition Tool.

2 Select Copy Selected Rules.

3 In the Rule Set field, select the desired destination Rule Set.

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

5 Select **Replace** or **Append**.

Replace will overwrite all rules that already exist in the destination rule group. Append will add to the rules already in the destination rule group.

6 Click the **Next** button.

7 Select the rules to be copied by selecting the check boxes for the desired rules.

8 Click **Finish**.

Find and replace measures in the copied rules

1 Click the **Find/Replace** button.

2 In the Find field enter the portion of the measure you wish to replace.

Note: The Find function is case sensitive.

3 In the Replace With field enter the measure replacement.

4 Select Forwards or Backwards to search the rules either forwards or backwards in the rule list.

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

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

OR

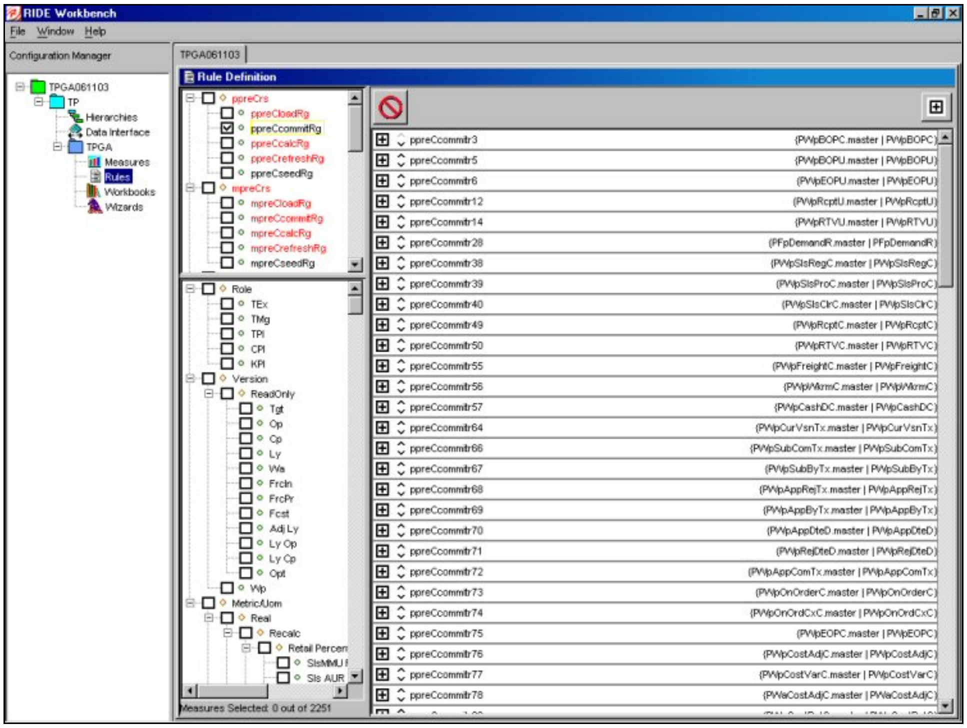
Click the **Replace All** button to replace all instances of the measure.

7 When you are finished, close the Find/Replace box.

8 Click the **Finish** button.

Example

In the following example, the rules in the *commit* rule group for the ppreCrs rule set are displayed. Because filtering is disabled, all rules in the selected rule group are displayed.



Chapter 8 – Workbooks

Overview

The workbook designer allows you to select and integrate the various components of a workbook template: tabs, worksheets, rule groups, wizards, and workflow processes.

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. 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.

A workbook tab is a major subdivision of a workbook. By default, each workbook contains one workbook tab, but it can contain many tabs.

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.

Wizards are used to build workbooks in the RPAS client. By specifying requested information on wizard pages, a workbook will be created. The workbook designer allows you to specify which wizards will be used to build the workbooks.

Working with workbooks

Creating a workbook

To create a workbook:

1. In the Workbook Designer window, click the New Workbook button.

   A workbook will be created with one tab and one worksheet.

Editing workbook properties

To edit workbook properties:

1. Select the workbook whose properties you want to edit.
2. Click the Edit Properties button and select the workbook name.
3. In the General tab, type the RPAS Name, User Label, and Group.
4. In the Rules tab, select the Rule Set to use with the workbook.
5. If you do not want to use the default rules associated with the rule set, uncheck the Use Default Rules check box. Then specify the Load Rules, Commit Rules, Calc Rules, and Refresh Rules for the workbook.

Note: If you are setting properties for the first time for a workbook, the next step you would want to take is assigning measures to worksheets. See Adding measures to a worksheet for more information.

6 In the Default Axis Layout tab, specify the default axis layout to use with all worksheets in this workbook. Hierarchies will be positioned in the worksheet based on this. You can override the defaults by specifying axis information in the worksheet properties.

7 In the Wizards tab, select either the Standard or Custom wizard.

Wizard Type	Procedure
Standard	1. Click on the **Advanced** button. Hierarchies that are used in the base intersections for the measures used in the rule set assigned to the workbook will be displayed. 2. Select the desired dimension from each hierarchy. This will specify the base dimension from which a Workbook can be built in the RPAS client.
Custom	If custom wizard layouts have been defined using the Wizard Designer, click on the pick list and select the custom wizard that you wish to be used to build the Workbook.

8 In the Custom Menus tab, use the right-click menu to add the custom menu. 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.

Property	Description
Menu Label	The name of the menu that will be displayed in the RPAS client.
Label	The label that will be displayed in the menu.
Function	Defaults to RuleGroupProcessor and cannot be changed.
Arguments	The rule groups to be used in the process, separated by commas.

Note: Duplicate label names are not allowed.

9 In the Hier Mods tab, for each hierarchy, enter the dimensions separated by commas.

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.

10 In 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

Property	Description
Name	The name of the dynamic hierarchy. **Note:** Duplicate names are not allowed.
Measure	Click on the Select Measure button 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 invalidated are displayed in red.

11 The measures tab allows you to override the following measure information at the workbook level: Label, Description, Base State, Agg State, UI Type, and Range. In the Measures tab:

a Click in the Identifier Column.

b Click the Select Measure button.

c Select the desired measure from the list of realized measures used in the rule set assigned to the workbook.

d Specify the desired property information.

12 If you want the workbook to be accessible through a wizard only, check the Wizard Only box.

13 Click **OK** to accept the changes you have made.

Removing a workbook

To remove a workbook:

1 Select the workbook that you want to remove.

2 Click the **Remove** button and select the workbook name.

3 Click **Yes** in the Confirm Delete dialog box to delete the workbook. Associated Worksheets and Tabs will also be removed.

Working with workbook tabs

Creating a workbook tab

To create a workbook tab:

1. Select the workbook in which to create a new workbook tab.

   Note: A single workbook tab is automatically created when you create a new workbook.

2. Click the **Create a New Workbook Tab** button.

Moving a workbook tab

To move a workbook tab:

1. Select the workbook containing the workbook tab you want to move.
2. Select the workbook tab that you want to move.
3. Click the Left Arrow or Right Arrow buttons, depending on where you want to move the workbook tab.

   Note: You must repeat this step for each position you want to move the workbook tab.

Editing workbook tab properties

To edit workbook tab properties:

1. Select the workbook containing the workbook whose properties you want to edit.
2. Select the workbook tab whose properties you want to edit.
3. Click the Edit Properties button and select the name of the workbook tab.
4. In the General tab, type the RPAS Name and the User Label.
5. Click **OK** to accept the changes you have made.

Removing a workbook tab

To remove a workbook tab:

1. Select the workbook containing the workbook tab you want to remove.
2. Select the workbook tab that you want to remove.
3. Click the **Remove** button and select the name of the workbook tab.
4. Click **Yes** in the Confirm Delete dialog box to delete the workbook tab.

Working with worksheets

Creating a worksheet

To create a worksheet:

1 Select the workbook in which to create a new worksheet.

2 Select the workbook tab in which to create a new worksheet

Note: A single worksheet is automatically created when you create a new workbook or workbook tab.

3 Click the **Create a New Worksheet** button.

Editing worksheet properties

To edit worksheet properties:

1 Select the workbook containing the worksheet whose properties you want to edit.

2 Select the workbook tab containing the worksheet whose properties you want to edit.

3 Select the worksheet whose properties you want to edit.

4 Click the **Edit Properties** button and select the name of the worksheet.

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

6 In the Base Intersection tab, select the measure for the base intersection.

7 In the Axis tab, specify which hierarchies should be assigned to each axis. This setting overrides the default axis information specified in the workbook properties dialog.

8 In the Position Queries tab, enter information for the following fields:

Note: Position queries allow the user 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.

- Name – The name of the property.

  Note: Duplicate names are not allowed.

- Label – Enter the label.
- Measure – Click on the **Select Measure** button 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.

9 Click **OK** to accept the changes you have made.

Removing a worksheet

To remove a worksheet from a workbook:

1. Select the workbook containing the worksheet you want to remove.
2. Select the workbook tab containing the worksheet you want to remove.
3. Select the worksheet you want to remove.
4. Click the **Remove** button and select the name of the worksheet.

Adding measures to a worksheet

To add measures to a worksheet:

1. Check the checkboxes next to the classes in the Selected Class column for the corresponding measures to add. The matching measures will appear in the Matching Measures column as the classes are checked. Only realized measures will be displayed.
2. Select the measures to add from the Matching Measures column.
3. Drag and drop the measure into either the Viewable or Default column.

   If a measure is added to the Default column it will also be added to the viewable column in bold font.

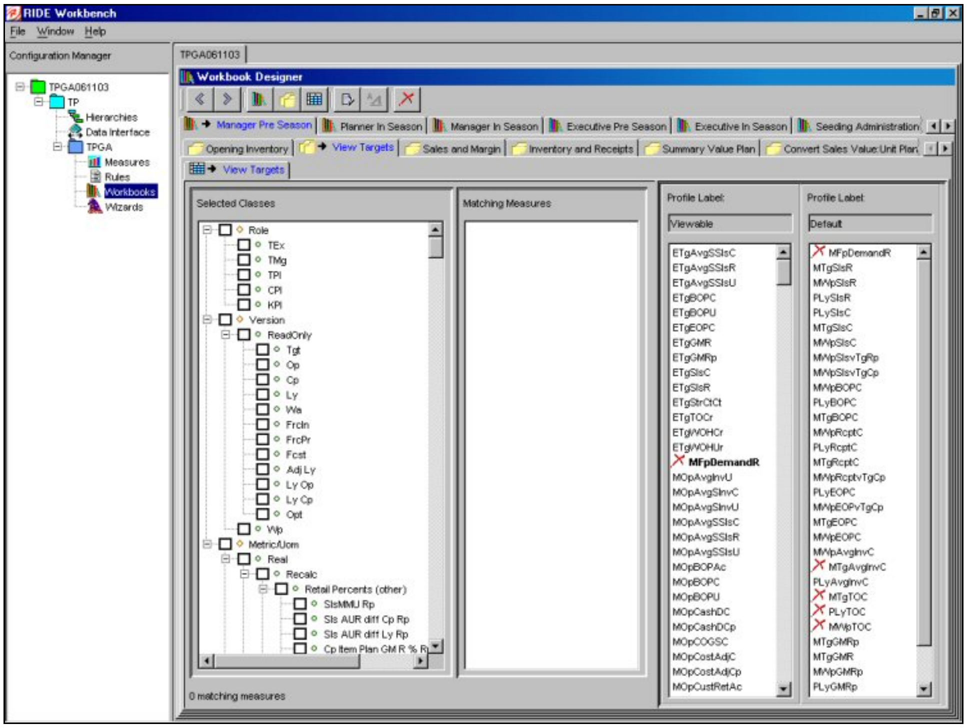
   The default profile contains the list of measures that will initially be displayed for this worksheet in the RPAS client. There must be at least one measure on the default profile or the worksheet will be invalidated in the Tools.

   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.

   There is also a third category of measures that cannot be viewed in the workbook designer – hidden measures. Measures that are considered hidden 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.

Example

The following example shows a solution with several workbooks. The View Targets tab on the Manager Pre Season workbook is open. The workbook tab has one worksheet, also called View Targets.



Chapter 9 – Wizards

Overview

The wizard designer allows you to define the graphical wizard layout for custom wizards.

Working with wizards

Creating a wizard group

To create a new wizard group:

1 In the Wizard Designer window, click the Create a New Wizard button.

Creating a wizard page

To create a new wizard page:

1 In the Wizard Designer window, click the Create a New Wizard Page button.

2 Click one of the following buttons, depending on which control you want to add:

Button	Description
Add a label	A non-editable text field.
Add a button	A button that performs an action when the user clicks it (for example, Save, Cancel, or Next).
Add a radio button	Used to select one option from a group of options, when only one can be specified.
Add a check box	Used to select from a group of options when one or more can be specified.
Add a combo box	The user can select from choices presented on a drop-down list.
Add a text box	A single- or multi-line text field that the user can edit.
Add a group box	A labeled box in which to group other wizard elements.
Add a tree	A drop-down listbox that displays hierarchical elements in a tree structure.
Add a two tree	A control consisting of two related tree controls. The left tree control is typically used to display all the possible selections that can be made from a hierarchy structure. The right tree displays only those elements of the left tree that are selected.

3 Click on the wizard page grid to place the selected control on the page.

4 If necessary, reposition the control by dragging it to the appropriate place on the grid.

Copying a wizard group

To copy a wizard:

1 On the wizard page, click the control whose properties you want to edit.

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

Copying a wizard page

To copy a wizard page:

1 On the wizard page, click the control whose properties you want to edit.

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

Editing control properties

To edit control properties:

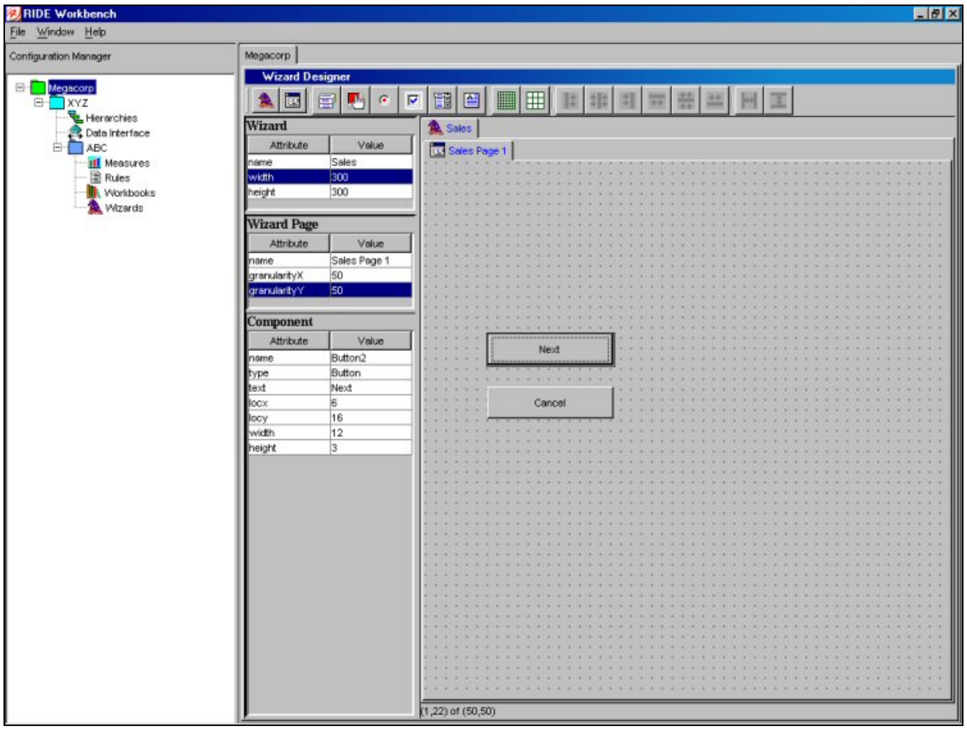
1 On the wizard page, click the control whose properties you want to edit.

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

- type – The control type.
- text – The text to be displayed on the control label.
- name – The RPAS internal name of the control.
- 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.

Example

In the following example, a wizard with one wizard page is displayed. At this point, the page has two buttons, Next and Cancel. The information for the wizard, wizard page, and selected component (Next button) is displayed in the left pane of the Wizard Designer window.



Chapter 10 – Using the Data Interface Tool

Overview

The data interface tool is used to specify the data import files from which measure data will be loaded.

Working with the Data Interface Manager

To specify where measure data is loaded from:

1. Click the **New Measure** button. The list of realized measures defined in the measure tool is displayed.
2. Select the desired measure and click **OK**.
3. If the measure load intersection is different from the measure base intersection, click on the Load Intersection field and select appropriate dimensions using the list boxes.
4. Enter the file name from which data for the measure will be loaded.
5. Enter the file start position for the measure data.
6. Enter the file column width for the measure data.

Chapter 11 – Configuration Utilities

Overview

The following utilities are stand-alone utilities that can be run outside of the Configuration Tools . The utilities provided include the Measure Extractor, the Rules Extractor, and the Function Manager, which are described in detail below. These utilities are not accessed from within the Configuration Tools user interface, but from the Configuration Tools directory in Windows Explorer.

Measure Extractor

Within the Configuration Tools build directory, in the *bin* subdirectory, there is a file called Measure Extractor.bat. You can run the Measure Extractor utility by double-clicking on this file.

As its name suggests, the Measure Extractor can be used to extract all measure information within the configuration to a flat file. Upon running this utility, the user is prompted to load a configuration and select the project and solution within the configuration. A file is then 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.

Rule Extractor

Within the Configuration Tools build directory, in the *bin* subdirectory, there is a file called Rule Extractor.bat. You can run the Rule Extractor utility by double-clicking on this file.

As its name suggests, 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. The Rule Extractor functions similar to the Measure Extractor. Upon running this utility, the user is prompted to load a configuration and select the project and solution within the configuration. A file is created in the root directory of the configuration called ruleout.txt. This file contains a listing of the rules within that solution, one expression per line, not necessarily one rule. The format of each line within the rule file is the name of the ruleset, the rule group, the rule, and finally the expression, separated by commas.

Function Manager

Within the Configuration Tools build directory, in the *bin* subdirectory, there is a file called Function Manager.bat. You can run the Function Manager utility by double-clicking on this file.

The Function Manager allows extension and custom function libraries to be linked into the RPAS JNI for validation purposes inside the Configuration Tools.

Once you run this utility, click on the Add button to specify a function library to use with the JNI. Enter the name of the library, such as Forecast, *without* the .dll or .so extension. Click on the Accept button. Close the utility. The custom function will now be validated when used in a rule inside the Rule Tool. This custom function must exist in the /applib directory of your RPAS_HOME variable setting. If necessary, you can also use this utility to remove custom function libraries.

Note: In order to use the Function Manager, the %RPAS_HOME%/applib directory must be defined on your system PATH environment variable. Please refer to the RPAS Installation Guide for more details.

Chapter 12 – Building an RPAS domain

Overview

This chapter provides detailed instructions for how to create an RPAS 11 domain using an existing configuration created using the configuration tools.

Because building an RPAS domain is currently a manual process, the person building the domain should be skilled in administering UNIX or NT servers and should have scripting skills.

This chapter contains the following sections:

- Prerequisites
- Client-side procedures
- Server-side procedures
- Loading measure data.

Prerequisites

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

- Installation of RPAS on your server. Refer to the *RPAS 11.0.4 Installation Guide* for more information.
- Installation of the Configuration Tools on your server. Refer to the *RPAS 11.0.4 Installation Guide* for more information.
- Pre-existing configuration built using the Configuration Tools

Client-side procedures

1 Create the Configuration.
To begin the domain build process, you need to start with a configuration 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. Once you have your configuration ready, continue with the following steps to build the domain.

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

2 Zip up the Configuration folder

a Find the location where you saved your configuration.

b Using Windows Explorer, go to the path of the configuration.

c Right-click on the Configuration folder, and select Add to Zip. You should zip up the entire contents of the Configuration beginning with the Configuration root folder such that the zip file will include the Project(s) and Solution(s). It is important that you zip up the entire Configuration for the entire directory structure and not just the specific .xml files.

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

Configurations – (directory for all configurations)

TPGA – (Configuration Name directory)

TP – (Project directory)

TPGA.xml – (Configuration .xml file)

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

Server-side procedures

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.

1 Unzip the configuration file

a Find the location where you want to put your configuration file.

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

b Move the <Configuration>.zip file to this location.

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

```
unzip –a <Configuration>.zip
```

2 Verify the environment variable settings.
Prior to beginning 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 11.0.4 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.

3 Get the input files ready.

a Designate a directory for the location of your input files, and move your files into this directory.
As a recommendation, Retek Services has used 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.

b 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.

4 Build the domain.

a 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>] [-am <measure_attribute>] –log
<logfile_name>
```

where:

Argument	Description
-version	Optional. Provides the version number of the Tools build as it relates to the release or patch number.
-testinstall	Optional. Used for testing, it just generates configuration files.
-fullinstall	Builds a full domain and loads the hierarchy files.
-patchinstall	Updates or unregisters and reregisters measures that have changed.
-cn config_name	Specifies the name of the configuration.
-ch config_home	Specifies the directory containing the root directory of the configuration.
-dh domain_home	Specifies the directory in which domains will be built (not the domain root).
-in input_directory	Specifies the name of the directory containing the input files.
-rf function_name	Optional. The filename of the function to register. It is case sensitive.
-log logfile_name	Specifies the name of the logfile.

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 itself. 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, please be sure to also 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.

rpasInstall options

Note: The **–testinstall**, **-fullinstall**, and **–patchinstall** options are all optional parameters, although one of them *must* be specified for the rpasInstall command.

–testinstall

The –testinstall option should only be used for a new domain. It only generates the configuration files, meaning it creates the domain shell with only the repos and setup directory. Because it does not build out the domain, it runs very fast. Within the setup directory, the configuration will be converted into the **init_hparm.mspl** file, which is a text version of the long-used standard procedure asetup.init_hparm, to be loaded into the domain.

–fullinstall

The –fullinstall option will build a pristine domain with only the hierarchy input files. It does not load measure data.

–patchinstall

The –patchinstall option will update the measure properties that are indeed updateable without impacting any previously loaded data.

Warning: For those measure properties that are **NOT** updateable, the –patchinstall option will unregister the measure and re-register it, thus losing any previously loaded data. Please refer to the list of measure -properties in the Applying Patches section as to which measure properties are updateable and which ones are not. The Configuration Tools will allow you to update all of theproperties, but the Installer will determine whether the measure must be unregistered or not depending on whether the property is updateable or not. The patchinstall option will not rebuild the domain. Please be aware that if you change a non-updateable measure property, and thus, the measure will be unregistered, all data that has been stored for that measure will be permanently deleted from the domain. Any data for that measure will have to be reloaded into the domain.

–cn config_name

The –cn config_name option is used to specify the name of the configuration (without the .xml extension) that you specified in the Configuration Tools. This is also known as the root directory for the configuration. So, in the example above, the name of the configuration is TPGA. For this option, you should *not* specify the path leading up to configuration directory name as that is specified using the –ch config_home option. The –cn config_name option is required.

–ch config_home

The –ch config_home option is used to specify the full directory path leading up to the configuration name. Do not include the configuration name as part of this option. This is the path that contains your configuration located on the server where you unzipped it, not including the configuration directory name. The –ch config_home option is required.

–dh domain_home

The –dh domain_home option is used to specify the directory path where the domain(s) will be built. This directory MUST already exist. Therefore, you must manually create it. Within the domain_home path, a directory will be created with the name of the configuration, and then within that directory, a subdirectory will be created for each Project. You do not specify the name of the domain root as it will use the name of the Project that you specified in the Configuration Tool to be the name of the domain root. The –dh domain_home option is required. We recommend that you do not assign the domain home path to be the same path as your configuration home. You should use a different directory path for the domain home for better organization of files.

–in

The –in option designates the location of the input_directory, which is the directory path that contains the hierarchy files to load or the measure registration file. The –test option should not be specified if the input_directory parameter is specified. The –in input_directoy option is required if the –fullinstall option is used. Otherwise, it is optional.

–rf register_function

The –rf register_function option is used to specify the name of the function that you want to register. This option should be used to register a shared object file that is located in the RPAS_HOME/applib directory and is not automatically validated by the RPAS JNI file. Currently, the Forecast procedure must be registered using this option if it is used in a rule within the configuration. When you specify the function name with this option, do not include the *lib* or *.so* part of the function name. The –rf option is an optional parameter.

–am attribute_measure

The –am attribute_measure option is used to specify the name of the attribute measure to register. The –am attribute_measure option is an optional parameter.

–log

The –log option designates the path and logfile_name that will contain output from the domain creation process. The logfile will be located in the directory from where the Installer is initiated, meaning the present working directory, if a path is not specified. There will be output from the Installer in the window from where the process is initiated. You can use the UNIX command: `tail –f logfile` to view the logfile as the process is executing.

At a high-level, the **rpasInstall** script performs the following functionality automatically:

- Creates a “pristine” domain or shell domain without any data or hierarchy files loaded – only contains source and application extension code
- Builds the hparm configuration file
- Builds the measure configuration file
- Builds the rule configuration file

- Executes abatch.create, which loads in hierarchy files
- Registers measures, rules, and templates

Note: The rpasInstall script only loads the hierarchy files and builds the domain. It does not load any measure data. The hierarchy files are copied to the **/input/processed** directory of the domain and appended with a time-date stamp.

You should execute this script in the path where your configuration is located (see Step 1 above). For ease of execution, we recommend creating a script to execute the rpasInstall command. Remember to change the permissions of the file to executable permissions (that is, 755). We recommend that you add lines before and after the rpasInstall call to set the usermode to single for better domain build performance.

For example, here is a script called **createDomain.sh**:

```
export USERMODE=SINGLE

rpasInstall –fullinstall -cn TopPlan –ch /test/Config
-dh /test/domains –in /test/to_rpas –log rpasInstall.log

export USERMODE=
```

Note: During the domain build or patch process, a directory called tmp will get created at the same level as the installs directory. The tmp directory will automatically be removed at the end of the process. Do NOT remove this directory during the execution of the process.

Note: If you are building a domain on NT, then in your script to run the Installer, you need to use forward slashes ‘/’ in all of the paths used in the parameters.

Validate Domain Build Results

Once the domain build process finishes, you should check the logfile to verify that the process executed successfully. You should 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
```

Applying Patches to a Domain

In order to patch your domain with changes to your configuration, use the –patchinstall option when running the Installer. Prior to beginning the patchinstall process, we recommend that users commit their workbooks as needed, and a backup copy of the domain is made.

As well, before beginning the process, verify that you are using the –patchinstall option in the rpasInstall command instead of the –fullinstall option. 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 not being able 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 setting. Please refer to the RPAS Administrative Guide for further details regarding the usage of the loadHier utility.

The patchinstall operation does allow for changes to some of the measure properties as indicated in the table below. As well, the patchinstall operation allows for changes to rule sets, rule groups, rules, and workbook templates. During the patchinstall process, the rule sets, rule groups, rules, and workbook templates are completely rebuilt.

Patching Measure Properties

The following table indicates those measure properties that can be updated via the patch process and those that will require the measure to be unregistered and reregistered.

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, but now they are guaranteed to be 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	Base types: int, real, string, date, boolean. The numeric representations of these types are 1, 2, 3, 4 and 6 respectively. **Note:** This is a required property for adding a measure.	Not updateable
navalue	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	Not updateable

Property	Description	Updateable
	change in order to optimize the compression factor.	
baseint	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. **Note:** This is a required property for adding a measure.	Not updateable
defagg	Default aggregation method (see Rule Functions document for master list) Note that changes to defagg will only be reflected in new or rebuilt workbooks.	Updateable unless set to recalc
defspread	Default spread method Note that changes to defspread will only be reflected in new or rebuilt workbooks.	Updateable
allowedaggs	Permitted aggregation methods. Syntax is as "MAX MIN TOTAL"	Updateable
basestate	Editable at base level: read, write. Default to read.	Updateable
aggstate	Editable at aggregate levels: read, write. Default to read.	Updateable
db	Database where measure is stored in domain (in future, a partitioned database which comprises many partitions). If database is NA then measure is not stored. It specifies the relative path from domain root. Syntax is as "data/test".	Not updateable
viewtype	Indicates if the measure is calculated when viewed. If the view type is not 'none' then the measure must not be materialized (ie. not stored in workbook or domain) and must appear on the LHS of one and only one expression in a rule group (ie. the view expression). Synchronized view types are maintained immediately at cell edit time. View types (and their corresponding integer values) include:	Updateable

Property	Description	Updateable
	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 from second measure 1..N-1 (lag) [eg. bop ⇔ os & eop] sync_lead_last (3)– periods 1..N-1 from first measure 2..N (lead), period N from second measure (no calendar) [eg. eop ⇔ bop & cs] sync_first (4) – get period 1 from measure (similar to first agg type along calendar dimension) [eg. os ⇔ bop] sync_last (5) – get period N from measure (equivalent to last agg type along calendar dimension) [eg. cs ⇔ eop]	
syncwith	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
refreshable	Indicates whether measure is refreshable in workbook	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 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.	Updateable
ui_type	Indicates whether the measure is a picklist or not.	Updateable
materialized	This property is not viewable or	Not updateable

Property	Description	Updateable
	updateable within the Tools Workbench. There are two possible values for materialized: Persistent: a "normal" measure Display: a measure that is only calculated when it is displayed (e.g. in the client). Only the displayed values are calculated, rather than the whole measure, which significantly improves performance with these measures. Display-only measures have three major restrictions: 1. Must have agg type "recalc" 2. Cannot appear on the RHS of any rule 3. Cannot be edited.	
lowerbound	A measure that provides minimum bounding values for this measure's data cells.	Not updateable
upperbound	A measure that provides maximum bounding values for this measure's data cells.	Not updateable
tokenmeas	Values are true or false to indicates if the measure is a token measure	Not updateable

Notes:

- When a recalc measure is registered, its aggstate and basestate are set to user-specified value no matter what is specified for defspread field.
- When a measure with defagg "PopCount" or "NobCount" is registered, this measure's defspread is set to NONE, aggstate is set to read only no matter what user specifies for this field.
- For measures that don't fit into the above descriptions, if their defspread is NONE, their aggstate is read only all the time.

Measure Loading Properties

Property	Description	Updateable
loadint	Intersection at which measure is loaded (for ovr, inc loads). Default to *baseint*. Changing loadint will make it impossible to back out previous data loads.	Not updateable
filename	File used to load measure (suffix of ovr, inc, clr). Default to *measure name*.	Updateable

Property	Description	Updateable
start	Start column of data (defaults to column after dimensions).	Updateable
width	Width of data (defaults by type). Note: For Boolean types, width can be >= 1, but only the first character is read.	Updateable
loadstokeep	History of loads (that may be backed out). Default to 1.	Updateable
loadagg	Used when aggregating from load intersection.	Updateable
stageonly	When true, data is not loaded into measure, but left in a staging array for custom processing. Default to *false*.	Updateable
clearint	Intersection at which to clear measure data (for clr loads).	Updateable
purgeage	Limits the history of data in a measure along the time dimension (purge occurs during load measure process). Default to *na*.	Updateable

Loading measure data

Note: This step 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.

The measure configuration is stored in the */repos/measures* directory of the domain in a file called **<*Solution*>_meas.cfg**. Similarly, the rules will be stored in a **<Solution>_rule.cfg** file and the templates will be stored in **<Solution>_tmpl.cfg**.

The configuration files will be used to register the measures when the domain is created.

As an interim solution, below is the process to load measure data.

1 Move the data files to the /input directory of the domain. They must have a valid measure filename, such as xxx.ovr, xxx.rpl, xxx.inc, or xxx.clr, where xxx is the measure name.

2 Use the ***loadMeasure*** utility to load your data. LoadMeasure should be in your path, meaning 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}

Argument	Description
–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.

For example:

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

After building the domain

After building the domain and loading measure data, the next steps are as follows:

- Use the RPAS Configure utility to denote the location of the domain on the server and designate a DomainDaemon port number. See The Configure utility in Chapter 11.
- Execute a DomainDaemon process with the port number specified in the Configure utility. Refer to the *RPAS 11.0.4 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 11.0.4 Administrator's Guide* for instructions on using these workbooks.

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. We recommend that you make a backup copy of your pre-existing configuration before using it with the new Tools release.   We also recommend that you create a new directory for the new Tools release rather than overlaying the old release. Refer to the RPAS Installation Guide for more details.

Chapter 13 – Configuring the RPAS client

After creating an RPAS domain and starting the DomainDaemon, you must configure the RPAS client to connect to the domain. This section provides instructions for configuring the Windows client and configuring the Web client. For details on installing the Windows client and Web client, refer to the *RPAS 11.0.4 Installation Guide*.

Configuring the Windows client

The Configure utility

Configure® is a Windows® application that configures the client/server communication for RPAS. Configure lets users specify communication parameters and produces files that are used as input to the client. These files must be in FCF (Foundation Configuration File) format. The files contain the necessary information for the client to start up the communication with the server. These files can be stored on the client machine or on the network. When the client is executed, a file named Foundation.FCF is expected in the same directory. If the file has a different name or if it is stored somewhere in the network, then the path to this file must be passed in as an argument to the client.

Configure consists of a menu bar, a main view and the advanced settings dialog box.

To launch Configure, double-click the Configure.exe file.

The Menu bar

The files produced by Configure may contain multiple connections. Each connection will be specific for a server with certain communication settings. Connections need to have unique descriptions, and they can be added and deleted using the menu bar.

Configure menu bar

The Main view

The main view has the basic connection parameters. On this view, three groups of controls are available:

- The connection group
- The domains group
- The Advanced Settings dialog

The Connection group

Field	Description
Database server	The hostname or the IP address of the server. Example: atldev03 or 10.2.1.23
Daemon port	The port number on which the domain daemon is listening. Must be an integer between 2048 and 65535. Example: 55278

The Domains group

Field	Description
Domain	The name of the domain. Select a domain from the list or type the name of a new domain and click Add Domain. You can delete a domain from the list by selecting it and then clicking Delete Domain.
Domain path	Domain path is the full path to the directory containing the domain. Example: /acufs001/atltest/topplan
User	The database user that is specified for the domain. Example:adm
Password	The database password that is specified for the domain. Example:adm

The Advanced Settings dialog

Default database login

Field	Description
User	The database (Acumate) user that will be used by the client if a domain specific user has not been entered. Example: adm
Password	Like the default database user, default database password will be used if a domain specific password has not been entered. Example:adm
Database port range	Port range is used to specify the range of ports on which the Acumate processes will be started by the DomainDaemon. The port start and port end fields are the lower and upper limits of this range respectively. These fields must be integers between 2048 and 65535. Example:Start: 40000End: 45000

Field	Description
Compression threshold	The number of bytes above which client and server will be using compression. Only advanced users should be manipulating this number.

Web tunnelling

Web tunneling is not currently supported in RPAS 11.0.

Proxy settings

Proxy settings are not currently supported in RPAS 11.0.

Language configuration

By default, the language of the RPAS client is determined by the language of the client-side operating system. You can override this behavior by setting the Language entry in the Options section of the foundation.ini file, like this:

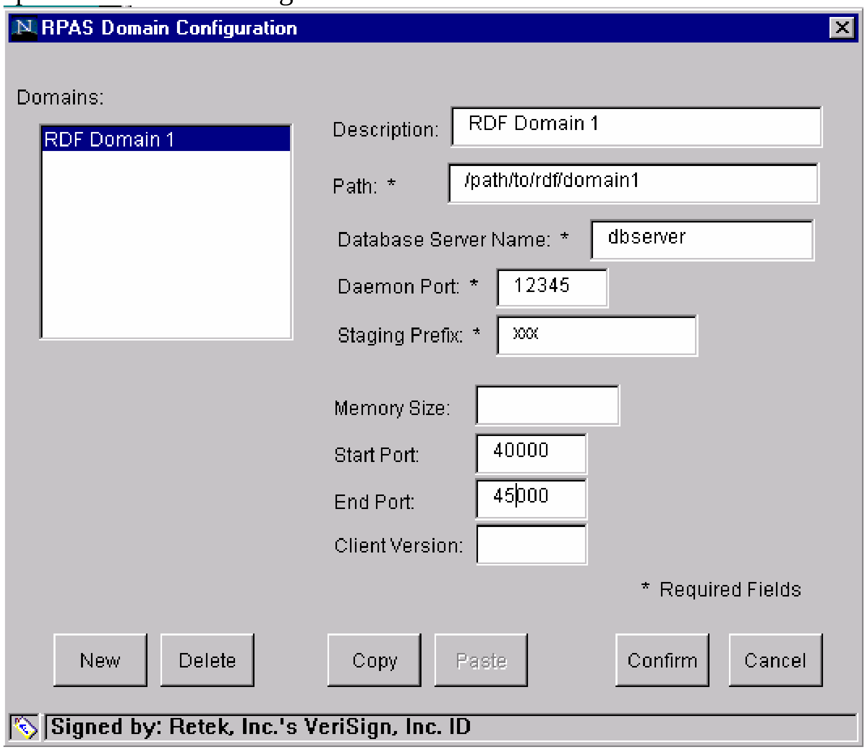
```
[Options]
Language=10
```

If the RPAS client cannot find the library for the language you specify, it will default to English. The following languages are currently supported:

Language	Code
Korean	18
English	9
Spanish	10
French	12

Configuring the Web client

1 Click **Domains** on the Administration Console to open the RPAS Domain Configuration window. This window allows specification of the location of RPAS domains. Each domain that can be accessed by a user must be specified with the dialog.



RPAS Domain Configuration Window

2 To add a new domain, click **New**, enter the following information, and then click **Confirm**.

- Description – This is displayed to a user when they are selecting a domain to log in to. Required.
- Path – The full path to the directory containing the domain on the database server. Required.
- Database Server Name – The hostname of the database server containing the domain. Required.
- Daemon Port – The port number of the DomainDaemon process running on the database server. Required.
- Staging Prefix – Only used by retail.com deployments. This can be set to anything you wish.
- Memory Size – Leave blank. Not used right now.
- Start Port – Start of the range of ports used by a client PC to connect to the database server. This must be > 1025. If not specified, The RPAS database server will try to find a free port whenever a client connects.
- End Port – End of the range of ports used by a client PC to connect to the database server must be < 65536.
- Client Version – Leave blank. Not used right now.

3 To change an existing domain configuration, select the domain from the Domains list, modify any of the fields, and click **Confirm**. If you click **Cancel**, any changes you have made are discarded.

4 If you click **Delete**, you delete the currently selected domain configuration.

5 To copy all of the domain settings, select a domain from the Domains list and click **Copy**.

6 To update the selected domains settings with the copied settings, select another domain and click **Paste**.

Appendix A – Translation configuration

Overview

String translation must be performed manually. This appendix describes the process for finding strings to be translated.

Procedures

The procedure for string translation within RPAS is in three parts:

1. Analyze the string in question to determine if it needs if it needs to be translated, and, if so, the best approach to take.
2. Place the string into the appropriate input file templates.
3. Insert the appropriate RPAS library invocations to dynamically build the string.

Analyzing the string

Follow these steps to analyze the string:

1. Determine if the message in question has direct meaning to the end-user (rather than just to RPAS developers or application developers). If so, the message should be translated.

   Note: If your schedule demands are such that you do not have time to prepare your string for translation, we suggest using the utility wrappers **LanguageManager::needToTranslate()** and **LanguageManager::englishOnly()** to temporarily designate your string. You can use other methods, such as special TODO comments that can be searched for later.

2. Are there simple variable substitutions for the string, such as names or numbers? If so, these can be parameterized using the ***%n*** notation. For example:

   ```
   String(firstName + lastName + " purchased " + amount + "
   shares today.")
   ```

   will eventually become

   ```
   %1 %2 purchased %3 shares today.
   ```

   in the messages overlay file.

3 If, however, your message string depends on certain conditions in order to be formulated, such as

```
String(firstName + lastName + " will be coming in " +
(arrivalTime <= 8 ? "early" : "late") + " today.")
```

it becomes a bit trickier to translate. In theory, one could turn **early** and **late** into separate resources, then develop the message as

```
%1 %2 will be coming in %3 today.
```

but this assumes that the language in question does not have a peculiar translation rule for what seems to us to be a polar opposite situation, for example:

```
Joe Smith will be coming in early today.
```

```
Joe Smith will be late coming in today.
```

Although unlikely, it is conceivable for non-English languages to use grammatical rules unfamiliar to English-speakers for even slightly more complex cases. So rewriting the string composition code as

```
arrivalTime <= 8 ?
```

```
String(firstName + lastName + " will be coming in early
today.") :
```

```
String(firstName + lastName + " will be coming in late
today.")
```

may be more appropriate.

Adding the string to resource files

1. In ***wherever_your_data_file_templates_are*/input**, there are two types of files that are specifically configured to handle localizable messages: **msgs.dat** and **r_msgt.ovr.*language***.

   Choose a unique identifier for your message string. Prepend an application-specific acronym to whatever ID you use (for example, ASO_NO_LOC_SELECT).

   Note: RPAS core message IDs are prepended with **R_** to delineate them from messages that others might add.

2. Add this identifier, along with a *short* description, to **msgs.dat**.

   Important: This input file has a fixed-width column format. Make sure you have enough trailing spaces at the end of the description to match the other entries already there. You can use `wc -lc filename` to return the number of lines and characters, respectively, in the file in question. Then, ensure that the resulting division does not have a remainder.

3. Add the message ID and corresponding resource string to the appropriate **r_msgt.ovr** file. Adding a resource here is slightly trickier. You are still dealing with a fixed-width file, except now the line lengths are considerably longer, exceeding the 80-column width that you might be used to in a terminal window or text editor.

4. If the string in question represents a *dimension label*, then you will only need to modify the appropriate **r_dimt.ovr.*language*** file, *not* the **r_msgt** files. These files are even trickier, as there are no spaces between the four-character dimension label ID and the label itself. The good news is that the label ID does not require a separate entry in a **dims.dat** file (or **msgs.dat**, for that matter).

Adding the C++ code to build the string

This *only* applies to messages. Dimension labels, as well as other types of RPAS object labels, are handled implicitly within the RPAS infrastructure.

1 If your source code does not include **<rpas/LanguageManager.h>** , add it.

2 Next, grab the message resource in question, using the message ID you previously created:

```
String msgTemplate =
LanguageManager::instance().getMessage("ASO_NO_LOC_SELEC
T");
```

3 If the resource didn't have any parameters, you're done. Otherwise, what you have is a string format, requiring further processing. For example, if your message required three separate parameters:

```
String::StringVectorT msgParms;

msgParms.push_back(firstName);

msgParms.push_back(lastName);

msgParms.push_back(amount.toString());

String msgFormatted =
msgTemplate.formatMessage(msgParms);
```

Glossary

configuration

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

dimension

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

expression

Expressions are the basis of all calculations of the relationships between measures, and are evaluated by the calculation engine during a calculation. Expressions are written in a syntax that allows for the calculation of a single measure from other measures, constants and parameters, using standard arithmetical functions and a rich set of mathematical, technical and business functions. Expressions are therefore an algorithmic statement of a relationship between measures.

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.

major class

A major class specifies the default measure property information that is inherited by all minor classes. Ultimately, this property information is inherited by the measures that are generated from the class 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 class

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

project

Each project within a configuration represents an RPAS domain. For example, configuration XYZ could contain Planning and Forecasting projects.

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.

solution

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

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. By default, each workbook contains one workbook tab, but it can contain many 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.

Index