

**PowerEditor Version 5.10**

**Reference Manual**

Revised: September 7, 2018

Notice: This document includes confidential and proprietary data that is within the scope of non-disclosure agreements between MindBox and its alliances. The document shall not be duplicated, used or disclosed -  
in whole or in part - for any purpose other than that specified within the scope of such an agreement. Acceptance of this document presumes adherence to the above conditions.

The PowerEditor documentation suite includes the following materials:

PowerEditor Business Analysts Guide

PowerEditor Custom Reports Guide

PowerEditor Installation and Customization Guide

PowerEditor Reference Manual

PowerEditor Release Notes

PowerEditor Rule Writers Toolkit

PowerEditor Web Service API Guide

*Release Notes* for earlier versions back to and including 4.0 are also provided in the release package. For technical assistance with upgrading or any other PowerEditor-related issue please contact CoreLogic Technical Support at 1-855.369.2410, or [ADESupport@corelogic.com](mailto:ADESupport@corelogic.com)

**Table of Contents**

1. Introduction 7

1.1 Overview 7

1.1.1 Major PowerEditor Components 8

1.2 Domain Model 9

1.3 Rules 9

1.4 Messages 10

1.5 Guidelines 10

1.6 Guideline Templates 11

1.7 Guideline Actions 11

1.8 Parameters 11

1.9 Parameter Templates 12

1.10 Entities 12

1.11 Context 12

1.12 Activation, expiration, and version control 12

1.12.1 Effective Date Range 12

1.12.2 Customizing Status 13

1.12.3 Status and Deployment 13

1.12.4 Pre-Dating Expiration Dates 13

1.12.5 Status-based Deployment authorization 14

2. Using PowerEditor 14

2.1 Starting PowerEditor 14

2.2 Screen and Window Management 15

2.3 Views 16

2.3.1 Tree Views 16

2.3.2 Table Views 17

2.3.3 Grid Views 18

2.3.3.1 Editing Grids 18

2.3.3.2 Editing Grid Rows 18

2.3.3.3 Copying a Row in the Grid 18

2.4 Edit Features 19

2.4.1 Drop-down Lists 19

2.4.2 Lookup Fields 19

2.4.3 Date Fields and Date Synonyms 20

2.4.4 Multi-Select Boxes 21

2.4.5 Command Icons 22

2.5 Deployment Status 23

2.6 Deletion Confirmation 23

2.7 Exiting PowerEditor 24

3. Entities 24

3.1 Entity Search 25

3.1.1 Filter 26

3.1.1.1 Executing the Search 26

3.1.2 Navigate 27

3.2 Manage Categories 28

3.2.1 Add, Edit, and Delete Product Categories 30

3.3 Managing Entities 31

3.3.1 Entity Details 31

3.3.2 Editing an Entity 32

3.3.3 Viewing an Entity 33

3.3.4 Creating a new Entity 33

3.3.5 Deleting an Entity 33

3.4 Managing Compatibility 34

Creating a New Compatibility Relationship 34

3.4.1 Editing or Deleting Existing Compatibility Relationships 35

4. Guidelines 35

4.1 Manage Templates 35

4.1.1 Create a New Template 36

4.1.2 Copying an Existing Template 37

4.1.3 Create a New Version of a Template 37

4.1.4 Editing Existing Templates 38

4.1.4.1 Description Tab 38

4.1.4.2 Completeness and Consistency 39

4.1.4.3 Columns Details Tab 39

4.1.4.4 Entity List Columns 42

4.1.4.5 Managing Column Messages 43

4.1.4.6 Rules & Messages Tab 44

4.1.5 Removing Existing Templates 45

4.1.6 Authoring Template Rules 45

4.1.6.1 Creating a Condition 46

4.1.6.2 Moving Rule Elements 49

4.1.6.3 Using Logical Operators 49

4.1.6.4 Using Exist Expressions 50

4.1.6.5 Removing a Condition 51

4.1.6.6 Selecting a Guideline Action 51

4.1.7 Authoring Messages 53

5. Managing Guidelines 54

5.1 Search Policies 54

5.1.1.1 Date Synonyms 57

5.1.2 Context Selection Search 57

5.1.2.1 Using the Include Controls in a Context Search 58

5.1.3 Template Selection Search 59

5.1.4 Using Search Results 60

5.1.5 Guideline Report 61

5.2 The Manage Guideline Screen 62

5.2.1 Context Selection 63

5.3 Associating Guidelines and Entities 63

5.3.1 Guideline Type Selection – Template Reference Specification 67

5.3.2 Guideline Selection 68

5.3.3 Guideline Detail Screen 70

5.3.3.1 Selected Guideline Template 70

5.3.3.2 Selected Guideline Context 71

5.3.3.3 Guideline Details 71

5.3.3.4 Activations 71

5.3.3.5 Rules 72

5.3.3.6 Deployment Status 72

5.3.3.7 Comments 72

5.3.3.8 Edit Guideline Toolbar Controls 72

5.4 Guideline Editor Operations 73

5.4.1 Editing Guidelines 73

5.4.2 Viewing Guidelines 73

5.4.3 Changing Guideline Effective Dates 73

5.4.4 Creating Guidelines 74

5.4.5 Removing Guideline Versions 74

6. Parameters 74

6.1 Creating Parameters 75

6.1.1 Creating a new parameter: 75

6.1.2 Copying an Existing Parameter 76

6.2 Editing Parameters 77

6.3 Viewing Parameters 77

6.4 Deleting Parameters 77

7. Dates & Date Synonyms 78

8. Administration 80

8.1 Deploy to Rule Engine 80

8.1.1 Deployment Details 81

8.1.1.1 Deploy Rules within Status 81

8.1.1.2 Filter Policies By Date 82

8.1.1.3 Export Entities to Server 82

8.1.1.4 Generate Guideline Rules 82

8.1.1.5 Generate Parameters 82

8.1.1.6 Filter Policies By Guideline Type/Templates 82

8.1.1.7 Monitor Deployment Progress 83

8.1.2 Deployment Monitor 83

8.1.3 Status-based Deployment Authorization 83

8.1.4 Deployment Troubleshooting 83

8.2 Export Data 84

8.3 Import Data 85

8.4 Manage Users 87

8.4.1 User Information 88

8.4.2 Roles 89

8.5 Manage Roles 89

8.5.1.1 Role Assignment Restrictions 90

8.5.2 Screen-Based Permissions 90

8.5.3 Entity Permissions 92

8.5.4 Usage Type Permissions 94

8.6 Manage Process 96

8.6.1 Concepts 96

8.6.2 Managing Phases 96

8.6.2.1 Create New Phase 97

8.6.2.2 Edit Phase 97

8.6.2.3 Move Phase 98

8.6.2.4 Delete Phase 98

8.6.2.5 Refresh Phases 98

8.6.3 Managing Requests 98

8.6.3.1 Create New Request Object 98

8.6.3.2 Edit Request Object 99

8.6.3.3 Delete Request Object 100

8.6.3.4 Refresh Phases 100

8.7 Domain Model 101

8.7.1 Classes 101

8.7.2 Attributes 102

8.8 Rules 103

8.8.1 Conditions 103

8.8.2 Logical Operators 104

8.8.3 Actions 105

8.9 Messages 105

9. troubleshooting 106

9.1 Read-Only Mode 106

9.2 Message Validation Errors 106

9.2.1 Missing “%” Symbol 106

9.2.2 Missing “|” Symbol 106

9.2.3 Bad Value 107

9.2.4 Class Not Found 108

9.2.5 Attribute Not Found 108

9.2.6 Invalid Column Number 108

9.3 Deploy Script Issues 108

9.4 Audit Trail 109

9.4.1 Captured Changes 109

9.4.2 View Audit Reports 109

# Introduction

## Overview

The MindBox PowerEditor is a multi-level editing application that allows business users to modify or update automated business policies. Empowering business users with the ability to maintain and develop business rules for an automated system has been an elusive challenge. Many companies have attempted to create rule editors that use a natural language approach. These editors are at first glance appealing; however, they fail to scale and therefore are not practical for commercial applications. MindBox’s PowerEditor has taken the approach of creating an editor that manages rules through an intuitive graphical user interface that is designed to use the paradigm and language with which the business user is familiar. In addition, the MindBox PowerEditor accomplishes this functionality in a user-friendly environment that is scalable and does not sacrifice performance. It also provides a robust feature set for knowledgebase versioning, auditing, and migrating development systems to production.

Through the PowerEditor the following examples of functionality for business users are as follows:

* Specify price adjustment or product eligibility rules based on specific criteria such as credit, product, investor or any other transaction data attribute.
* Specify any number or combination of criteria for a given rule.
* Specify criteria represented in a large variety of data types, including, but not limited to Boolean, numeric, enumerated.
* Copy existing adjustments to make new ones that are similar to existing ones.
* Set activation and expiration dates for rules. This feature provides both a consistent record of criteria offered, as well as the ability to qualify and price a product based on criteria that were in effect during some specific previous time period.
* Create dynamic explanations associated with each rule. The explanations can include transaction-specific text, such as property type, coverage amount, etc.
* Define master / subordinate product relationships.
* Associate multiple products to a rule, and multiple rules to a product attributesHigh-Level Editor Architecture

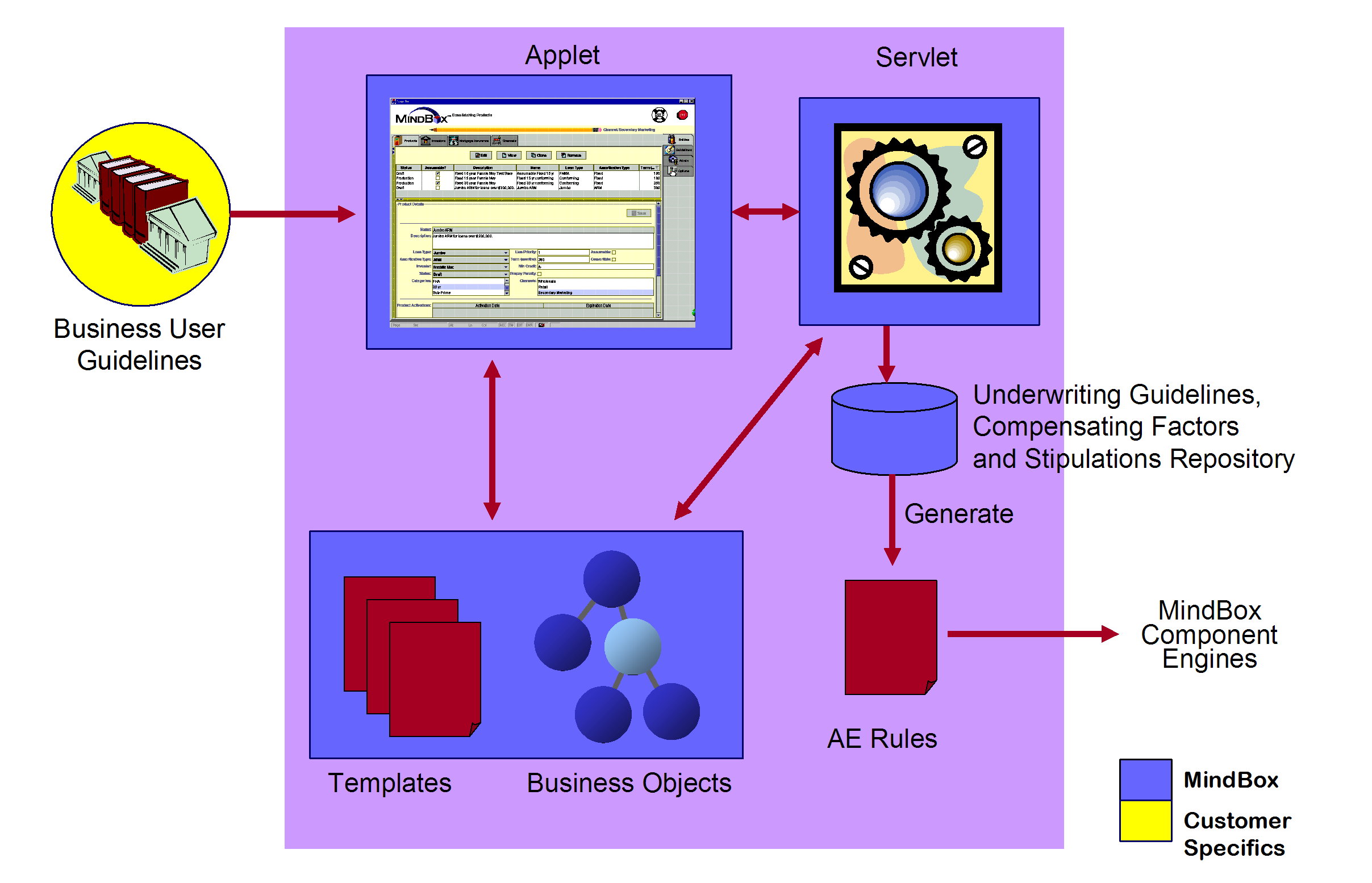


Figure 1: MindBox PowerEditor High-Level Architecture

The following section describes the technical architecture of the MindBox PowerEditor. If the reader is not interested in the technical details, please feel free skip to the next section.

The PowerEditor architecture utilizes a combination of an applet and servlet as shown here. The editor is highly configurable through XML – this includes display properties (e.g. labels, images, menu items, etc.), operational properties (e.g. logging, deployment, session control, etc.) and the knowledge base including the parameter templates. There is an additional Domain Editor software package, which enables the construction of domain objects.

### Major PowerEditor Components

1. Java Servlet (served by the clients’ application server)

This runs on a Servlet Container (e.g., IBM WebSphere, BEA WebLogic, Apache Tomcat, etc.) and provides the security, control logic and the interface support.

1. Customizable User Interface

PowerEditor provides customizable user interface based on technologies like Java Applet and JavaServer Pages (JSP).

1. RDBMS (served by the clients’ database server)

This acts as the repository for the knowledge base as well as administrative functionality such as audit trails.

Knowledge Base Concepts

The MindBox PowerEditor is built on a conceptual knowledge base model that provides the framework for defining many types of business concepts and the linkages between them. This includes concepts such as products, investors, guidelines, rules, etc. The model also consists of a structure for modeling the problem domain. It is recommended that the user become familiar with these concepts prior to delving into the PowerEditor itself. Therefore, the purpose of this section is to provide an overview of these concepts.

## Domain Model

The Domain model provides a definition of the object model used by the rule engine. The model defines the hierarchy of classes and their associated attributes and is typically used to model both input data and derived data. The maintenance of the domain model is achieved via the Domain Editor tool, which is external to the PowerEditor itself.

At the highest level the domain model consists of *classes*. A class is used for containing a set of related attributes. For example, the class “Borrower” would be used to contain information (e.g. name, address, credit score, etc) about a specific borrower.

An *attribute* defines a specific piece of data contained within a class. Examples of attributes could include Loan-Amount, Credit-Score, etc. An attribute is always accessed through the class in which it is a member. For the purpose of illustration the previously mentioned attributes can be represented as LOAN.Loan-Amount and BORROWER.Credit-Score. The term to the right of the period (“.”) indicates the attribute name. The term to the left of the period in all CAPS represents the name of the class in which the attribute belongs. Joined together, they provide a unique identifier for the attribute within the knowledge base.

For more information about the domain model, classes, and attributes, please refer to Section 11, “Advanced Knowledge Base Concepts.”

## Rules

A rule is a means of systematically expressing all or part of a specific business policy or procedure. A rule contains one or more conditions along with an action(s). The set of condition(s) are sometimes referred to as the **Left-Hand-Side** (**LHS**) of the rule while the action(s) is sometimes referred to as the **Right-Hand-Side** (**RHS**). The logic of a rule is such that whenever the conditions on the Left-Hand-Side are true then the actions on the Right-Hand-Side are performed. This can also be thought of as an “IF – THEN” logical statement where “IF” represents the conditions (LHS) and “THEN” refers to the action(s) (RHS) to be performed.

## Messages

A message is a parameterized text explanation used to provide a description to end users of a rule or guideline that has executed. The message will typically include the identity/purpose of the rule or guideline and the corresponding action taken by the rule engine. For example, in the mortgage domain, a price adjustment guideline will describe an adjustment that has been made to the rate or price of the loan; and a qualification guideline will describe why a loan is not eligible for a particular product

Messages can be made more specific to the request through the use of Message Arguments. Message Arguments are implemented as placeholders within a message that can be dynamically populated during the processing of the request.

## Guidelines

Guidelines represent business policies that have been encoded into the knowledge base via the PowerEditor. A guideline consists of one or more rules that share the same action and structure. The action is derived from the purpose of the business policy that the guideline was built from. For example, a common guideline purpose in the mortgage domain is Loan Qualification. The structure refers to the set of attributes used to evaluate the guideline. This structure will also be shared by all of the rules within the guideline. In this case, each rule would check for a different allowable combination of values among the attributes. The presence of an allowable combination of values would indicate that the guideline is satisfied and therefore the associated action is performed. A guideline also has an associated message. The message is used to describe to the end user the action that is performed and why that action was taken.

The figure below depicts a mortgage industry guideline used to define the maximum allowable loan-to-value limits (LTV) for loan applicants. The purpose of the guideline is loan qualification. The guideline is presented in a table view in which each row represents a single rule defined within the guideline. The structure of the guideline is represented by the columns, each one representing an attribute in which the guideline is evaluated.

|  |  |  |  |
| --- | --- | --- | --- |
| ***Occupancy*** | ***Min Loan Amount*** | ***Max Loan Amount*** | ***Max LTV*** |
| Primary | 100 | 400,000 | 95 |
| Primary | 400,001 | 500,000 | 90 |
| Primary | 500,001 | 650,000 | 80 |
| Second Home | 100 | 300,000 | 90 |
| Second Home | 300,001 | 500,000 | 80 |
| Second Home | 500,001 | 650,000 | 75 |
| Investment | 100 | 49,999 | 75 |
| Investment | 50,000 | 400,000 | 80 |
| Investment | 400,001 | 650,000 | 70 |

Table 1: Maximum LTV Guidelines

As is often the case, the business policy employed is more complicated than just establishing a single maximum allowable LTV value. If this were the case the guideline would only contain a single rule. Instead the maximum allowable LTV value varies based on the values of other attributes (e.g. Occupancy, Loan Amount, Credit). Because of this the guideline requires multiple rules to cover all of the possible combinations of conditions that exist within the business policy.

## Guideline Templates

Guideline templates, or simply “templates”, define a data entry grid into which guideline information is entered and the structure of rules generated from the guidelines entered into it. A guideline template must exist in order to enter guideline information in PowerEditor. For example, in order to enter guideline data into a table that looks like Table 1, you would create a “Maximum LTV” template with four template columns, each of which represent a column in Table 1. Guideline templates are maintained in PowerEditor.

## Guideline Actions

Each rule that a guideline template defines has a RHS that performs an action. We shall call the action that RHS of a rule performs guideline action. The details of the action to be performed are defined in the application itself as an ART\*Enterprise function. The action is initiated by executing a call to the function. A guideline action defines this call to the function. Thus, managing guideline actions requires knowledge of MindBox rule engine.

## Parameters

Parameters are similar to global variables that can be viewed and modified by users through the PowerEditor. However, they are considerably more complex than global variables in that the value assigned can be different based on the rule engine input. Examples of parameter uses may include numeric limits (e.g. min, max), calculation terms (e.g. ratios, multipliers), etc. Modifying the value of a parameter allows the user to affect the overall behavior of the rule engine as opposed to modifying guidelines where the effect is much more limited.

The value assigned to the parameter can vary based on the following types of criteria:

* ***By product*** - Parameters are always product specific and must be linked to a product(s) upon creation. Multiple parameters of the same type can exist for different products. For instance, given a parameter type called “Max Loan Amount”, a parameter instance could be defined for each product offered. In each case, the value(s) assigned to the parameter at runtime would be product specific.
* ***By other entities*** - Optionally, a parameter can also be made to assign a specific value(s) based on other entities as well (e.g. channel, investor).
* ***By input data*** - In addition, rules can also be defined to assign a specific value to the parameter based on input data such as loan amount, credit score, etc.
* ***Effective Date*** - There can be multiple activations (versions) of a parameter. As a result, a different value(s) could be assigned to the parameter based on the input date.

## Parameter Templates

PowerEditor employs the use of parameter templates for defining the structure of Parameters in the knowledge base. Parameter Template definitions are managed outside of the PowerEditor.

The symbol is used to represent templates throughout PowerEditor.



## Entities

Entities represent basic central business concepts that are commonly associated with business policies (e.g. guidelines and parameters). The PowerEditor supports this by providing a linkage capability between guidelines and entities. The types of entities available within PowerEditor can be customized for the specific application. Examples of entities in the mortgage domain include concepts such as product, investor and business channel.

## Context

A context can be described as the combination of one or more entities that have been linked to a particular guideline or parameter. A context has the effect of placing additional restrictions on the guideline or parameter. To illustrate this point, let’s say that the context for a specific guideline has been defined to include the Product “FHA-Fixed” and the “Wholesale” business channel. In this example, the guideline would only be evaluated if the product and business channel of the incoming request matched the context of the specific guideline (“FHA-Fixed”, “Wholesale”). In a sense, this can be thought of as a means of filtering guidelines.

## Activation, expiration, and version control

Guidelines, parameters, entities, and messages have Activation dates and Expiration dates. Activation date, also known as the Start date or Effective date, is the date the element becomes effective. Expiration date, also called End date, is the date that element expires.

### Effective Date Range

People typically want to maintain a record of different versions of the guidelines, templates, and parameters they've used. This is done for a variety of reasons e.g. to handle historical requests to the rule engine or to process customers whose loans were approved prior to a policy change. Different versions of guidelines and parameters are distinguished by their effective date range, i.e. the period of time established by their activation and expiration dates.

Deployment Status

Guidelines, parameters, and entities have a Deployment Status that states where in the PowerEditor development cycle they are, e.g. Draft, QA, or Production:

* **Draft**. Data is in a working state, new data or changes should start with this status. Typically, only data with a status of Draft can be edited.
* **QA**. Data is ready to be tested. If problems are found the data is typically reclassified to Draft status so the problems can be resolved.
* **Production**. Production data has been successfully tested and is ready for general use in the production system.

### Customizing Status

As of PE *v.*4.3.8, deployment status can be customized, i.e. you can remove status values you don't need and add new ones. However, Draft and Production status must be maintained. Guideline and parameter status is defined in the MB\_ENUM\_TYPE table in rows where enum\_type is set to ‘system.status.’ Each status has an enum\_id. You may define any number of status values provided there is a Draft status of enum\_id=1 and a Production status of enum\_id=*n*, where *n* is greater than any other status of value defined.

### Status and Deployment

Rules can now be deployed based on their status. PE will deploy all activations in the specified status and "higher" in the range between Draft and Production. For example, if your status range was Draft, Dev Test, QA Alpha, QA Beta, Production and you specified Draft status when deploying you'd get all the rules. If you specified QA Alpha, you'd get rules for activations of status QA Alpha, QA Beta, and Production.

### Pre-Dating Expiration Dates

When working with guideline and parameter activations with a status other than Draft—i.e. when editing dates, creating new activations using the Add button, or copying existing activations—note that the expiration date and time may not be earlier than the current date and time. This restriction can be by-passed two ways.

The easiest is to create a copy of an existing activation, set the activation date for the new copy, and select the option to automatically expire the current activation. You can also change the status to Draft, perform the operation, then change the status back to what it was. Note also that PE does not currently validate activation or expiration dates of parameter activations.

### Status-based Deployment authorization

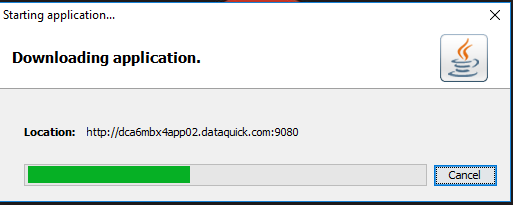
Prior to version 5.8.0, PowerEditor provided a single privilege for deployment. In version 5.8.0 and later, PE supplies a deployment privilege for each deployment status. This allows, for instance, setting up a user’s authorization in such as a way that he/she can deploy policies in Draft status but not in Production.

# Using PowerEditor

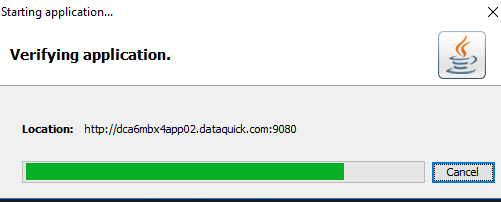
## Starting PowerEditor

To Start PowerEditor:

1. Use the icon provided by PowerEditor Administrator.
2. A temporary dialog will display while the PowerEditor is downloaded (your URL will differ)



and another while it is validated.



It is possible the download and verification are fast enough so that you may not see either of these dialogs.

1. The PowerEditor login screen will appear. Login using your provided User ID and Password to begin using PowerEditor.



Figure 2: The PowerEditor log-in screen

## Screen and Window Management

PowerEditor screens share many common elements.

* Search panes, typically on the left side of the screen, provide tools for finding different things in the knowledge base. The information can often be displayed several ways to facilitate a search, e.g. table or “tree” view.
* Listing panes, often at the top of the screen, to the right of the search pane, display search results, typically in list form. Select one or more of them to review them in the Detail pane, below.
* The Detail pane, below the listing area, displays detailed information about items selected in the listing pane, above. Clicking on the Edit, View, or Copy buttons in this pane often take you to a new screen used for data entry and specifying even greater detail.

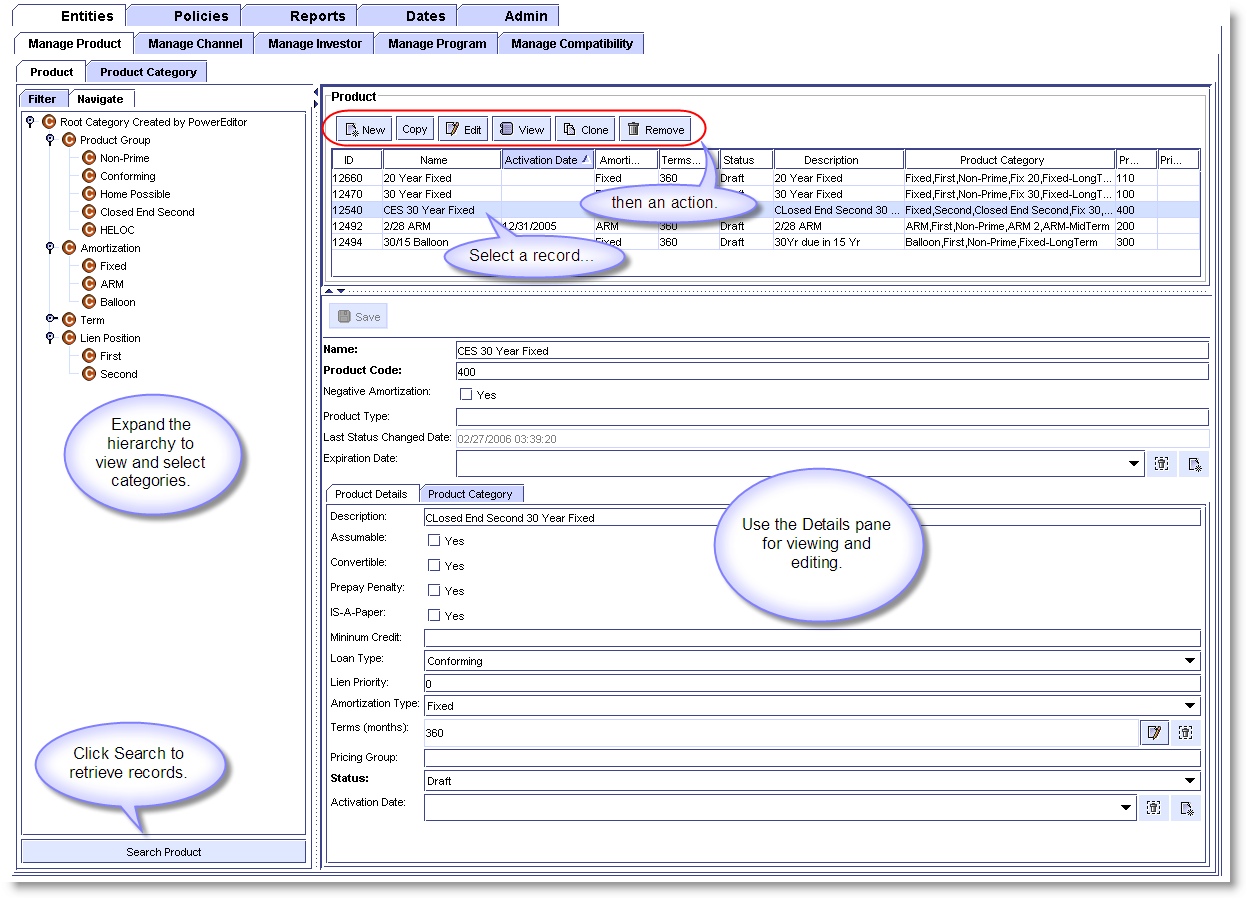


Figure 3: Common screen elements: Search, Listing, and Detail panes.

Drag pane boundaries (marked with a dotted line) to resize them for better viewing. Minimize and maximize panes by double-clicking the small black triangles. Inside a pane use Tab key to move the cursor from field to field. In a table or grid use your keyboard’s Up Arrow and Down Arrow keys to scroll through row data.

## Views

PowerEditor uses several types of views for the purpose of displaying and navigating through lists of records. These include Tree Views, Table Views and Grid Views.

### Tree Views

Hierarchical data structures, where elements share a parent/child or “contains” relationship such as product hierarchies and class/attribute lookups, are usually shown in a “tree” view. Click the controls on the left of the tree (or double click elements) to expand and collapse the branches. For ease of navigation, when they first appear hierarchies are shown collapsed, though in hierarchies with selectable checkboxes if any elements have been selected those branches will be expanded (Figure 4).

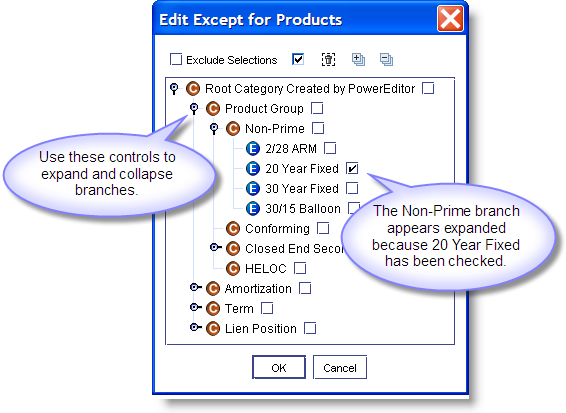


Figure 4: Branches with a selected element are displayed.

### Table Views

Where information is displayed in a table click the column headings to change the column the data is sorted by. To reverse the sort (i.e. from ascending to descending order) click the active column's heading.

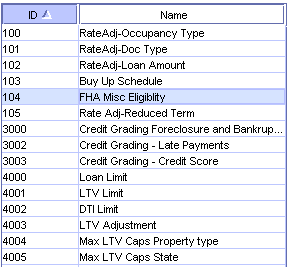


Figure 5: Table View listing of records

Click a row for editing or viewing in the detailed screen to the right.

### Grid Views

PowerEditor guidelines are displayed in a "grid" (guidelines are sometimes called "guideline grids.") Many guidelines can be defined using a single template. Each row within the grid represents a guideline rule. Grids also make it easier to visually check rules for accuracy and overlap.

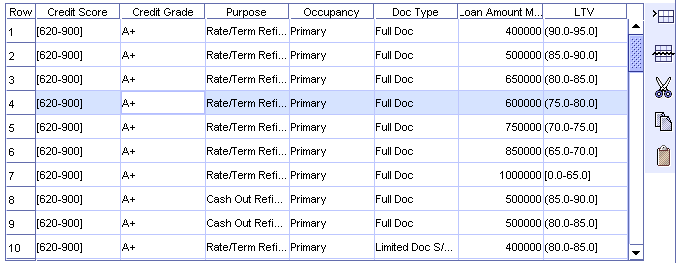


Figure 6: Grid View of guideline rule data

#### Editing Grids

Here are some tips for working with grid data:

* Click on a cell to edit it.
* Click on a row number to select it.
* Double-click a cell to display a list of valid options. If a cell takes data from the keyboard, type in a value.
* Use the Tab key to navigate between neighboring cells.
* You can copy and paste a cell value into one or more other cells.

#### Editing Grid Rows

Click the icons at the right of the grid for these operations:

**Add Row.** Inserts a row above the selected row.



**Append Row**. Appends a new row at the bottom of the grid.



**Delete Row**. Deletes the selected row.



**Cut Row**. Removes the selected row and stores a copy in the buffer.



**Copy**. Copies the selected row and stores a copy in the buffer.



**Paste**. Places the contents of the buffer into the selected row.



#### Copying a Row in the Grid

To copy an existing row:

1. Create an empty row in the grid using the **Add Row** or **Append Row** icons.
2. Select the row you want to copy and click the **Copy** icon.
3. Select the empty row and click the **Paste** icon.

 Note: You can use these same tools to copy and paste data into PowerEditor from an Excel spreadsheet, or to copy grid data from PowerEditor into Excel.

If you try to edit a record that is already being edited by another user, a message box will appear to tell you that the record has been locked by another user. Wait a few minutes, when that person has saved their changes and left the edit screen you can continue with yours.

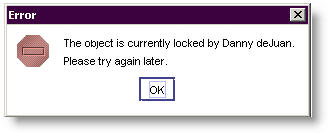


Figure 7: Locked record message.

## Edit Features

Many specialized editing features are provided by PowerEditor to assist the user with data entry and validation. Many of these features are re-used throughout the editor and therefore are discussed in this section for the purpose of familiarizing the user.

### Drop-down Lists

Enumerated data elements, i.e. those with multiple-choice options, are displayed using a drop-down list. (Some drop-down list values are defined in the domain file, others are defined in PowerEditor.)

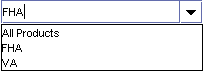


Figure 8: Drop-down list

Click on the downward arrow at the right hand edge of the field to display the list of valid options and choose one.

### Lookup Fields

Use lookup fields to view and select values from the PowerEditor database. Lookup fields are marked with a binoculars icon, and a trash can.



Figure 9: Lookup Field

Click the binoculars to display a list of valid options. Classes and attributes are displayed in a tree view, non-hierarchical data is displayed in a table.

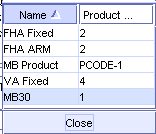


Figure 10: Lookup Field-Table View

Click the trash can to reset the data entry box to empty.

### Date Fields and Date Synonyms

PowerEditor dates appear in the “01/12/2006” format (**mm/dd/yyyy**). Some date fields include hour and minute information e.g., “01/12/2004 12:00” (**mm/dd/yyyy hh:mm**). Use the down arrow at the right of the date field to display a list of available dates or create a new one by clicking the New icon and selecting a date from the calendar.

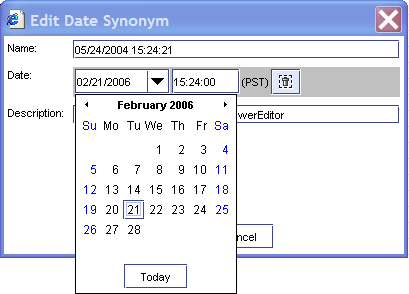


Figure 11: Date Field with calendar.

PowerEditor dates are actually date “synonyms.” They can appear in typical date format or optionally as a descriptive text string, e.g. “New Years Day,” for January 1. Changing a synonym’s date automatically updates it everywhere the synonym is used.Check Boxes

Click to toggle (check and uncheck) the boxes.

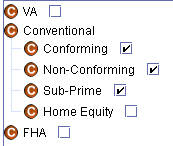


Figure 12: Checkboxes

### Multi-Select Boxes

In a multi-select box you can choose multiple values (unlike the drop-down box, that allows only one). These enumerated values are also defined in the domain file.

Click on a value to select it, to select (or de-select) multiple values use Control+Click. You can use the Exclude Selections box as a shortcut when selecting all but one or two values, when it's checked all items *except* the ones selected apply.

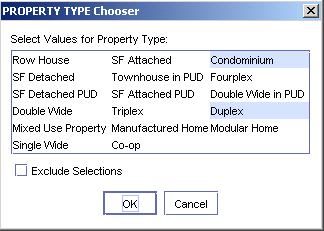


Figure 13: Multiple Select Box for Property TypeEdit Range Box

The Edit Range box is used to define the upper and lower limits of the range check for the data element. The default is the non-inclusive operator: (> for lower limit and < for upper limit).

Check the Inclusive box to use the inclusive operator (>= for lower limit and <= for upper limit).

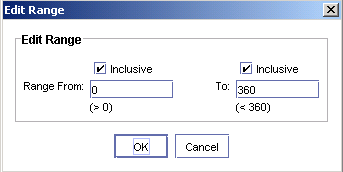


Figure 14: Edit Range Box

Note the use of end brackets and parentheses in the grid cell.

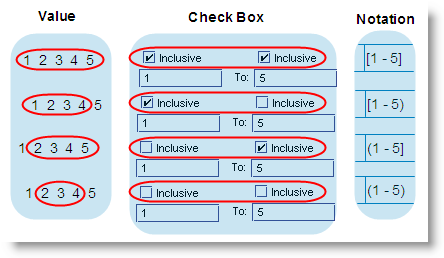


Figure 15: Working with ranges

### Command Icons

Common PowerEditor user interface icons include:

**Validate –** Checks the validity of an item (e.g. a guideline or message).



**Delete -** Deletes selected item.



 **View -** Opens selected for viewing (item cannot be edited).

**Copy –** Places a copy of the active record in the paste buffer.



**Paste -** Places the contents of the paste buffer into the highlighted area.



**Edit -** Opens the selected item for editing.



**Find -** Accesses search/filtering tools for locating specific knowledge base items.



**New -** Creates a new knowledge base item (the new item will be empty).



**Save -** Saves the current state of the knowledge base.



**Cancel -** Exits current screen or dialog box without saving changes.

## Deployment Status

Most knowledge base elements (e.g. guidelines, parameters, and entities) have a Deployment Status. Use the deployment status drop down list to display and modify status.

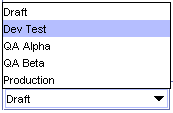


Figure 16: Deployment status drop-down list

The Last Status Change field appears to the right of the deployment status field and is automatically updated whenever the deployment status is modified.



Figure 17: Last Status Change

## Deletion Confirmation

As a protective measure PowerEditor requires confirmation prior to the deletion of any knowledge base item.

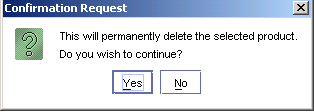


Figure 18: Confirmation Request

Timeout Notification

Users are notified of impending timeouts prior to being timed out, giving them opportunities to save any unsaved work and extend session. Users will be notified five minutes and one minute before being timed out. When the session times out, users will not be able to access any screen.

## Exiting PowerEditor

To exit PowerEditor save your work and close the window it’s running in. At the prompt click **OK** to exit.

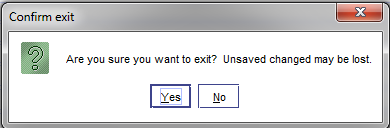


Figure 19: Exit prompt

# Entities

The Entities tab is used to manage entities in the knowledge base. Entities contain information about products and other things you want to associate with guidelines such as programs, channels, and investors. Product entities are the most frequently used but you can have any number of entities, of any type.

Entities are used to associate specific products (or programs, or channels, or investors) with specific guidelines; this connection gives each guideline its context. Context is the set of entities—e.g. a specific group of products or programs—to which a guideline applies.

Entities allow you to manage guidelines based on their product type, category, or combination of categories. For example, you might use the product entity to create guidelines for all conforming fixed rate products, for ARMS, or for specific Jumbo Fixed 30 products. A powerful indexing tool, entities contribute significantly to the speed and agility of PowerEditor.

The fields and columns shown in Figure 20 may differ from what’s displayed on your Manage Products screen, which has been customized for your implementation.

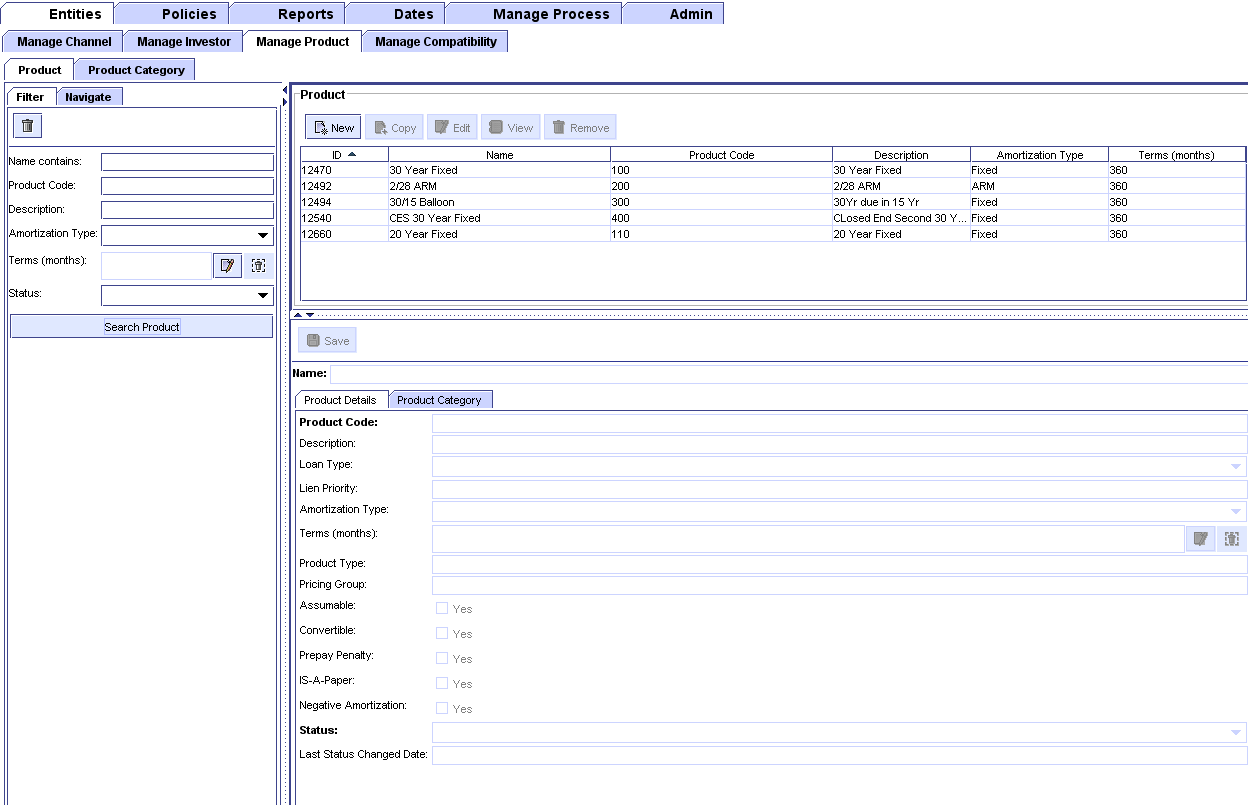


Figure 20: The Manage Products screen

## Entity Search

The Entity search function provides two methods for finding entity records: by Filter, or Navigation.

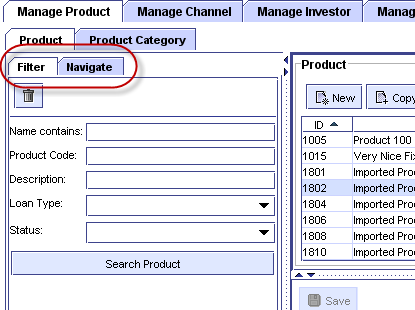


Figure 21: Search Products’ Filter and Navigate tabs

### Filter

The Filter tab allows the user to search for entity records by entity properties. Each property is presented as a search criteria.

Any combination of criteria can be utilized; blank fields will be ignored in the search for entity records. Some of the more common search fields are described below (the fields on your screen may be different).

* *Name Contains -* Checks to see if the criteria value is a subset of the product name. For example, in this case, a product entity named “FHA Fixed” would match because “FHA” is a subset of the product name.
* *Description -* A blank criteria value means that the Product Description field will be ignored when searching for products using this filter.
* *Status is -* This criteria says to search for all products that have a Deployment Status of draft. In this case, the criteria value for deployment status can be selected from the corresponding drop-down box.
* *Loan Type is -* The criteria value “Any” means that the Loan Type will be ignored when searching for products using this filter.
* *Term (in months) is -* No drop-down box is provided for this numeric field.

#### Executing the Search

Click the Search <Type> button to execute the search and display the results on the right side of the screen (Figure 22), where <Type> is the type of the entity you're working on (e.g., Search Product). Again, the column headings in your listings may be different.

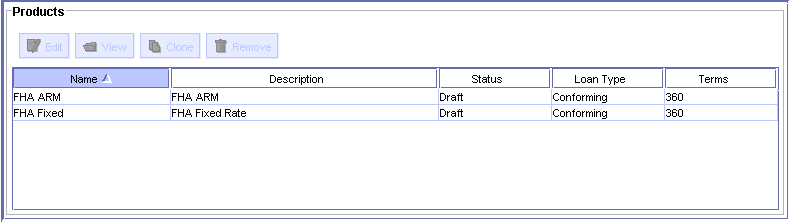


Figure 22: Filter results in listing window

### Navigate

Click the Navigate tab to display the product hierarchy and search for entity records by category.

Use the toggle controls  to expand and collapse categories, you can also double-click categories to open and close them. The results of your search will appear in the list view at the top-right.

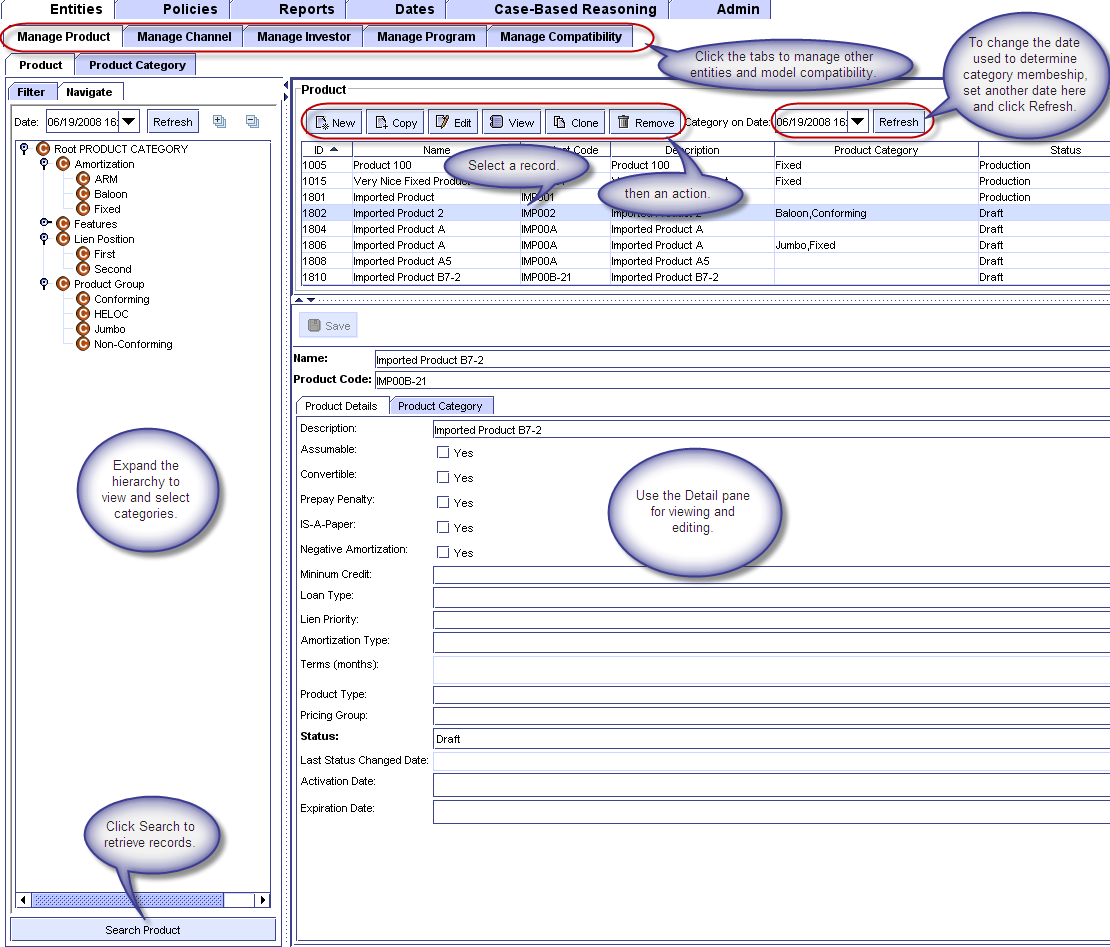
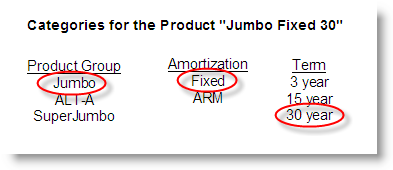


Figure 23: Navigate Entities

PE does not yet have a way to automatically create a category of entities that are not members of another category. For example, this would be useful if you had to manually create a category for a small number of Interest only products, and one for Not Interest Only products. If you’re faced with this task you can use the Navigate tab and Control+Click to select multiple categories and display a list of all the entities in those categories.

## Manage Categories

Entity information can be aggregated into categories. For example, products can be linked to multiple categories, to a single category, or to none at all. Typically products belong to several categories.



Categories are organized hierarchically and can have one or more subcategories.

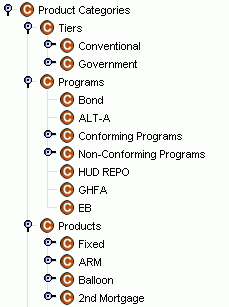


Figure 24: Sample Product Hierarchy

Link a guideline to a category and it’s automatically associated with every product in that category; add a product to a category and it’s associated with every guideline in that category.

Use the Manage Product Categories tab to add, update, and delete categories. For a description of how entities and guidelines are associated, please see “Associating Guidelines and Entities,” on page 62.

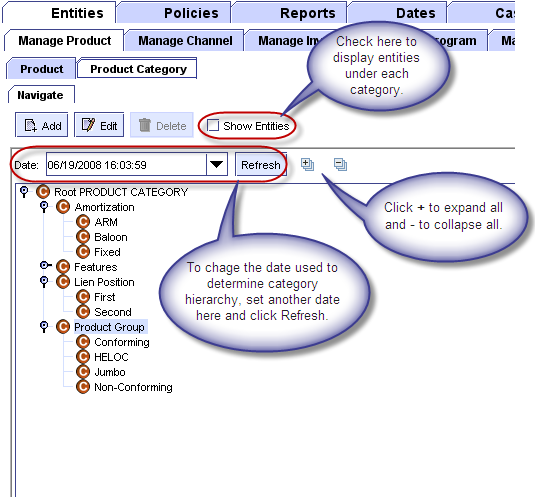


Figure 25: Manage Product Categories

 Note: For some installations editing Product Categories might impact your rule engine, check with your engine administrator prior to editing Product Categories.

### Add, Edit, and Delete Product Categories

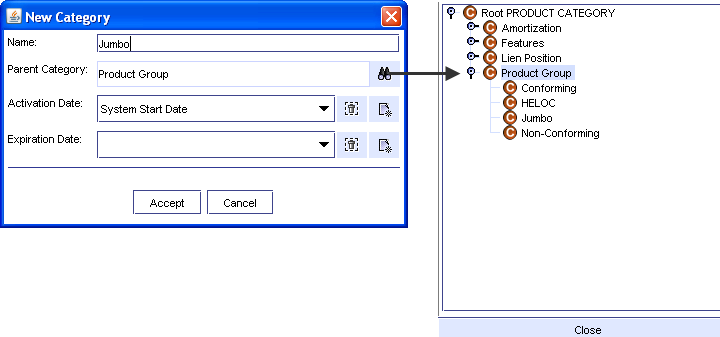


Figure 26: Create Category

* To add a category click the Add button. Enter a descriptive name for the new category. If you want to change its parent, click the binoculars to display the product category hierarchy and select one. Be sure to specify Activation Date for the new category.
* To change a category’s name or parent, select it and click the Edit button.
* To delete a category select it and click the Delete button. If the deleted entity was part of a guideline context, it is removed from the context, but the rest of the guideline remains in tact.

## Managing Entities

Select the entity you want from the list and click on the appropriate button: Edit, View, Copy, Remove. Click New to create a new entity.

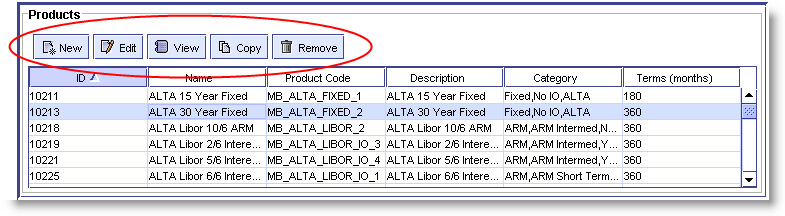


Figure 27: Listing Window for Products

You may select the entity you wish to manage from this screen by single clicking on it. The selected item will then appear highlighted. Entities can then be managed using the tool bar buttons (e.g. edit, view, clone, remove) at the top of the window.

### Entity Details

Make your additions and changes in the Entity Detail pane. Note that the fields and controls in this pane change depending on the type of entity you’re working on (e.g. Product, Investor, Channel, etc.) Figure 28 shows the details for a mortgage product.

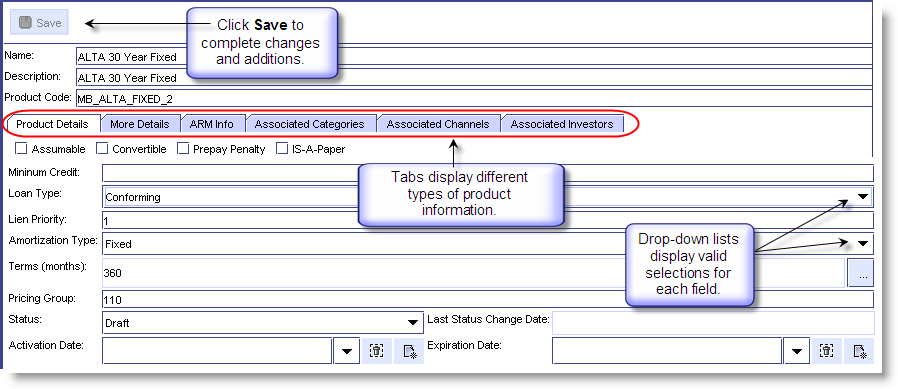


Figure 28: Entity Detail Window

There are some fields that can be common (but not required) to different types of entities. These have more to do with the identification and the migration process than the entities themselves. These are all configurable and will vary from installation to installation.

* *Name -* This is a short name for the entity used primarily or identification and display purposes.
* *Description -* This is a longer description of the entity used primarily or explanation purposes.
* *Code -* Some entities may have a code associated with them (e.g. product-id). This is typically used to link the entity record to an external source of data.
* *Status -* This field contains the deployment status of the entity.
* *Last Status Change -* This field contains the date/time of the last status change. This field is not editable and will be automatically updated whenever a change to the entity status is saved. Entity status changes are typically associated with the migration of data.
* *Activation Date -* The date/time that the entity record becomes effective.
* *Expiration Date -* The date/time in which the entity record expires.

### Editing an Entity

To edit an entity:

1. Select it and click the Edit button, or double click it in the listing window. The Entity Detail window will be automatically populated with an editable version of the record details.
2. Update the information as needed. Some properties allow multiple values to be selected. Use the “…” button to modify values for such properties.
3. Click the **Save** when you are satisfied with your changes.

### Viewing an Entity

To view an entity:

Click on the **View** button to display a read-only version of the entity details. The gray text indicates that the record is not editable.

### Creating a new Entity

There are two methods of creating a new entity. The first is to create a new entity and populate all of the required details, the second method is to copy an existing entity.

**To create a new entity:**

1. Click on the New button. The entity detail window will display a new, empty record.
2. Enter data for the new entity.
3. Click **Save** when you are satisfied with your changes.

Please note that when you copy an entity, guidelines that have the original entity as part of their context do not retain the copied entity as part of their context. But when you clone an entity, guidelines with the original entity as part of their context also have the cloned entity as part of their context. (The Clone option is not available for all entity types.)

**To copy an entity:**

1. Select the entity from the listing and click the Copy button. The entity detail window will be populated with a copy of the selected entity record.
2. The string “ – Copy” will be appended to the name. The activation date will be set to the current date and time.
3. Modify the copied data as needed for the new entity.
4. Click the Save when you are satisfied with your changes.

**To clone an entity:**

1. Select the entity from the listing and click the Clone button. The entity detail window will be populated with a copy of the selected entity record.
2. The string “ – Copy” will be appended to the name. The activation date will be set to the current date and time.
3. Modify the copied data as needed for the new entity.
4. Click the Save when you are satisfied with your changes.

### Deleting an Entity

To delete an entity:

Click the Remove button and click Yes at the confirmation prompt.

## Managing Compatibility

The use the Manage Compatibility tab to model compatibility between entities, e.g., “Product X can be offered over channel Y.” The MindBox rule engines use this information when creating packaged deals containing entity combinations. Compatibility relationships are established for a specific time period.

Use the Entity Type pull-down lists to set or modify the rows and columns of your compatibility matrix (in this case, channels down the left hand side and products and their attributes across the top). Click the Search bar to populate the matrix.

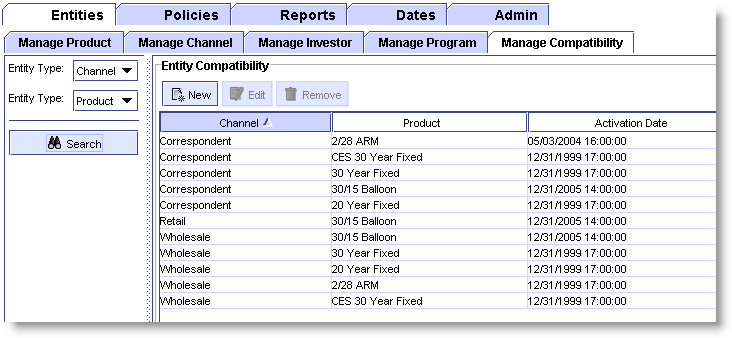


Figure 29: The Manage Compatibility Screen

### Creating a New Compatibility Relationship

To establish a new compatibility relationship click the New button to bring up the New Compatibility dialog box (Figure 30). Select the entities you want in the drop-down boxes and set the activation date and expiration dates to the desired values. Click the OK button to establish the relationship.

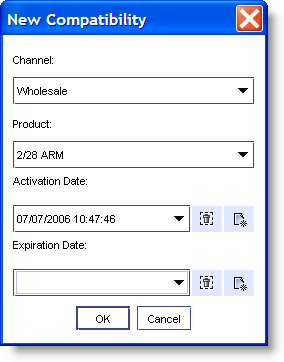


Figure 30: Creating a new Compatibility relationship

### Editing or Deleting Existing Compatibility Relationships

To edit an existing compatibility relationship select it from the matrix and click the Edit or **Remove** buttons.

# Guidelines

The Guidelines Module allows users to manage guidelines in the knowledge base. Guidelines represent business policies that have been encoded into the knowledge base via the PowerEditor. The Guidelines Screen contains three menu items: Manage Templates, Manage Guidelines and Search Guidelines.

## Manage Templates

Use the Manage Template screen to maintain templates (Figure 31). Template folders and their templates appear in the Guideline Template Selection pane.

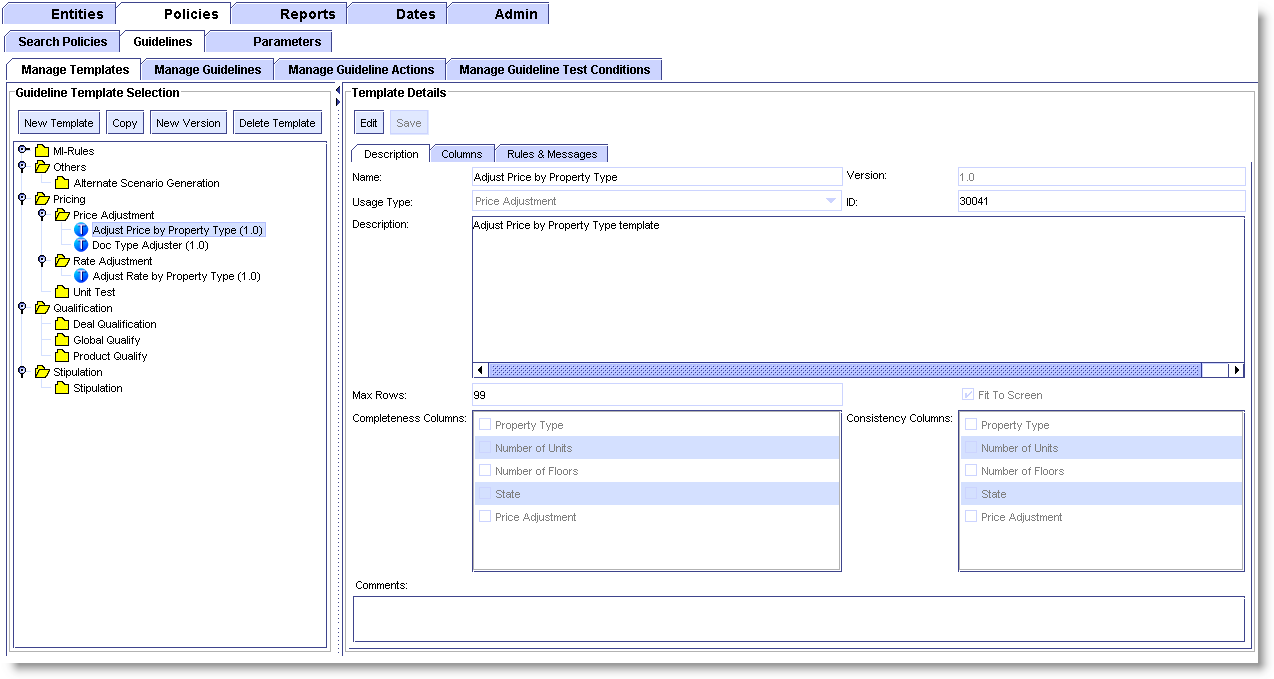


Figure 31: Manage Templates Screen

### Create a New Template

Launch the New Template wizard to create a new template. Use the Next and Back controls to change screens in the wizard, click Cancel at any time to exit.

1. Select a guideline type for your new template from the tree view and click New.
2. In the wizard’s first screen enter the template name and select a Usage Type. Use the drop-down box to display a list of valid Usage Types. Click Next**.**
3. Select attributes for the new template. Use the New button to display and select at least one domain class attribute; use the Delete button to remove attributes from your list. Click Next.
4. Select the guideline action for rules generated from guidelines created from this template. Use the drop-down box to display a list of valid actions. Click Finish.

PowerEditor will create the template, generate the template details, including the rule, and display it in the Template Details pane on the right. It creates a template column for each attribute you selected (Step 2) and each parameter of the guideline action selected (Step 3).

1. Review the new template and click the **Save** button to complete the process (Figure 32).

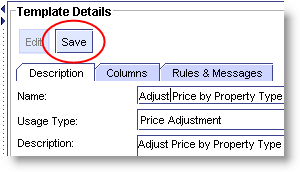


Figure 32: Click Save to finalize the new template

### Copying an Existing Template

Use the Copy button to copy an existing template. This creates a copy of the template, but not its associated guidelines.

1. Select the template you want to copy and click Copy.
2. Make any required changes. (See Section 5.1.4, “Editing Existing Templates,” below.)
3. Click the **Save** button.

### Create a New Version of a Template

The **New Version** button makes a copy of the template you select and gives it a new version number.

 Note: Copy vs. New Version

The Copy button creates a copy of a template without its associated guidelines. Use Copy when you want to build a different template with a similar structure, e.g. to use an existing LTV template to create a similarly structured CLTV template.

Use New Version when the template you want to copy has guidelines in production. (This feature is useful when you want to be able to look back, at some future date, and see what guidelines were in place.)

If you have a template with six guidelines, where two are in production and four are not, **New Version** will leave the two in production with the old template and carry the remaining four (that aren’t in production) to the new version of the template. The new version will have the same name and a new version number.

In fact, when users choose to create a new version of a guideline template, they are presented with a list of non-production activations for the template that would cut over to the new version. Users may select the ones they want to carry over to the new version of the template. Only the selected activations are carried over when creating a new version of a template. If no selection is made, all activations will be carried over.

To create a new version of a template:

1. Select a template from the hierarchy and click the New Version button.
2. Enter a version number for the guideline.
3. Select a cutover date, i.e., the day and time the new version of the template should take effect. For example, if you make the cutover date today’s date guidelines associated with the old template will remain in effect until the following day, at which point the new template and its guidelines will take effect. Click the trashcan to clear the field, click the New icon to display the Edit Date Synonym window. (More on date synonyms in Section 9, below.)
4. Review the cutover guidelines you’ve specified and click Finish.

### Editing Existing Templates

Select the template you want to edit from the tree view in the Guideline Template Selection pane and click the Editbutton. A detailed description of the editable fields on the Template Details pane appears below.

#### Description Tab

The Template Detail Description tab (Figure 33).

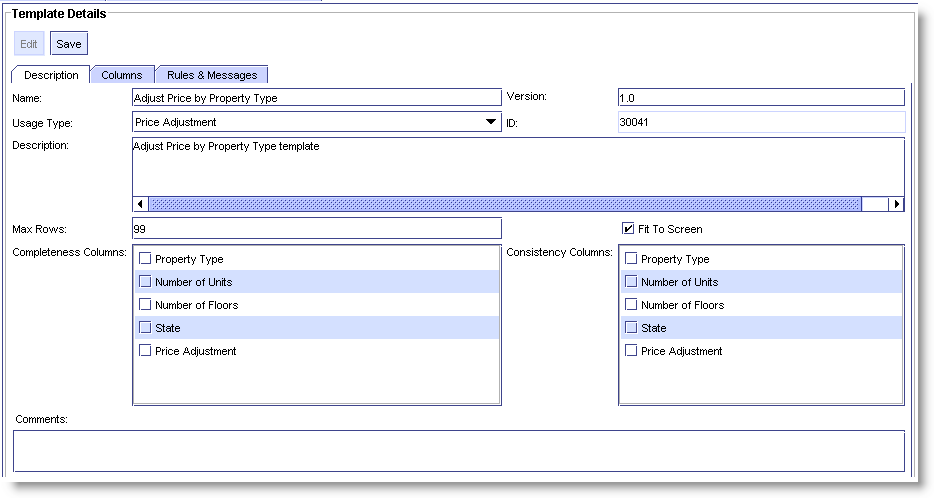


Figure 33: Template Details - Description Tab

Table 2 summarizes fields on this screen; fields can be changed at any time (unless noted otherwise).

| Field | Description |
| --- | --- |
| Name | Template name. |
| Version | Template version, entered by template author. |
| ID | Template ID, automatically assigned by PowerEditor. |
| Usage Type | Template guideline type. |
| Description | Template description; field data appears on the Guideline Detail screen (See Section 6.3.3.1) |
| Max Rows | The number of rows allowed for guidelines |
| Fit To Screen | When checked, guideline grids are expanded to fit the screen width. |
| Completeness Columns | Selected columns in this field are checked for completeness when validating guidelines. |
| Consistency Columns | Selected columns in this field are checked for consistency when validating guidelines. |
| Comments | Place to enter comments, including change logs. |

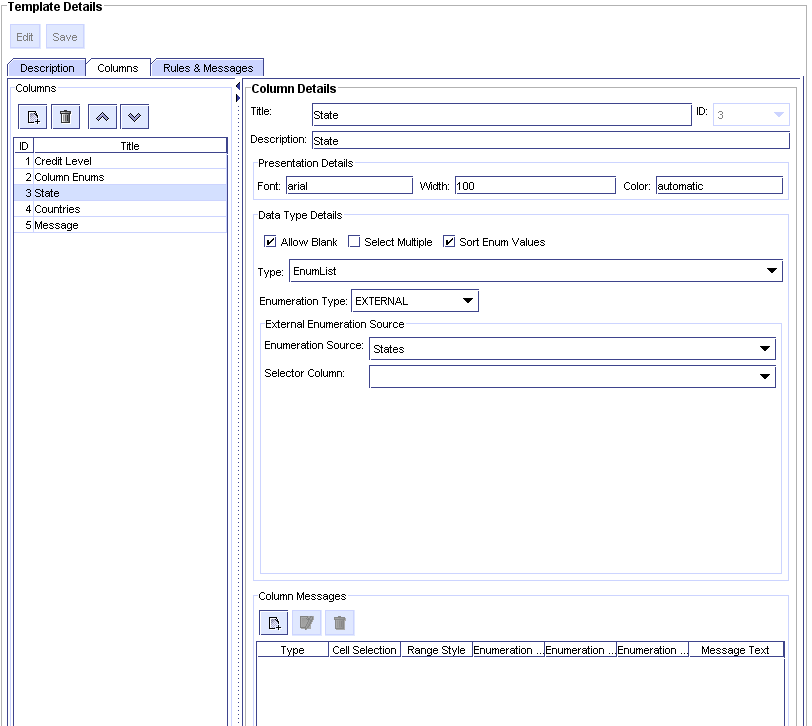
Table 2: Fields on the Template Details screen.

#### Completeness and Consistency

When you run a validate (either for this activation, or all activations), a row by row comparison is performed (each row with every other row). Each of the columns marked for consistency is tested in the comparison, and if any two rows show an overlap, it is flagged. An overlap means there is at least one value for each marked column that is in common between the rows under test. For a numeric range, some part of the range is the same (although I think this is broken - it incorrectly reports overlap when two ranges are completely disjoint. Not sure if there is a tracker item in there already). For an enum, at least one value is in common. For a scalar value, it would be identical. The idea is that the conditions for the rows showing overlap could be simultaneously met by one set of object data. Thus, two rules might fire. This isn't always bad, but sometimes can be - price adjustments are likely times when it would be bad. Another way of putting this, is that you are testing for uniqueness of the rows (at least for the columns so checked off).  
  
Completeness means a value is required for the column so marked.

#### Columns Details Tab

Select a column to display its details in the Details pane.



Template Details – Column Details tab.

**Managing Columns**

Use the New button in the Column Selection pane to create a new column. The Trash button deletes a selected column. Use the down and up arrows to change the order columns are presented in the Guideline Detail screen.**Editing Column Details**

To edit the details of a column select it from the list to display its fields on the right. Table 3 describes the fields on this screen. Note that not all fields are displayed. The exact set of fields displayed differ by Type. Fields can be changed any time, unless noted otherwise.

| Field | Description |
| --- | --- |
| ID | Column ID (column position). Do not edit this field. To change column ordering use the **Up** and **Down** arrow buttons. |
| Title | Column title (the column heading on guideline grids). |
| Description | Column description (does not appear in guideline grids). |
| Font | Column font, defaults to Arial. |
| Width | Column width (relative), defaults to 100. |
| Color | Column color. |
| Type | Column data type. Either EnumList, EntityType, or matches the attribute type specified in the Attribute Map field.  Specify EnumList when the selected attribute contains enumeration values.”  Specify EntityList when you want to create an exception, i.e., a “one-off” rule within a guideline based on context. When the type is EntityList users editing grid data can select from a list of entities and/or categories (see Figure 34, below). |
| Allow Blank | If checked, column accepts empty values. Available for all types. |
| Min | Used to specify the minimum value allowed for this column (numeric and numeric range types only). Leave blank for no lower limit. |
| Max | Used to specify maximum value allowed for this column (numeric and numeric range types only). Leave blank for no upper limit. |
| Precision | Used to specify the decimal precision allowed for this column (float and float range types only). |
| Show LHS Attributes | If checked, makes attribute references in the RHS of the rule of this template or column available for selection for this column in guideline grids (DynamicString type only). |
| Select Multiple | If checked, allows the user to select more than one value. (EnumList type only). |
| Sort Enum Values | If checked, enumeration values display in the order of their display value, unchecked, they appear in the order listed. (EnumList type only.) |
| Enumeration Type | Indicates the type of enumeration values. (EnumList type only.) Valid values are DOMAIN\_ATTRIBUTE, COLUMN, and EXTERNAL.   * DOMAIN\_ATTRIBUTE: Uses enumeration values declared for a domain attribute. Used in conjunction with the Attribute field. * COLUMN: Uses enumeration values defined in the Enumeration Values field. * EXTERNAL: Uses one of pre-defined external enumeration sources. Used in conjunction with the Enumeration Source and the Selector Column fields. |
| Attribute | Class and attribute that the column is mapped to. For DOMAIN\_ATTRIBUTE enumeration type only. |
| Enumeration Values | Defines a list of enumeration values from which the user can select. For COLUMN enumeration type only. |
| Enumeration Source | Selects one of pre-defined external enumeration sources. For EXTERNAL enumeration type only. |
| Selector Column | Specifies a selector column used to narrow available enumeration values for this column. For EXTERNAL enumeration type only. |
| Column Messages | Column message values, used as conditional message fragments. See “Managing Column Messages,” below, for more on conditional message fragments. |

Table 3: Column fields.

#### Entity List Columns

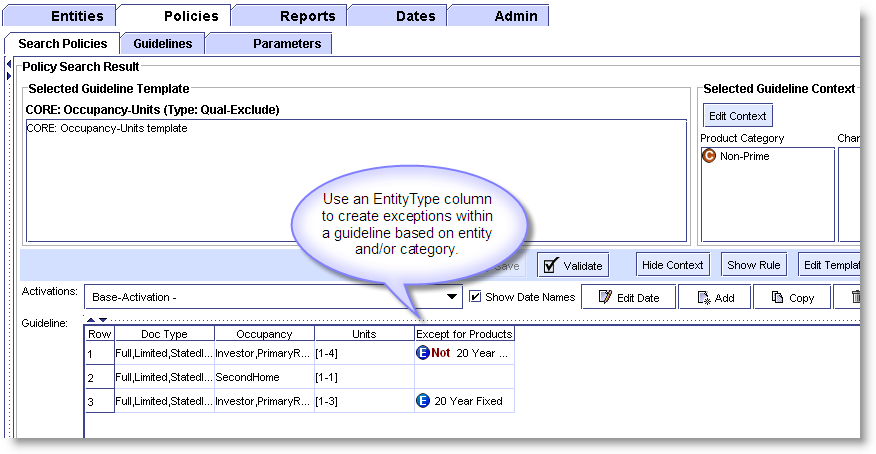


Figure 34: The EntityList column type.

The EntityList column type offers grid editors a dialog box from which to make entity and category selections; the options they’re offered depend on how the field has been defined, e.g. multiple select, entity only, entity, category, etc. (Figure 35). Entity and category-related data can appear in the same cell.

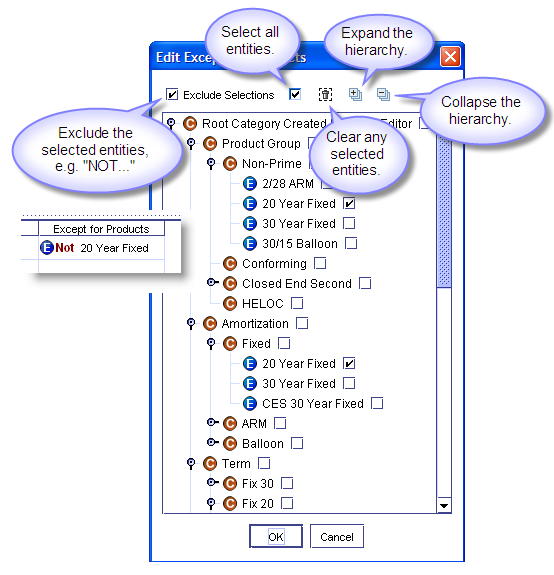


Figure 35: An EntityList selection box.

#### Managing Column Messages

The fields below are used to construct column messages (also called “message fragments”). Every column message includes:

* Type
* Cell Selection
* Range Style
* Enumeration Delimiter
* Enumeration Final Delimiter
* Enumeration Prefix
* Message Text

Instructions for authoring messages appear in Section 5.1.7. Unless otherwise specified, the message components described in Table 4 are optional, and default to values specified in the configuration XML.

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Use one of the following:** | | |
| Type | range: | applies to any of the Range DataSpecs. The attribute rangeStyle is required. | |
| enum: | applies to EnumList DataSpec. The attribute cellSelection is required, the attributes enumDelimiter, finalEnumDelimiter, enumPrefix are optional. | |
| any: | applies to DataSpecs of any type. The only attribute allowed is prefix, which is optional. | |
| rangeStyle | For: type="range", Specifies the style of the message that is generated. This must be either “verbose,” “symbolic,” or “bracketed,” for example: | | |
| verbose: | “greater than or equal to zero and less than 100” | |
| symbolic: | “>=0, <100” | |
| bracketed: | “[0, 100)” | |
| cellSelection  **Note:** Must be used with multipleSelect EnumList columns only, or message will not generate, e.g. the column message might be different if the cell contains single vs. multiple enumerated items, or if the “Exclude Selection” option is checked from an enumerated list. | For: type="enum", specifies phrases specific to options selected for a given cell. Possible values for this attribute are: | | |
| default: | | Specifies default phrase and/or options, overridden by those below: |
| enumExcludeSingle: | | Single enumerated value selected, “Exclude Selection” selected. |
| enumIncludeSingle: | | Single enumerated value selected, “Exclude Selection” not selected. |
| enumExcludeMultiple: | | Multiple enumerated values selected, “Exclude Selection” selected. |
| enumIncludeMultiple: | | Multiple enumerated values selected, “Exclude Selection” not selected. |
| enumDelimiter | For: type="enum" with multipleSelect enumerated values, this specifies the phrase that is used between multiple selected items. | | |
| enumFinalDelimiter | For: type="enum" with multipleSelect enumerated values, this specifies the phrase that is used between next-to-last and the last of multiple selected items. | | |
| enumPrefix | For: type="enum," the text will be pre-pended to the cell value. | | |

Table 4: Column Message components.

#### Rules & Messages Tab

Figure 36 shows the Rules & Messages tab populated with rules from the selected template.

Detailed instructions for authoring rules are provided in Section 5.1.6.

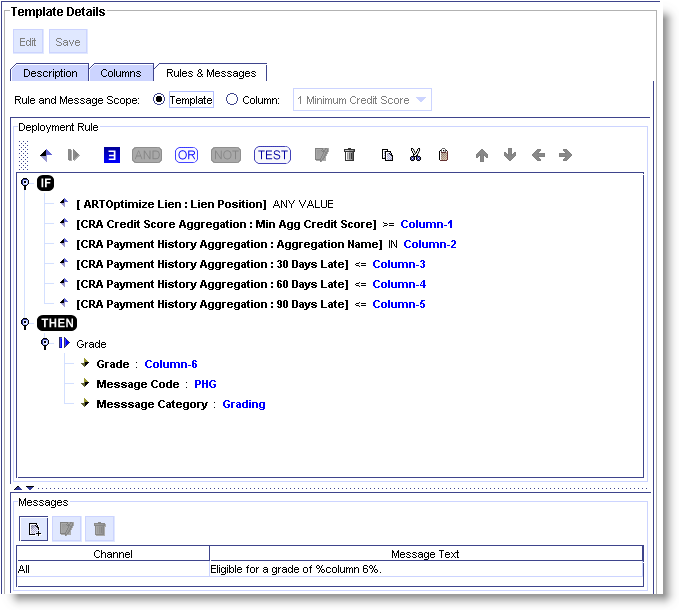


Figure 36: Template Details - Rules & Messages Tab

### Removing Existing Templates

Select the template you want to delete and click the **Delete** button.

### Authoring Template Rules

Authoring template rules require a fairly technical understanding of the following concepts: domain model, classes, attributes, logical operators and rule structure. If you intend to author template rules, be sure to first understand the concepts presented in Section 11, “Advanced Knowledge Base Concepts.”

Template rules are authored in the Template Details Rules and Messages tab, depicted in Figure 36. It is used to edit the conditions and action of the rule. The conditions, or “Left Hand Side (LHS)” of the rule are represented as components on the **If** side of the rule. (In addition, this includes use of the logical operators AND,” “OR,” and “NOT,” which control how conditions are evaluated. The actions, or “Right Hand Side (RHS)” of the rule are represented as components on the **Then** side of the rule.

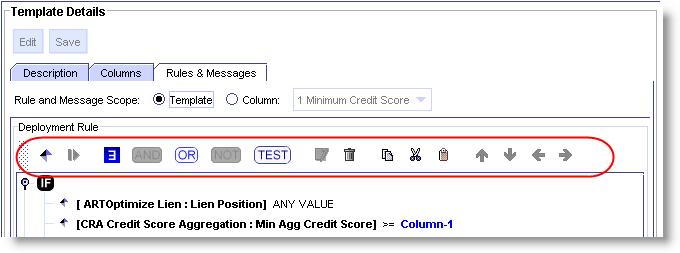


Figure 37: Deployment Rule Authoring Screen

The toolbar at the top of the screen provides a set of tools for the user to manipulate and navigate the rule definition. They include:

 Add a new condition

 Add an Action

Add a new Exist expression



 Create an ‘AND” clause between conditions

 Create an ‘OR’ clause between conditions

 Negates or Inverts a condition.

 Edit the selected element

 Delete the selected element

 Cut the selected element

 Paste the selected element

 Move selected element up

 Move selected element down

 Move selected element left

 Move selected element right

#### Creating a Condition

To create a condition:

1. Select where in the Left-Hand-Side (LHS) of the rule you would like to place the condition. A condition is inserted at the top-level by selecting the  symbol. Otherwise, you must insert the condition under one of the logical operators **(AND**, **OR, NOT)**. Only items preceded by the  symbol can be selected for inserting conditions.

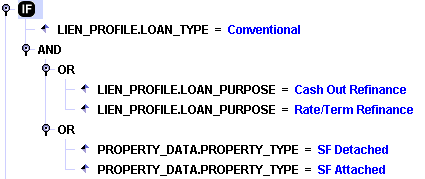


Figure 38: Rule definition with IF node selected

1. Once the location has been selected, click on the  button. If the button is grayed in color this means that you have not selected a valid location in the rule LHS to insert a condition. Otherwise, the “New Condition” window will be displayed.

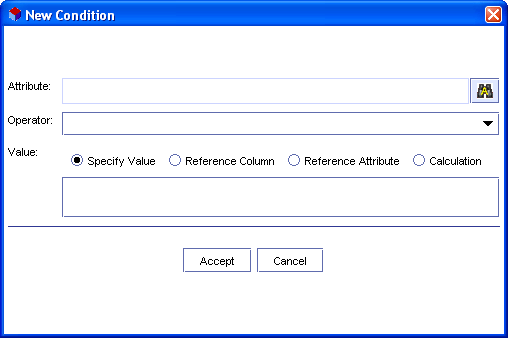


Figure 39: New Condition Dialog Box

1. Click the binoculars icon to display a tree view of the domain class definitions and expand the class names to view attributes. Select the attribute you want your condition to operate on and click the OK button.

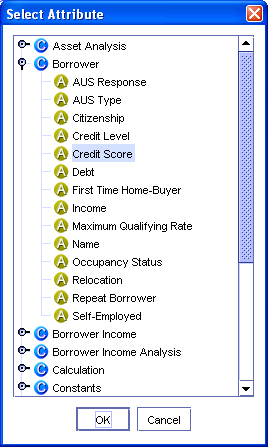


Figure 40: Select Attribute Dialog Box

1. Select an operator from the drop-down list. The list of allowable operators displayed will vary based on the data type of the attribute.
2. Provide a value in the “Value” field. A value can be a specific value, column reference, attribute reference, or a simple calculation. Table 5 summaries their differences:.

| Value Type | Description |
| --- | --- |
| Column Reference | Reference a value in the column. When PowerEditor evaluates this condition, it will use the value from the specified column. To specify a column reference, select the Reference Column radio button and choose a column. |
| Attribute Reference | Reference an attribute of a domain class. When PowerEditor evaluates this conditions, it will use the value of the specified attribute. To specify an attribute reference, select the Reference Attribute radio button and choose an attribute. |
| Calculation | Produces a simple calculation using a column and an attribute. When evaluating this condition, PowerEditor will use the result of performing the specified calculation. To specify a simple calculation, select the Calculation radio button, and select a column, an attribute, and choose a mathematical operator. |
| Specific Value | Enter a specific value. This will also vary based on the attribute’s data type. For example, enumerated fields will have a drop-down list of values while numeric fields will require manual input by the user. |

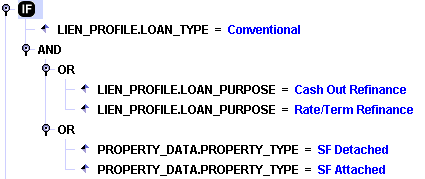
Table 5: Differences between Value fields.

1. When completed click on the Accept button. The new condition will appear on the Rule Authoring screen.

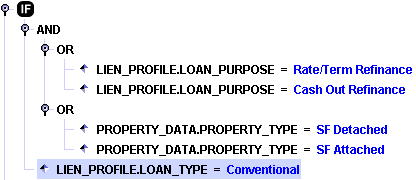
You can also copy an existing condition within the rule and paste it into another location on the LHS. Then the copied condition can be modified as needed.

#### Moving Rule Elements

The arrow buttons (, , , ) on the Rule Authoring screen toolbar allow the user to move existing rule elements within the LHS of the rule. The up and down arrows allow a selected rule element to be moved vertically within the rule’s LHS. Vertical movement can only occur within the same operating level. For example, in the figure below, the condition  represents one level of the LHS.



Selecting the aforementioned condition and clicking the  button would result in the following LHS.



#### Using Logical Operators

Logical operators (e.g. AND, OR, NOT) can be used to apply special handling to a group of one or more conditions on the LHS of the rule. As a result, all the conditions defined under the logical operator are effectively treated as a single condition. The logical operators and their method of special handling are described below:

**AND**. All conditions defined under the AND logical operator must be true in order for the rule to be satisfied.



To utilize, select the location for the operator and click the  button.

**OR**. At least one of the conditions under the ORlogical operator must be true in order for the rule to be satisfied.



To utilize, select the location for the operator and click the  button.

**NOT**. This logical operator negates the condition(s) it is applied to. For example, the NOT logical operator applied to the conditions below result in the rule evaluating that both conditions are not true.



To utilize, select the location for the operator and click the  button.

#### Using Exist Expressions

The primary purpose of using Exist expressions is to enforce lineage among parent-child classes, when more than one object of the same domain class will live in the rule engine. Since use of Exist expressions requires knowledge of inter-workings of the rule engine, it is recommended to consult with PowerEditor administrator before using Exist expressions.

The Exist operator, denoted by the button, is used to create Exist expressions.



An Exist expression consists of the following:

* A domain class
* A set of rule elements.

The domain class set for an Exist expression is said to be the parent class of child conditions of the Exist expression. That is, an Exist expression is satisfied, for rule firing, if the set of rule elements are evaluated to be true, and every class that all conditions, that are direct child of the Exist node, is a child of the domain class of the Exist expression.

For example, in Figure 41, in the rule engine, the Exist expression is satisfied if and if only Property Data’s Is Subject Property attribute is set to true and Property Address’s State Code is one of the values specified in column 1 is true AND the Property Data is a child of Property Container and that Property Address is a child of Property Container.



Figure 41: Sample Exist Expression

Exist expressions can contain logical operators and other Exist expressions, as well as conditions.

**Adding a New Exist Expression**

1. To add a new Exist expression:
2. Select a rule element that can contain other rule elements, such as the IF node. Then, click the Exist button. The Exist button will be disabled if you have selected a rule element that cannot contain an Exist expression.
3. Next, a dialog prompting for a domain class appears. Note that not all domain classes are available for selection. The dialog only lists the domain classes that have at least one child class.
4. Click the OK button to accept the selected domain class.
5. Then, while keeping the Exist expression selected in the Rule Authoring screen, add conditions or other rule elements, as usual.

#### Removing a Condition

To remove a condition, select the condition and click on the  button.

#### Selecting a Guideline Action

The actions performed by rules are selected from a list of available guideline actions within the system. New actions must be defined in the Manage Guideline Action screen. A guideline action has the following attributes:

* Display Name
* A list of argument types
* Allowed usage types

Perform the following steps to select an action for a rule:

1. Select the symbol so that it becomes highlighted.
2. Click on the Add New Action () button. This will result in the “Add New Action Window” being displayed.

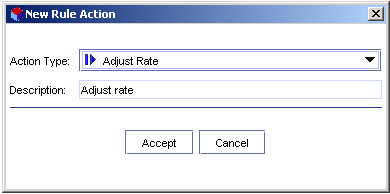


Figure 42: New Rule Action Dialog Box

1. Select a guideline action from the drop-down list and then click on the “Accept” button. The new action will appear in the Rule Authoring screen.



1. Provide values for any action parameters. Action parameters will appear directly below the Action Name in the Rule Authoring screen. In the figure above, the action has a single parameter called “Adjustment Amount”. To provide a value, double click on the parameter name. This will result in the “Edit Action Parameter” window being displayed.

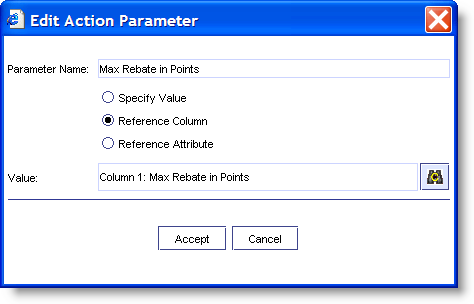


Figure 43: Edit Action Parameter Dialog Box

1. Enter a value for the action parameter and click on the “Accept” button. The value for an action parameter can be a specific value, a column reference, or an attribute reference. To specify a column reference, select the Reference Column radio button and choose a column.

### Authoring Messages

Messages are composed in the Message Edit dialog (Figure 44). Messages are constructed of plain text, message arguments, and conditional delimiters. Message arguments are placeholders for data that will be entered by PowerEditor, they tell the rule engine where to retrieve the value for this message at runtime. Message arguments can be inserted anywhere in the message text.

PowerEditor messages consist of:

* Channel – the channel for which this message is to be used, defaults to “Any.”
* Conditional Delimiter (optional) – Text inserted between delimiting conditional message segments, other than the last segment. A condition message segment is a column message.
* Condition Final Delimiter (optional) –Text inserted between the last and the next-to-last last conditional message segment. A condition message segment is a column message.
* Message Text – the message text. See Section 2.3 for more information on messages and the format of message text.

To create a new message for the template:

1. Click the New Message button in the Message pane, this displays the New Message dialog.
2. Fill in the fields, and click the Accept button. Select “Any” for the channel field if you wish the message to be used for all channels; the Conditional Delimiter and Conditional Final Delimiter fields can be left empty.
3. New message will appear in the messages sub-panel.

* Use the attribute lookup (“A” binoculars) to paste in attribute references for the message. It displays a list of classes, click on a class name to display and select the attribute name you want inserted into the message text. Click the OK button to insert the argument into the message text, (e.g. “DTI of |deal\_calc\_a.dti|”).
* Use the column lookup (“C” binoculars) to paste in a column reference for the rule.
* Use the Insert Column Message button to insert a column message.
* Use the Validate button to run a check on the message to verify that it is constructed properly. A list of message-related error messages appears in Section 12, “Error Messages.”

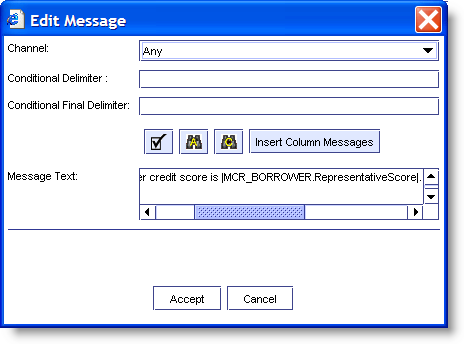


Figure 44: Edit Message Dialog

To edit an existing message:

1. Select the message you want to edit and click Edit.

To delete a message:

1. Select the message you want to delete and click Delete.

# Managing Guidelines

## Search Policies

Use the Search Policies screen to find and display guidelines and parameters. Search for guidelines and activations based on date, date synonym, attribute, context and template. These criteria can be used singly or in combination with one another.

 Note: To include parameters in a search check the Add Parameters box on the Policy Search Criteria screen.

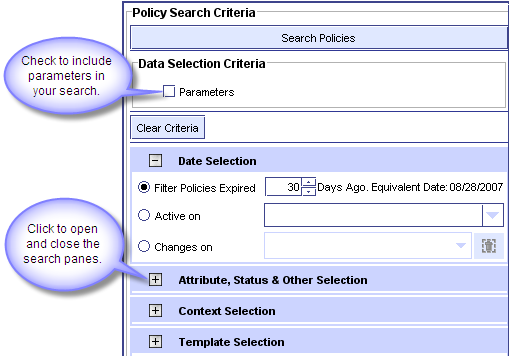


Figure 45: Policy Search window

Use the controls to open and close the Date & Attribute, Context, and Template panes. Click Search Policies to initiate a search and display its results in the Policy Search Result window on the right. The Clear Criteria button clears any search criteria left over from an earlier search. To search for every policy in the system, click the Search Policies button without entering any search criteria.

 Notes: Search Policies will only display guidelines and activations you have permission to Edit or View. Permissions are managed on the Manage Users and Manage Roles tabs under the Admin tab. (See the “[Admin](#_Admin)” section of this document).

The search mechanism performs an “and,” on multiple criteria, not an “or,” i.e., “Find records that meet this criteria **and** this criteria,” not “…this criteria **or** this criteria.”

Search criteria remain active even if a search pane has been closed. If your searches start returning unexpected results, e.g., no records or too many records, click the Clear Criteria button and try the search again.

Date Selection and Attribute Selection Search

Use the Date Selection and Attribute Selection panes to find guidelines based on date and attribute values.

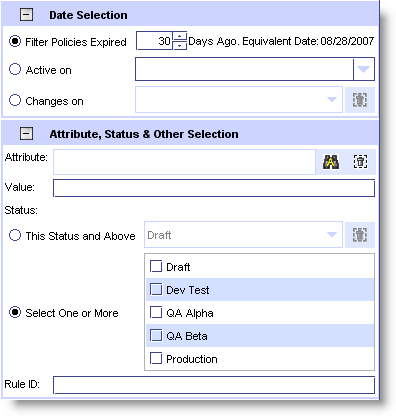


Figure 46: Date Selection and Attribute Selection Search panes

* **Filter Policies Expired**. Select a number representing the number of ago. The date that this corresponds to will be displayed next to the number selection field. Policies that expired earlier than that date will not be in the search results.
* **Active On**. Finds guidelines active on the specified date. Click the down arrow to select a date from the drop down list.
* **Changes On**. Use to find guidelines with an effective (i.e. start) or expiration (end) date on the specified date.

For example, If you have a guideline effective on march 15, and do an Active On search for March 16, the search will return that guideline. However, doing a Changes On search for March 16 will not return that guideline.

* **Attribute**. Use to find guidelines associated with a specific attribute. Click the binoculars to view valid Classes, double click a Class to view its Attributes. Click an attribute to place it in the search field.
* **Value**. Use this field to search on a specific value in the attribute specified in Attribute: above. Only policies with a rule that checks for the specified value will be returned.
* **Status**. This Radio button allows you to choose a particular status (acts as a minimum value returning policies associated with this status or anything above it) or to select any number of individual statuses for your search.
* **Rule ID**. A specific rule identifier to search for.

 Note: The Attribute and Attribute Value fields only work on attributes that have an attribute map. These are defined using the Manage Templates screens, usually by a template author.

#### Date Synonyms

Date synonyms can also be created here. Click the **New** icon to the right of the Active On: and Changes On: fields to create a descriptive label for a specific date. Changing the synonym date will automatically change the date everywhere in the system that synonym is used. For example: The ABC Loan Company planned to offer a new ALT-A product on March 1.  The Business Analyst created a date synonym called “ALT-A Rollout Date” with the value “03/15/2006”, and created guidelines for the new product.  However, as the rollout date drew near, the product offering was delayed by a couple of weeks.  When the analyst changed the value of the date synonym from “03/15/2006” to “03/29/2006”, PowerEditor automatically updated all the guidelines to reflect the new rollout date.

To search by date synonym enter its name in the field or select it from the list in date drop down box (they appear after the numerical dates).

### Context Selection Search

Click the Select Context button to search for policies based on context. Note that if you select multiple contexts for your search PowerEditor will return only those guidelines associated with all of those contexts.

To specify your search criteria:

1. Click the Select Context button to display the Guideline Context Selection window.
2. Click the tabs to display the available entities and categories.
3. Double click or use the “Add to Context” button to place the items you want in the window on the right. If you change your mind use the Remove Selected Context button (or double click) to deselect criteria. The Clear Context de-selects all.
4. When you’ve finished making your selection click the Accept button at the bottom of the screen to return to the Search window, which will display your search criteria.
5. Click Search Policies to execute the search.

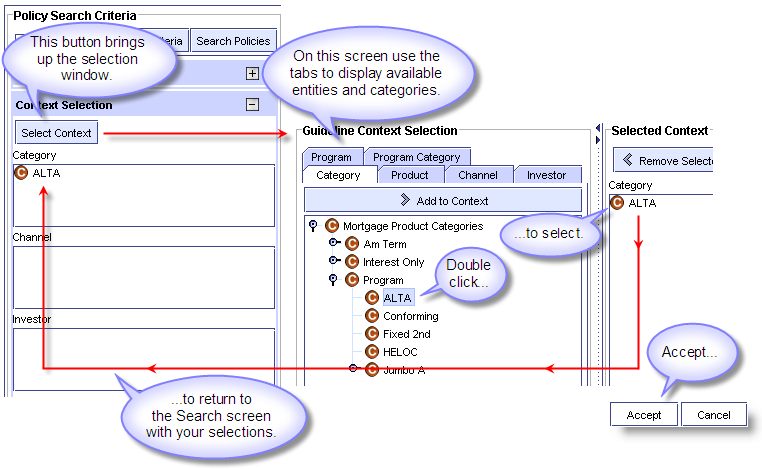


Figure 47: Context Selection

#### Using the Include Controls in a Context Search

Use the Include controls to limit or expand a context-based search. These switches can be used singularly or in combination with each other (Figure 48).

* Include Empty Contexts – includes guidelines in your search for which no context has been set.
* Include Parent Categories - searches for guidelines whose context contains the selected context or its parents. Use this switch to “walk up” the product hierarchy when you have reason to look for guidelines that might apply to your criteria at more general levels in the hierarchy. If you select multiple contexts for your search PowerEditor will return only those guidelines associated with all of those contexts, and those associated with the parents of all of those contexts.
* Include Children Categories – searches for guidelines whose context contains the selected category or its children. Use this switch to “walk down” the product hierarchy when you have reason to look for guidelines that might potentially apply to your search criteria. If you select multiple contexts for your search PowerEditor will return only those guidelines associated with all of those contexts, and those associated with the children of all of those contexts.
* Include Column Data – searches for guidelines with an entity list column that contains the specified categories. If you choose this option your selections for Include Parent Category and Include Children Categories will be applied in these columns as well as the context. The Row column in the search results listing will display the row numbers of the guideline(s) in which the category appears. If you select multiple contexts for your search PowerEditor will return only those guidelines associated with all of those contexts, and column data associated with all of those contexts.

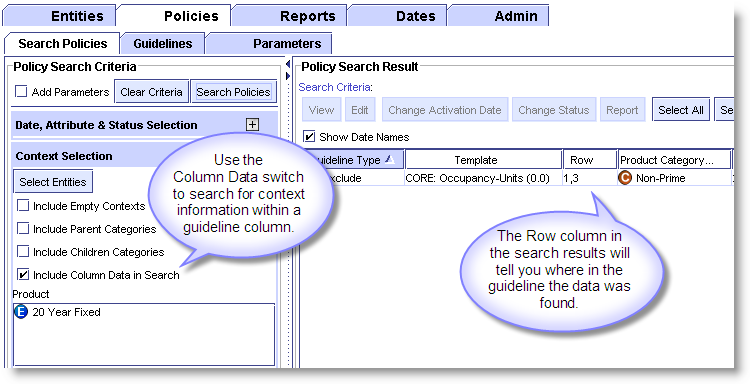


Figure 48: Using the Row column in column data searches.

### Template Selection Search

Use the Template Selection pane to conduct a template-based policy search. Display a list of templates to select from in either Table (alphabetic) or Tree (hierarchical) form.

This is the third type of search in the guideline search panel, and is probably the one you’ll use most often. Double click a template to add it to the search criteria or use the Add to Selection button. Selected templates are displayed under “Name” in the selection pane.

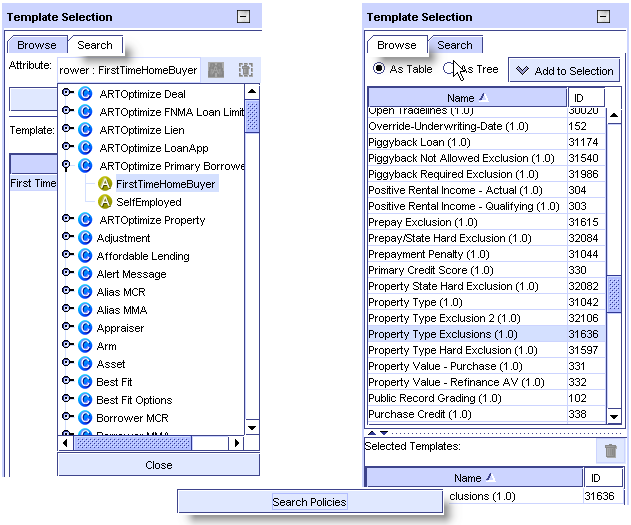


Figure 49: Template Selection Window

Click Search Policies to execute the search.

### Using Search Results

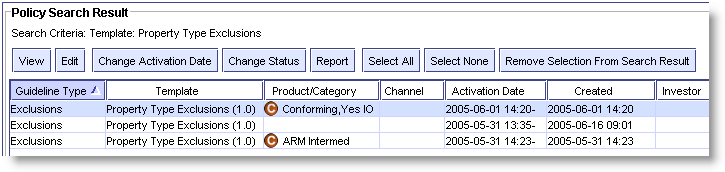


Figure 50: Policy Search Result window

Policy Search Results:

**View** – view selected guideline (on Guideline Detail screen).

**Edit** – modify selected guideline (Guideline Detail screen).

**Change Activation Date** Operates on one or more guidelines.

**Change Status**. Operates on one or more guidelines.

**Report** – generates a report on one or more guidelines. (Reports are discussed below).

Window controls:

**Select All/Select None** – select/de-select all guidelines displayed.

**Remove Selection From Search Result** – remove selected guideline from list.

### Guideline Report

Select the guideline(s) you want generate a report on and click the Report button to display the New Guideline Report dialog box. Use the Options check boxes to specify the format of the report you’d like to see and click Accept. The report will appear in a new browser window and be saved in the directory listed in the Destination File field of the report dialog box.

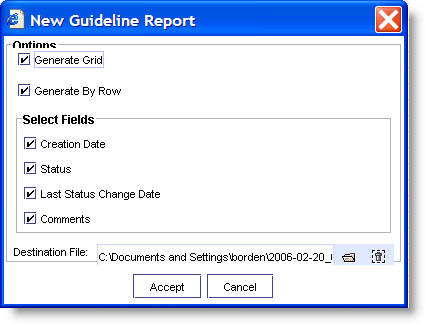


Figure 51: The Guideline Report dialog boxFigure 52 shows a sample guideline report.

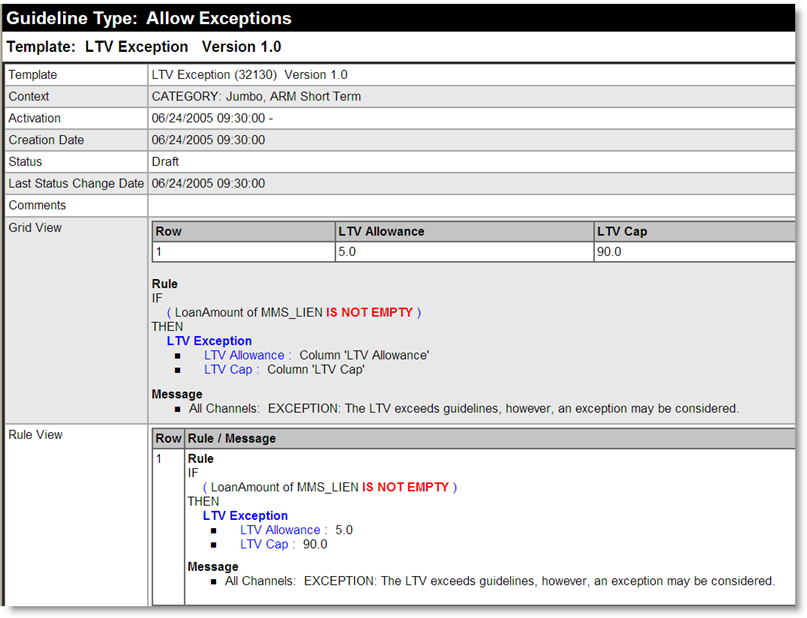


Figure 52: Sample Guideline report

## The Manage Guideline Screen

Use the Manage Guidelines screen to create and modify guidelines. (Note: the tabs on your screen may be different than the ones in the figure, e.g., not everyone will see the Manage Guideline Actions and Mange Guideline Test Conditions tabs.)

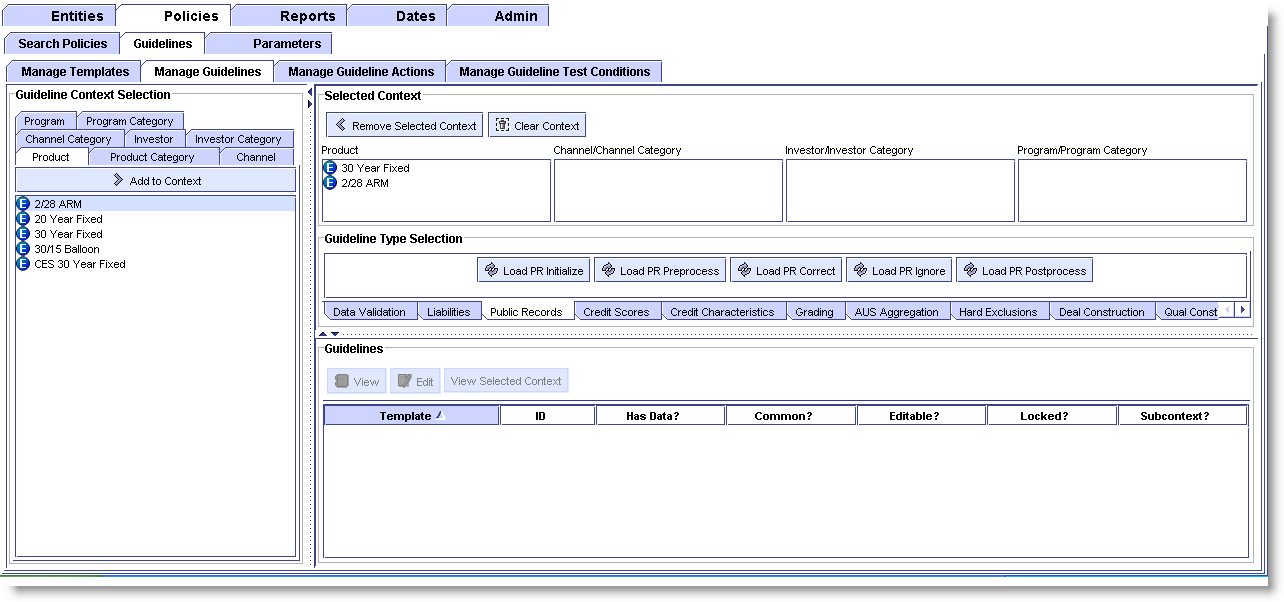


Figure 53: Manage Guidelines screen

### Context Selection

The first step in defining a guideline is to select the guideline context, i.e. to specify the relationship between the new guideline and other elements in the knowledge base (e.g. Product Category, Channel, Investor).

Click the tabs in the Guideline Context Selection pane to display the elements you want to associate with the new guideline. Select the ones you want by double clicking, to place them in the Selected Context pane on the right (or use the Add to Context button).

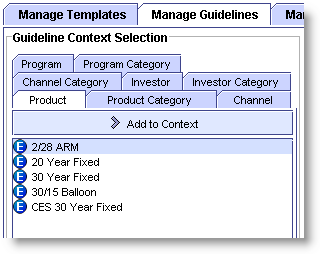


Figure 54: Menu Tabs for selecting guideline context

Use the Remove Selected Context and Clear Context (Clear All) buttons to remove items from the Selected Context pane.

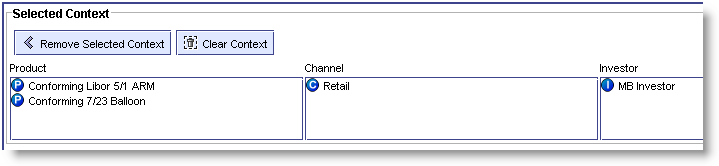


Figure 55: Selected Guideline Context pane

## Associating Guidelines and Entities

This section describes the way PowerEditor associates guidelines with entities and entity categories. In the following discussion products are used as an example, but the same logic applies to all entities. As you recall, the Manage Products screen is used to create and delete entities (products, channels, investors, programs, etc.) and their categories, and to structure entity information hierarchically within categories to show the relationships between them.

You can associate products (or other entities) with one, more than one, or zero (none) categories.

And while a guideline with a single category is automatically associated with all the products in that category, guidelines with multiple categories are not necessarily linked to all the products in those categories.

When a guideline has more than one category PowerEditor uses a simple formula to make the product connections, based on the structure of the product hierarchy. This process is described below. Though the examples given show guidelines and products, the same principles apply to all types of entities. Let’s start with a sample product hierarchy.

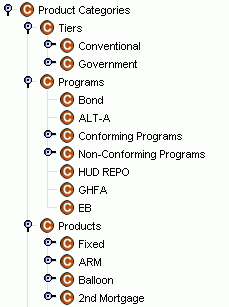


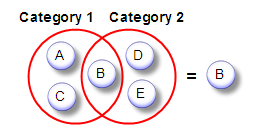
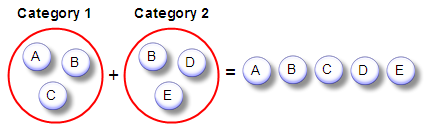
Figure 56: Sample Product Hierarchy

At the top sits the "root" category—in this case one called "Product Categories.” One level down from root we have “Tiers,” “Programs,” and “Products." These second-level categories are called “children” of root and “ancestors” of the categories below them. For example, Products is a child of Product Categories, and the ancestor of Fixed.

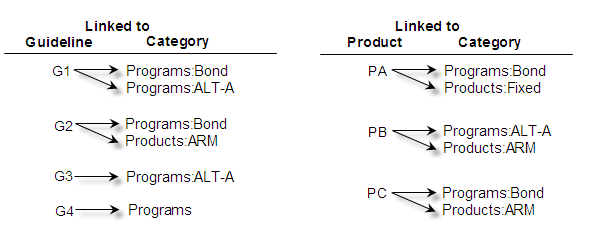
Here’s the formula PE uses to link guidelines and products (or other entities) when two or more categories are involved:

When two categories have the same ancestor under the root directory PE does a set union to make the association, i.e., all the products in either category are linked to the guideline.

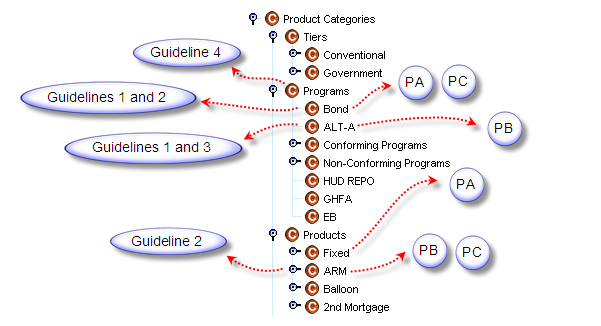
In all other cases, PE does a set intersection of the products. In other words, only the products common to both categories are associated with the guideline.



Let’s walk through a sample set of associations, using these guidelines and categories:



Map them to the hierarchy…



…and apply the formula. Here’s how PowerEditor will create the associations (diagramed below in Figure 57).

Guideline 1 is linked with Products A, B, and C. Its categories have the same ancestor under root, so PE does a set union of products.

* Guideline's 2 is linked with Product C. Its categories have different ancestors, so PE does a set intersection.
* Guideline 3 is linked with Product B (the only one in Programs:ALT-A).
* Guideline 4 is linked with Products A, B, and C. Programs is associated with Products A and C (through Bond) and Product B (through ALT-A).

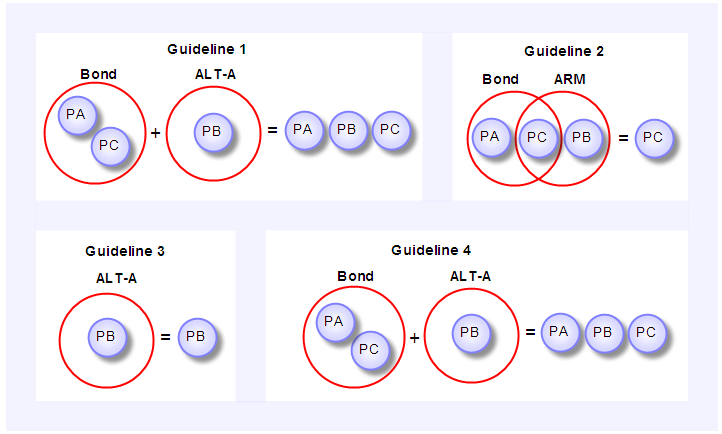


Figure 57: Guideline and entity associations

One final note: when a guideline has more than two categories PE processes the first two. The resulting set of products is then applied to the third category, that result is applied to the fourth, and so on.

### Guideline Type Selection – Template Reference Specification

Once the context has been created (i.e., you have associated your guideline with the relevant entities (e.g. products, categories, etc.), the next step in setting up a guideline is to choose the template to base your guideline on in the Guideline Type Selection pane (Figure 58). Click the tabs and their associated buttons to display template and guideline information for each usage type.

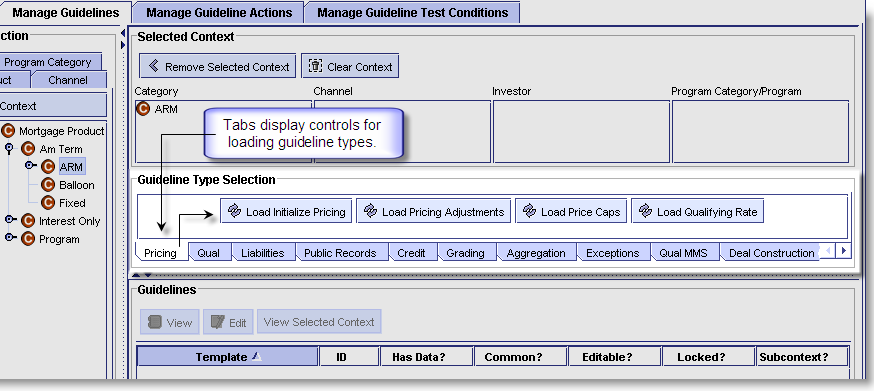


Figure 58: Guideline Type Selection pane

 Notes: The buttons and tabs in this pane on your system are probably different from the ones shown in this guide. Yours have been customized for your system; your PowerEditor administrator or MindBox consultant will be able to discuss them with you in detail. To add or change a guideline type, see your system administrator.

A guideline type is a collection of templates organized by usage type. The Guideline Type Selection pane’s *tab* controls match the template *folders* on the Manage Policies and Search Policies screens. Similarly, the Guideline Type Selection pane’s *button* controls match the *sub-folders* on the Manage Policies and Search Policies screens. The button controls (i.e. sub-folders) are also known as *usage types* in PowerEditor.

### Guideline Selection

Click the button to display a list of templates and guideline information that match the context and type information you’ve specified. Each row in the table represents a guideline. A checked box (affirmative) is blue.

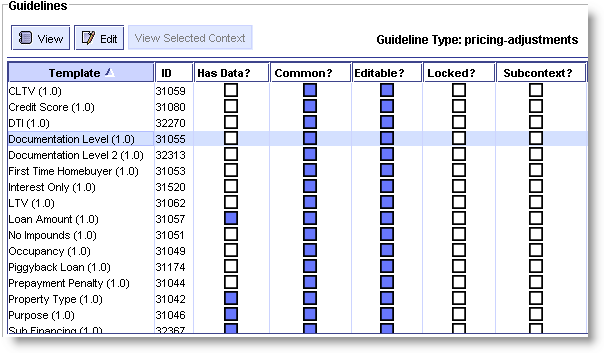


Figure 59: Guidelines pane

The Template and ID columns describe the template associated with the guideline and its specified context, the other columns provide information about each guideline. Select a guideline and click View to display details, or Edit to make changes. If a template has no guideline, i.e., the “Has Data” checkbox is blank, selecting it and clicking Edit will take you to the Manage Guideline screen where you can create a new one.

| *Column* | *Description (Checked: Blue)* |
| --- | --- |
| **Template** | Lists the templates for this usage type and their activations (version number). |
| **ID** | The template’s unique ID (a system-generated value). Note that your system may be configured so that this column is hidden. |
| **Has Data?** | Checked if there’s a guideline associated with this template for the specified context, i.e., it has an activation, a guideline, and grid values. |
| **Common?** | Checked if the guideline shares an association with the same products as another guideline. |
| **Editable?** | Checked if you have permission to edit the guideline. (Editing permissions are managed using the Manage Roles screen.) |
| **Locked?** | Checked if the guideline associated with this template is being edited by another user. |
| **Subcontext?** | Checked if the context you selected is a subset of the contexts associated with this template. |

Table 6: Guideline summary information, Manage Guidelines screen

### Guideline Detail Screen

The Guideline Detail Screen is used to view and edit a specific guideline within the knowledge base. To access the Guideline Detail Screen, you must first select the desired guideline template. To do this, click on the template in the “Guidelines” sub-window described above. Then click the Edit or View button to display the Guideline Detail screen.

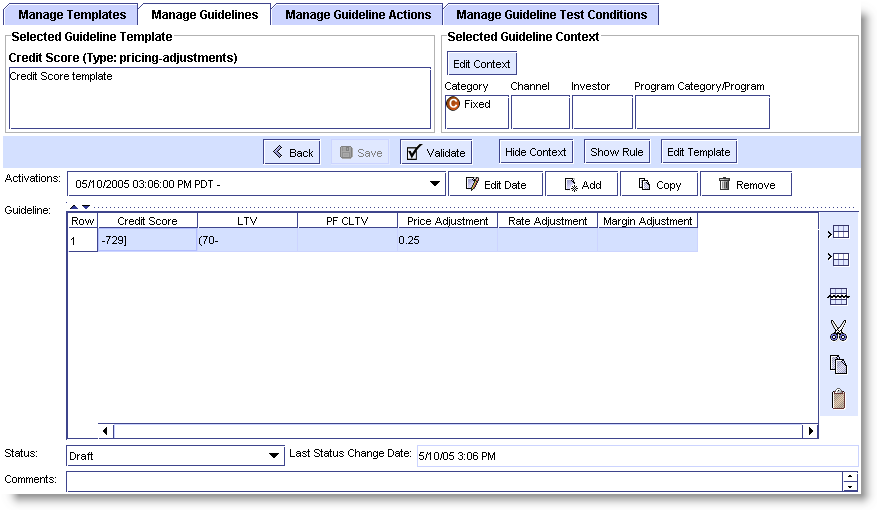


Figure 60: Guideline Detail Screen

This screen contains all of the details of the selected guideline including version information, rule data, context, descriptions, etc. The Guideline Detail Screen is made up of three principal sections. These include:

* Selected Guideline Template
* Selected Guideline Context
* Guideline Details

#### Selected Guideline Template

Located in the top left section of the Guideline Detail screen, this provides the name and a textual description of the template from which the guideline was created.

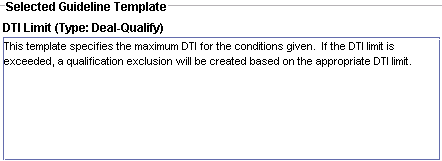


Figure 61: Guideline Template description

The selected guideline’s template cannot be modified in this screen, you need to click on the Back button to return to the previous screen and select a new template.

#### Selected Guideline Context

Located on the top right side, this section of the Guideline Detail Screen provides a summary of all the contexts elements associated with the guideline definition.

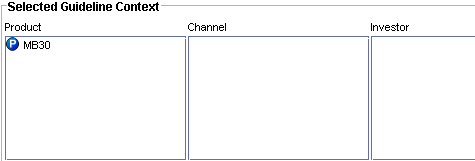


Figure 62: Guideline Context summary

The selected guideline context cannot be modified in this screen. You need to click on the Back button to return to the previous screen and select a new context.

#### Guideline Details

This section is located on the lower part of the Guideline Detail Screen, depicted in Figure 63. It contains the following types of information related to the guideline(s).

* Activations
* Rules
* Deployment Status information
* Comments

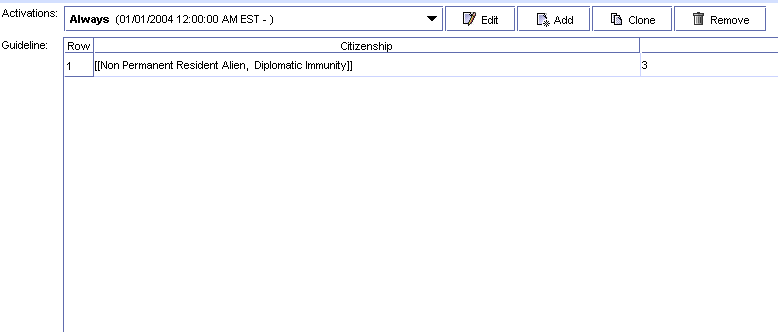


Figure 63: Guideline Details

#### Activations

This drop-down list displays all the Activations for the Guideline. Activations refer to the different versions of the guideline, each of which has a different activation label. Whenever a business policy associated with a guideline changes a new version is created. The rule engine distinguishes versions by their effective date range (activation date, expiration date) denoted by the activation attached to the guideline. This ensures that the appropriate version of the guideline is used for each request.

#### Rules

This includes all of the rules associated with the selected version of the guideline displayed in a grid view. Each row in the grid represents an individual rule in the guideline. The rule(s) can be modified using the Grid View edit operations described in Section 3.3.3.

#### Deployment Status

The Deployment Status of the guideline version being edited.

#### Comments

The comments field provides a place to maintain a description of the guideline version being edited. Typically, it is used to describe any changes that may distinguish this version from the previous one(s).

#### Edit Guideline Toolbar Controls

The buttons on the Edit Guideline toolbar perform the following functions:

* **Back**: returns to the main Guideline Details screen, where selected editing criteria (context, guideline type, etc.) can be adjusted as needed.
* **Save**: Saves the current Guideline Detail Information to the knowledge base (inactive unless something has been modified).
* **Hide Context**: Toggle that shows and hides the guideline context and description panes so you use the entire screen to view and edit. This is useful when working with guidelines with many rules (rows in the grid).
* **Show Rule:** Toggle that shows and hides the Rule pane for viewing and editing guideline rows.
* **Edit Template:** Takes you to the Template Details screen (Guidelines/Manage Template tab) where you can edit the guideline’s template.
* **Validate**: Performs a grid row check to ensure there is no overlap between guideline rules. You can run the check on All Activations (all versions of the guideline), or on the current activation only.
* An Effective Dates overlap occurs when the guideline has activations whose activation date and expiration dates overlap.

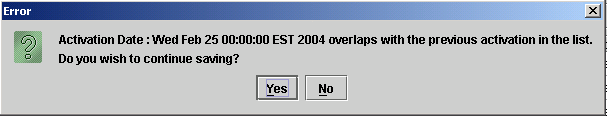


Figure 64: Dates Overlap Message

* A Rule overlap occurs if the guideline version has duplicate rules in the grid view; each rule must be unique in its criteria.

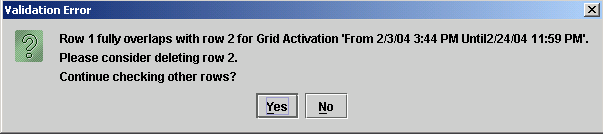


Figure 65: Rules Overlap Message

## Guideline Editor Operations

This section describes step-by-step how to perform common editing tasks (e.g. edit, delete, create, etc.) with guidelines using PowerEditor. *Note that any edit operations you perform are done to a single version (i.e., activation) of the guideline.* A guideline can have multiple versions, each of which will need to be explicitly modified.

### Editing Guidelines

Perform the following steps to edit a guideline version:

1. Select the desired guideline template from the “Guidelines” sub-window.
2. Click on the Edit button. This will bring up the Guideline Detail Screen.
3. Select the version of the guideline you wish to edit from the drop-down list of Activations. It should be identifiable by the effective date range. The most recent activation is shown initially by default.
4. Modify the guideline version as desired.
5. Click on the Validate button to validate your changes.
6. Click the Save button to save your changes.

### Viewing Guidelines

Perform the following steps to v*iew* a guideline version:

1. S*elect* the desired guideline template from the “Guidelines” sub-window.
2. Click on the View button. This will bring up the Guideline Detail Screen. However, in this case the guideline data will be read-only.

### Changing Guideline Effective Dates

You can also modify the effective dates of guideline activations, using the Guideline Detail Screen. Use the following steps:

1. Select the version of the guideline you wish to edit from the drop-down list of Activations.
2. Click on the Edit button. This will bring up the Edit Activation dialog box.
3. Choose another activation.
4. Click on the OK button, then the Save button.

Note that the expiration date and time of guideline and parameter activations with a status other than Draft may not be earlier than the current date and time. However, this restriction may be by-passed. The easiest way is to create a copy of an existing activation, set the activation date for the new copy, and select the option to automatically expire the current activation. You can also change the status to Draft, perform the operation, and then change the status back to what it was.

### Creating Guidelines

There are two methods of creating a new guideline version: the first is to create a new empty version and populate all of the required details. The second is to copy an existing version.

Use the following steps:

1. Select the desired guideline template from the “Guidelines” sub-window and click Edit to bring up the Guideline Detail screen.
2. To create a new guideline version click the Add button.

To copy an existing guideline version click the Copy button.

1. In the Edit Activations dialog box select the desired activation dates.
2. Modify the new guideline version as desired (e.g. add rules to grid).
3. Click the Validate button to validate your changes.
4. Click the Save button to save your changes.

### Removing Guideline Versions

Perform the following steps to remove a guideline version:

1. Select the version of the guideline you wish to edit from the drop-down list of Activations.
2. Click the Remove button.
3. At the confirmation prompt select “Yes” to permanently delete the selected entity.

# Parameters

Parameters are similar to global variables that can be viewed and modified by users through PowerEditor. The value assigned to the parameter can vary based on different criteria including entities (e.g. channel, investor) and input data (e.g. loan amount). Like guidelines, parameters are created from templates. A Parameter Template is representative of the parameter type. For example, a template could be defined to declare a parameter type called “Maximum Loan Amount.” New parameter instances could subsequently be created from this template for the purpose of assigning a value to the parameter based on channel, investor, input data, etc.

Use the Parameters tab to create and maintain parameter instances using the existing parameter templates (Figure 66). Use the list of Parameter Templates on the left to select templates you want to work on.

The Edit Context button brings up the Edit Parameter dialog.

 Note: To create a new parameter type you will require a new parameter template. New parameter templates are defined outside of PowerEditor in the Parameter Template Definition file. Contact your PowerEditor administrator if changes to parameter templates are required.

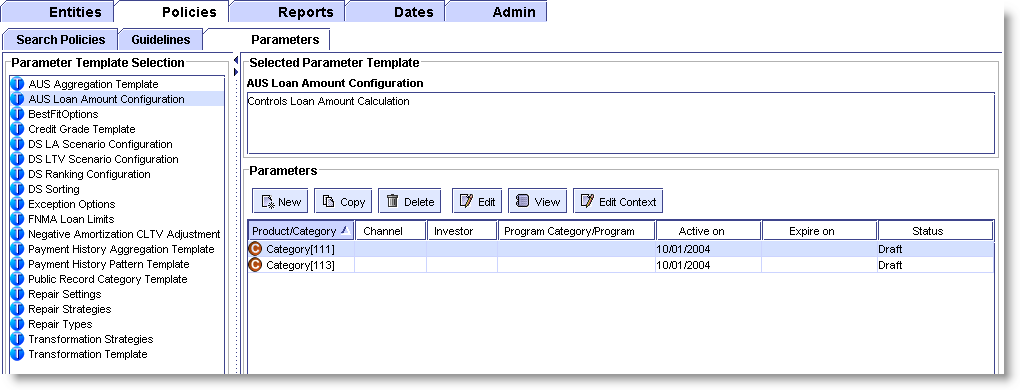


Figure 66: Parameters Module screen

## Creating Parameters

There are two ways to create new parameter activations. The first is to create a new empty parameter and populate all of the required details, the second is to copy an existing parameter, along with all its data and associations, and then edit the copy.

### Creating a new parameter:

1. Click the **New** button and display the New Parameter dialog box (Figure 67).
2. Select a category and/or product as the context for this parameter.
3. Select a channel (if the assigned value will be channel-specific).
4. Select an investor (if the assigned value will be investor-specific).
5. Enter or select (using the pull-down list) an activation date, i.e. the date you want the new parameter to become effective. Use the format “mm/dd/yyyy hh:mm” (month/day/year/hour/minute). This is a required field.
6. Provide an expiration date, if desired. A parameter without an expiration date will never expire.
7. Select a deployment status from the drop-down list, and click OK.

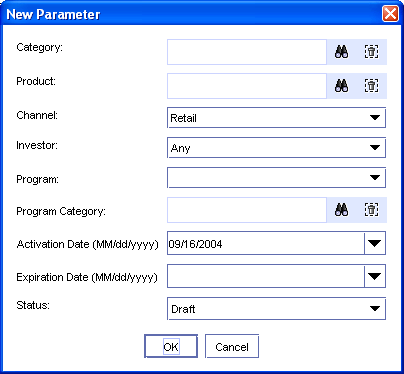


Figure 67: New Parameter Dialog

### Copying an Existing Parameter

To copy an existing parameter:

1. Select the parameter you wish to copy from the Parameter pane and click the Copy button.

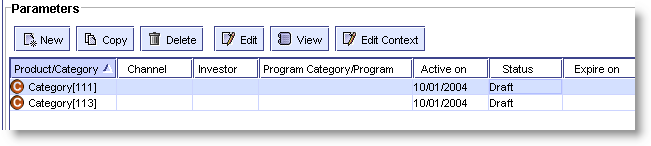


Figure 68: Parameter listing and toolbar

1. In the Copy Parameter dialog, change the copy’s context to distinguish it from the original by modifying it’s entity values, activation, or expiration dates.
2. Provide an activation date and expiration date (mm/dd/yyyy hh:mm). An activation date is required, an expiration date is not. The expiration date and time may not be earlier than the current date and time. (This restriction may be by-passed, please see “Changing Guideline Effective Dates,” on page 71.)
3. Select the deployment status.
4. Click the OK button. The new parameter will appear in the parameters listing window.

## Editing Parameters

To edit an existing parameter, select it from the list and click Edit. The parameter edit screen contains two sections: at the top, the parameter template name and description are displayed. Below it, the Parameters Pane displays the parameter rule data in a grid view. The rule data is used to set the value of the parameter based on the input. For instance, the figure below shows two rules that set the parameter value based on the input Amortization Term.

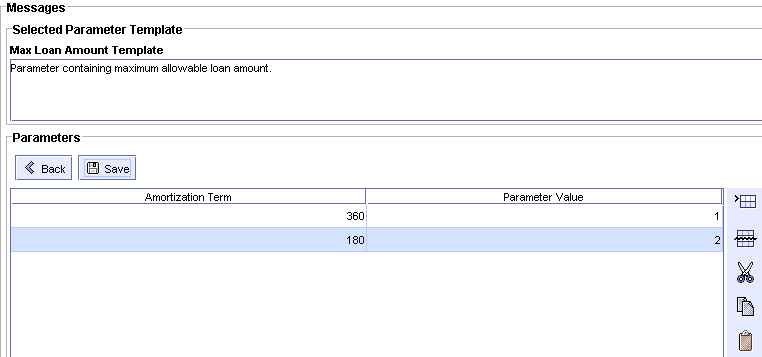


Figure 69: Parameter edit screen

You can edit the parameter data using normal grid view operations. The Back button takes you to the main Parameters screen. **Save** saves the current parameter data, remember to save your work before exiting the current screen.

## Viewing Parameters

To view a parameter, select it from the listing and click View. You can see the parameter details but they won’t be editable.

## Deleting Parameters

To delete a parameter, select it from the listing and click Delete. You will be prompted for confirmation.

View Custom ReportS

Starting with version 4.4.0 version of PowerEditor, you can view custom PowerEditor reports created with Crystal Reports software. Once custom reports have been defined a list of them will appear when you click the Reports tab. Click Refresh to update the list, click View or double click a report in the list to display it in a new browser window. For more information on creating custom reports please see the *PowerEditor Custom Reports Guide.*

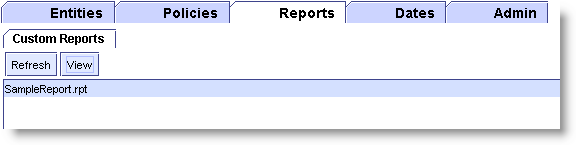


Figure 70: Reports tab

 Note: Crystal Reports that contain sub-reports cannot be viewed from PowerEditor, but can be viewed in the Crystal Reports Developer software.

# Dates & Date Synonyms

A date synonym is a “date stand-in” that allows you to associate a descriptive text string with a particular date, e.g. New Years Day, or “June 15 Rollout.” Use the dates tab to create date synonyms. Date synonyms are particularly useful when you know a date might change. Changing the date of a synonym changes that date everywhere that synonym is used.

Figure 71 gives some examples.

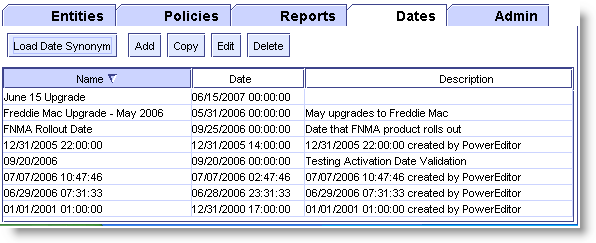


Figure 71: Examples of date synonyms.

Use the Dates tab to create, modify, and delete date synonyms in PE, or use the synonym icons that appear next to most date fields. Clicking the New Date icon brings up the New Synonym dialog box (Figure 72).

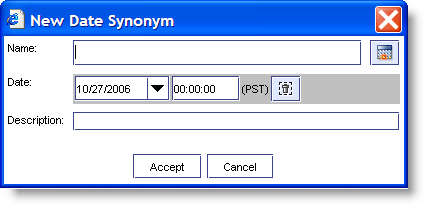


Figure 72: Creating a new date in PE.

Use the Show Date Names checkbox on the Edit Guideline screen to change the way dates are displayed, toggling it to display synonym names and actual dates (Figure 73).

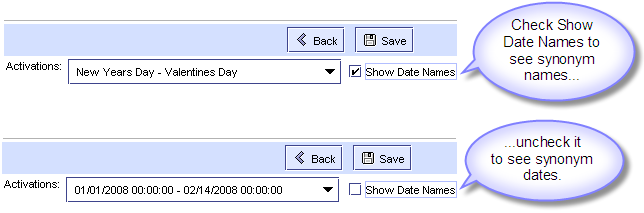


Figure 73: Date synonyms and the Show Date Names box.

To Add a Date Synonym:

1. On the Dates tab click the Load Date Synonym button to display all date synonyms.
2. Click the Add button.
3. Enter a synonym name, date, and description. Click the arrow to select a date using a calendar, click the trashcan icon to clear the field.
4. Enter a description for the synonym and click Accept.

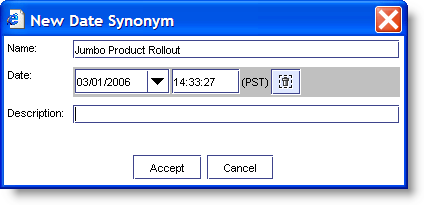


Figure 74: Create date synonym

To copy a date synonym:

1. Click the Load Date Synonym button to display all date synonyms.
2. Select the synonym you want from the list and click Copy.
3. At the prompt enter the desired name, date, and description information (see “Add a Date Synonym,” above), and click Accept.

To edit a date synonym:

1. Click the Load Date Synonym button to display all date synonyms.
2. Select the synonym you want from the list and click Edit.
3. At the prompt update the name, date, and/or description information (see “Add a Date Synonym,” above), and click Accept.

To delete a date synonym:

1. Click the Load Date Synonym button to display all date synonyms.
2. Select the synonym you want to delete from the list and click Delete.

# Administration

The Admin module provides a set of tools for performing administrative types of tasks, including managing user access and security, and knowledge base generation. Access to the Admin tab should be restricted to administrators or super users.

## Deploy to Rule Engine

This function of the Admin module allows a user with the proper accesses to deploy the knowledge base to the rule engine. Within PowerEditor deployment refers to the process of extracting knowledge-base data (e.g. guidelines, parameters, etc.) and converting it to MindBox ART\*Enterprise code (see ART\*Enterprise documentation) which can be executed by the rule engine application. The location of the generated code files is a configurable parameter in the static PowerEditor configuration settings.

After deployment, the generated ART\*Enterprise code can then be picked up by the rule engine directly or incorporated into an ART\*Enterprise deployed binary image which can by transported to any Rule Engine platform.

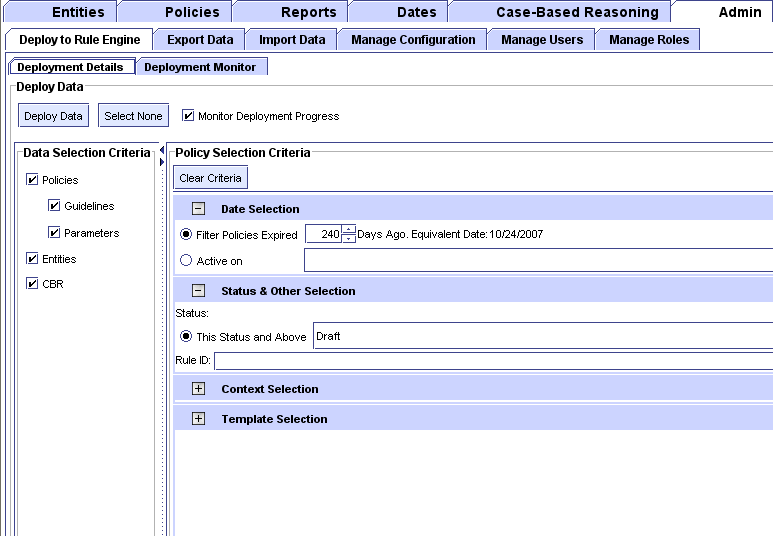


Figure 75: Admin: Deployment Details screen

### Deployment Details

This section of the screen provides a set of controls that can be used to control the deployment process. When you’ve selected the parameters you want click the Deploy bar at the top of the window.

The Policy Selection Criteria works the same as the one on the Deploy or the Search Policies screen.

#### Deploy Rules within Status

Use this setting to specify which items to extract from the knowledge base for the current deployment, depending on the purpose of the Rule Engine(s), e.g. for testing, or production. Make sure the status you select is appropriate for the intended rule engine environment.

The generated files resulting from the deployment process include all of the knowledge-base items having a current deployment status of the selected value or a level(s) below. For example, a deployment for the test environment would not only include those items with a status of QA but also anything with a status of production as well. Including the production data is required to produce a complete knowledge base. Using only items with a status of QA would result in an incomplete knowledge base. However, items with a status of Draft (development) would not be included. The following illustrates the previously defined typical set of deployment status’ and the lower-level status that would be included in a deployment.

|  |  |
| --- | --- |
| **Deployment Status** | **Included Deployment Status** |
| Draft | Draft, QA, Production |
| QA | QA, Production |
| Production | Production |

Table 7: Deployment Status Inclusion Table

#### Filter Policies By Date

The **Filter Policies Expired**  controls allow you to apply date-based filters, e.g. anything older than 30 days. The default date can be set in the configuration file. The **Active On** controls allow you to deploy only the policies that are active on a particular date.

#### Export Entities to Server

The “Entities” checkbox allows you to export changes to entity information and rules in a single step. This option should remain checked (unless you have a specific reason for not exporting entities).

#### Generate Guideline Rules

This **Guidelines** option controls the generation of Guideline Rules. If checked all guideline rules with the appropriate deployment status are generated as ART\*Enterprise rules.

#### Generate Parameters

This **Parameters** option is used to control the generation of Parameters. Parameters are generated as ART\*Enterprise object instances as opposed to rules. If this field is checked, all Parameters with the appropriate deployment status will be generated and written to the file parameter-instances.art. For example, the generated file “C:\ruleserver\SERVERS\SRC\Production\ parameter-instances.art” would be created from a deployment with a selected status of Production.

#### Filter Policies By Guideline Type/Templates

The Filter Template section allows the user to further restrict the deployment process so that only the guidelines of the selected usage type or templates are generated. Select the desired usage type or templates from the tree.

#### Monitor Deployment Progress

When checked, the screen displays the results of the deployment process on the Deployment Monitor.

### Deployment Monitor

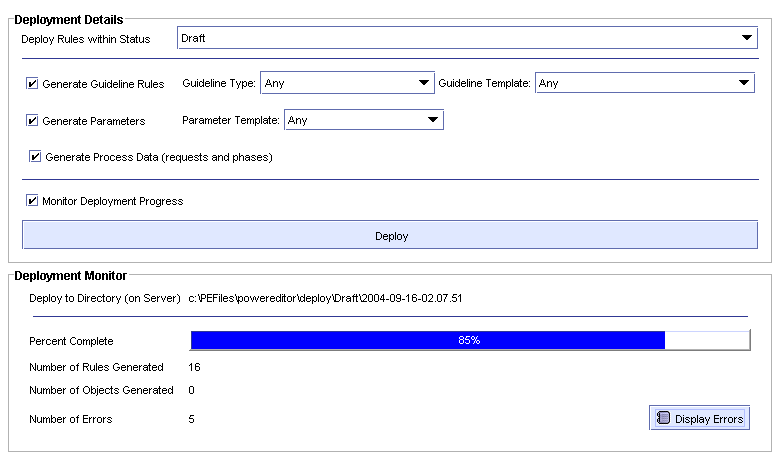


Figure 76: The Deployment Monitor

The Deployment Monitor displays deployment progress and results, including:

* **Deploy to Directory (on Server)** - Name of the directory where all the generated files are written. The full directory name is comprised of two parts: the Base Directory and Deployment Status. The Base Directory is stored in the static configuration settings. The deployment status is appended to the end of the base deploy directory (e.g. c:\base-directory\draft) to form the full deployment path.
* **Percent Complete** - deployment completion status.
* **Number of Rules Generated** - number of rules that have been generated.
* **Number of Objects Generated** - number of objects that have been generated.
* **Number of Errors** - displays the number of errors encountered during the deployment. The Display Error button appears if there have been any errors, click on it to view error messages. Error types are described below.

### Status-based Deployment Authorization

PowerEditor supplies a deployment privilege for each deployment status. A user must have the authorization for the selected status to perform deployment.

### Deployment Troubleshooting

Deployment errors include both fatal and non-fatal errors. Fatal errors cause the deployment process to terminate and are primarily due to template (e.g. domain and guideline) problems that prevent a proper rule from being formed. These are typically corrected using the Template Editor. Non-fatal errors do not stop the deployment process and allow syntactically correct rules to be formed.

If any errors occur (either fatal or non-fatal), open the deployment error log file with the **Display Errors** button to determine the cause. An error log file will be automatically created in the directory where the deployed rules/objects are created (e.g. …/draft, …/QA, etc.). The file name will be “deploy-errors.txt.” Each line in the error file represents a single deployment error report, and each report will include the following information:

* Date/Time the error was encountered
* Item in which the error occurred
* Error details

Using the log entries, make any required corrections and re-attempt the deployment process. Here’s a sample error log entry from deploy-errors.text:

2004-03-02 23:10:025 EST [Guideline:Minimum Loan Amount,act=01/21/2003-,row=1] Error generating rule for Rule=GuidelineRule27; Ruleset=DEAL-QUALIFY

PatternSet[type=1,size=3; objPattern...SUBJECT\_PROPERTY\_DATA/?prop1/prop1

AttrPattern[RESIDENCY\_TYPE in Column 1]

; objPattern...DEAL/?deal1/deal1

PatternSet[type=1,size=2; AttrPattern[nominal\_credit\_grade in Column 2]; AttrPattern[lien null null]

; objPattern...lien/?lien/null

AttrPattern[loan\_amount\_dollar < Column 3]

; rhs=pe\_action(Name[token=function\_name,optional=NodeListOptional[]] Name[token=disqualify\_product,optional=NodeListOptional[]] Name[token=message\_code,optional=NodeListOptional[]] 4007 Name[token=deal\_id,optional=NodeListOptional[]] "|DEAL.DEAL\_ID|" Name[token=rule\_handle,optional=NodeListOptional[]] "%ruleName%" Name[token=rule\_type,optional=NodeListOptional[]] "deal\_qualify" ): Could not locate attrib lien of class DEAL

## Export Data

The Export Data and Import Data functions provide a mechanism for exporting some or all your PowerEditor data to an XML file. This is useful when you are migrating from one version of PowerEditor to another, or, for example, if you wanted to export entities to a file and have the Rule engine read them back in.

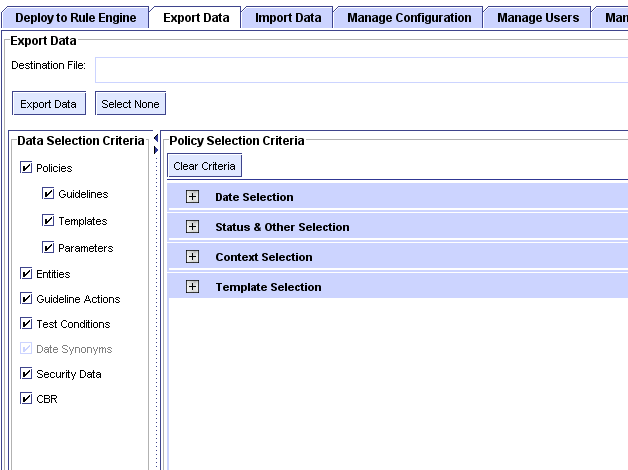


Figure 77: Export Data tab

You can also use these functions to feed PowerEditor data to another system, or import data from another system into PowerEditor.

The Policy Selection Criteria works the same as the one on the Deploy or the Search Policies screen.

## Import Data

Use the Import Data tab to import data in an XML file into PowerEditor, this is also a useful utility for updating template and guideline data (see below). Note that PowerEditor does not support importation of data exported from newer PE versions. (This will work periodically, but usually it does not).

Click Add File to display a window that lets you search your system for files, these are added to the Import File list. Use the Remove File button to delete a file from the list. When you have completed your file specification click the Import bar to start the process and display the Import Results window, where process messages, including errors, are displayed. Highlight a message and click the Copy Message button to copy it to the clip-board.

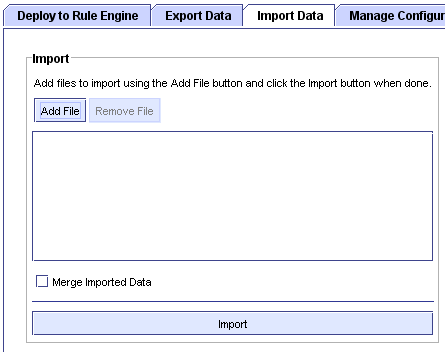


Figure 78: Import Data tab

Template and guideline records include a unique system-generated ID number, these are included in the XML file when you export a record from PowerEditor. When you import template and guideline records with the same ID number, any existing records in your system will be updated (overwritten). If you want template and guideline records with the same ID number *added* to your system (rather than having existing records updated), check the **Merge** checkbox on the Import Data screen. For example:

|  |  |  |
| --- | --- | --- |
| **If You Create a Template With** | **And Import Merge Is** | **Result** |
| The same ID and different data |  | The imported template will replace the old template. |
|  | A template with a new ID will be created using the imported data. |
| A different ID and the same data |  | A template with the import's ID will be created. |
|  | A template with a newly generated ID will be created. |

Table 8: Merge results.

Manage Configuration

Use the Manage Configuration tab to re-load domain parameters.

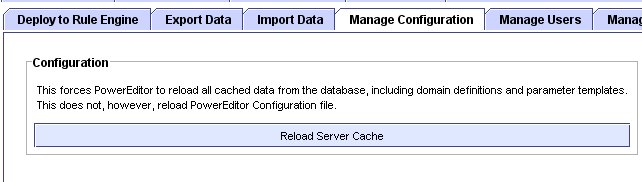


Figure 79: Manage Configuration tab

Click the button to reload the PowerEditor server cache, including:

* Domain classes and attributes
* All entities
* Guideline template templates
* Guidelines
* Parameter template definitions
* Parameters
* Phase and Process data

That is, all PowerEditor data except the static configuration settings that were established when PowerEditor was installed. Contact your PowerEditor administrator if changes to the static configuration settings need to be made.

## Manage Users

Use this tab to create and modify PowerEditor user accounts, including ID’s, passwords, and roles (Figure 80).

**Note that admin users cannot edit their own roles, nor can they see policy data.**

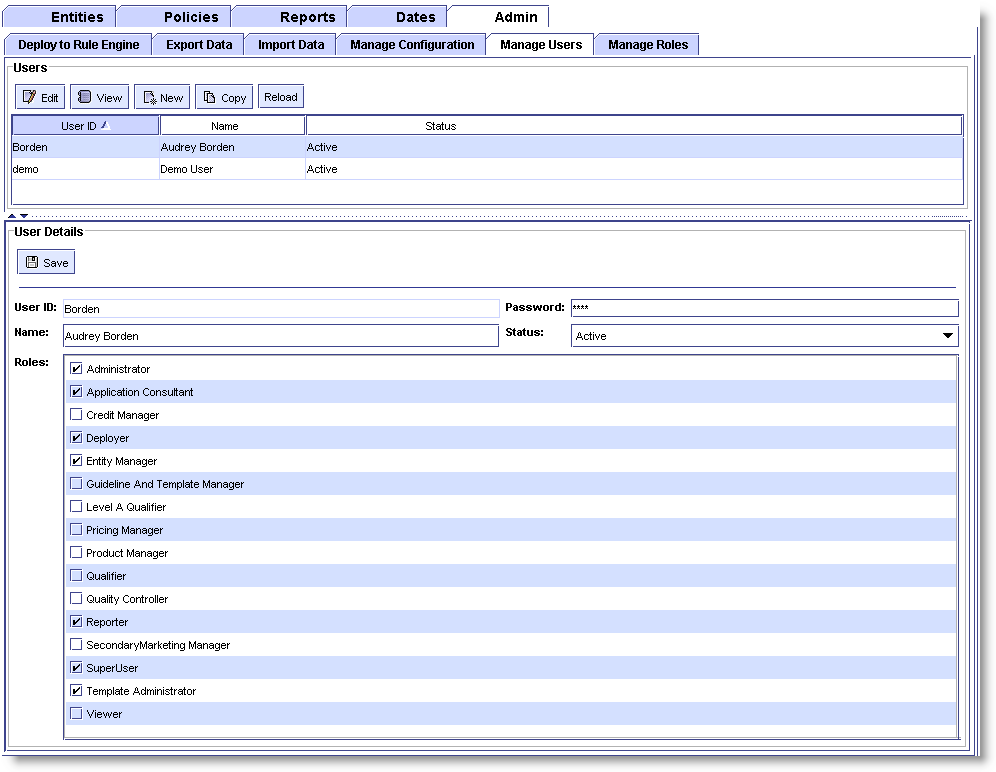


Figure 80: Manage Users screen

* Click Edit to make changes to an existing user account.
* Click View to see a read-only version of the user account.
* Use the New button to create a new blank account, entering user data and specifying access from scratch, or Copy to duplicate an existing account which can then be edited for a new user.
* Click Remove to delete an account.

After modifying or creating an account remember to save your work by clicking **Save**.

### User Information

All fields are required:

* *User Id* - A unique alphanumeric identifier for a user.
* *Password* – Password for the user account.
* *Name* – Full name (first, last) of the user.
* *Status* – Current activity status of the account. Select from the drop-down list. Current values include “Active” and “Inactive.”

### Roles

Users can only access a PowerEditor function if they have been assigned a role that has that privilege. To assign privileges to those roles use the Manage Roles screen.

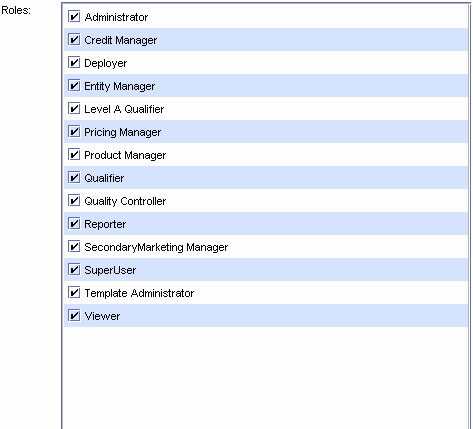


Figure 81: Roles window

## Manage Roles

Because users can wear many hats they can have more than one role, e.g., a user might be working as a System Administrator, a Credit Manager, and a Product Manager. Each role has a set of permissions and privileges required to perform their job. Use the Manage Roles screen to create user roles and set permissions for them (Figure 82). Privileges are associated with roles on this screen and assigned to individuals using the Manage Users screen.

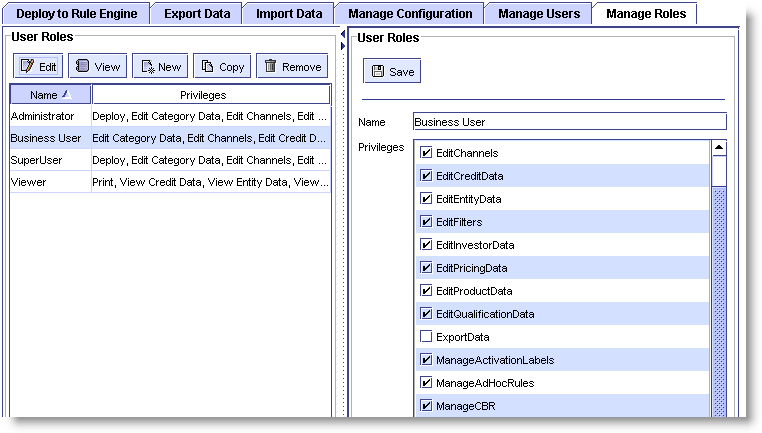


Figure 82: The Manage Roles screen

Click the Edit button to edit a role. You can change the name of the role; add and remove privileges by clicking the checkbox next to the privilege name.

* Use View to see a read-only version of the user role.
* Use New to create a new role. Enter the role name and assign the desired privileges.
* Use Copy to create a new role using an existing one as a starting point.
* Use Remove to delete a role.

### Role Assignment Restrictions

Note that for security reasons, a role with Manage Roles or Manage Users privilege cannot contain any other roles. It is possible, however, to have a role that contains both Manage Roles and Users privileges.

### Screen-Based Permissions

Most PowerEditor permissions are set on a screen-by-screen basis, i.e., users either have access to a particular screen or they do not. Users without access will see neither the screen or its tab. Users given access to the screen will be able to see and edit everything on the screen (though Entity Type and Usage Type permissions may effect this somewhat).

Usage Type and Entity Type permissions are set at a finer level of detail. A discussion of how they work and their effect on different PowerEditor screens is discussed in detail in sections 10.6.2 and 10.6.3, below. The following table (Table 9) provides an overview of permissions set on a screen-by-screen basis.

| **Privilege** | **Effect** |
| --- | --- |
|  Access to WebService API | A user without this privilege does not have access to WebService API. |
|  Access to WebService API | A user with this privilege has access to WebService API. |
|  Deploy | The Deploy tab will be hidden. Users will not be able to display or use the screen. |
|  Deploy | The Deploy tab will be visible to users with this privilege. Users can display and use all the elements on the screen. Note: the rules they can deploy may be effected by their usage type privileges. |
|  Export Data | The Export Data tab will be hidden. Users will not be able to display or use the screen. |
|  Export Data | The Export Data tab will be visible to users with this privilege.  Users can display and use all the elements on the screen. |
|  Import Data | The Import Data tab will be hidden.  Users will not be able to display or use the screen. |
|  Import Data | The Import Data tab will be visible to users with this privilege.  Users can display and use all the elements on the screen. |
|  Manage Configuration | The Manage Configuration tab will be hidden.  Users will not be able to display or use the screen. |
|  Manage Configuration | The Manage Configuration tab will be visible to users with this privilege.  Users can display and use all the elements on the screen. |
|  Manage CBR | The Manage CBR (Case Based Reasoning) tab will be hidden.  Users will not be able to display or use the screen. |
|  Manage CBR | The Manage CBR tab will be visible and users with this privilege can display and use all the elements on the screen. Note: the CBR feature must be turned on in the PowerEditor configuration file. |
|  Manage Date Synonym | The Manage Date Synonym tab will be hidden.  Users will not be able to display or use the screen. |
|  Manage Date Synonym | The Manage Date Synonym tab will be visible to users with this privilege.  Users can display and use all the elements on the screen. |
|  Manage Guideline Actions | The Manage Guideline Actions and Manage Guideline Test Condition tabs will be hidden. Users will not be able to display or use these screens. |
|  Manage Guideline Actions | The Manage Guideline Actions tab will be visible to users with this privilege. Users can display and use all the elements on the screen. Note: This privilege has no effect on the Manage Templates tab. |
|  Manage Parameters | The Manage Parameters tab will be hidden. Users will not be able to display or use the screen. Users without this privilege should not be allowed to search for parameters in the Search Policies screen. |
|  Manage Parameters | The Manage Parameters tab will be visible to users with this privilege.  Users can display and use all the elements on the screen. |
|  Manage Phase | Manage Phase tab will be hidden.  Users will not be able to display or use the screen. |
|  Manage Phase | The Manage Phase tab will be visible; users can display and use all the elements on the screen. Note: the Process feature must be turned on in the PE configuration file. |
|  Manage Request Type | The Manage Request Type tab will be hidden.  Users will not be able to display or use the screen. |
|  Manage Request Type | The Manage Request Type tab will be visible; users can display and use all the elements on the screen. The Process feature must be turned on in the PE configuration file. |
|  Manage Templates | The Manage Templates tab will be hidden.  Users will not be able to display or use the screen. |
|  Manage Templates | The Manage Templates tab will be visible; users can display and use all the elements on the screen. Note: This privilege has no effect on screens under the Manage Policies tab (see discussion of usage type permissions, below). |
|  Manage Roles | The Manage Roles tab will be hidden to users. Users will not be able to display or use the screen. |
|  Manage Roles | The Manage Roles tab will be visible. Users can display and use all the elements on the screen.  Note that with this privilege checked, only Manage Users privilege can be selected; no other privileges can be selected for the same role. |
|  Manage Users | The Manage Users tab will be hidden to users. Users will not be able to display or use the screen. |
|  Manage Users | The Manage Users tab will be visible. Users can display and use all the elements on the screen.  Note that with this privilege checked, only Manage Roles privilege can be selected; no other privileges can be selected for the same role. |
|  View Report | The View Report tab will be hidden.  Users will not be able to display or use the screen. |
|  View Report | The View Report tab will be visible to users with this privilege.  Users can display and use all the elements on the screen. |

Table 9: Privileges set on a screen-by-screen basis

### Entity Permissions

Entity Type permissions control access to entity-based operations in PE, e.g. Products, Programs, etc. For each entity type defined in the PowerEditor configuration file two checkboxes will appear on the Manage Roles screen, e.g.:

 View Product Data

 Edit Product Data

Users granted View permission will be able to see but not edit the entities and categories for that entity type. Users granted Edit permission will be able to view and edit both entities and categories for that entity type. Users without View or Edit permission will not see that entity tab in PE. (Note that when Edit permission is given, permission to View is granted automatically whether the View box is checked or not).

Table 10 provides an overview of the Entity Permission settings and their effects on PowerEditor screens and tabs. In the table the entity “Product” is used as an example but all entities will work the same way. Note that when selecting context or searching for guidelines on the Search Policies screen, or when entering grid data, all entities and entity categories will be available regardless of how the entity type permissions have been set.

| **Entity Privileges** | **Manage Product**  **Screen** | **Effect** |
| --- | --- | --- |
| **View Only Permission**   View Product Data   Edit Product Data | Product pane | View Button will be enabled; the buttons New, Copy Edit, Clone, and Remove will be disabled. |
| Product Details tab | When a product is selected it’s information will be displayed but not editable. |
| Product Category tab | Category hierarchy is viewable, but categories can not be selected or unselected (checked or unchecked). The buttons Add, Edit, and Delete will be disabled. |
| **Edit Permission**   Edit Product Data   View Product Data  Alternate setting:   Edit Product Data   View Product Data | Product pane | All buttons and controls enabled. |
| Product Details tab | All screen data editable. |
| Product Category tab | All buttons and controls enabled. Category hierarchy is viewable and nodes can be expanded and collapsed. |
| **Disabling Permission (No Access)**   View Product Data   Edit Product Data | Product tab | Product tab does not appear. |

Table 10: EntityType Permissions

### Usage Type Permissions

Usage Type permissions control template and guideline operations. For each usage type privilege specified in the PowerEditor configuration file four checkboxes appear on the Manage Roles screen:

 Edit *(UsageType Privilege)* Guidelines

 Edit *(UsageType Privilege)* Templates

 View *(UsageType Privilege)* Guidelines

 View *(UsageType Privilege)* Templates

Users granted Edit permission will be able to create, read, update, and delete the templates and guidelines associated with the specified usage type privilege. Users granted View permission will be able to see but not edit templates and guidelines associated with the privilege, and users with neither Edit or View permission will see neither templates or guidelines associated with that privilege.

To eliminate clutter on the Manage Roles screen the privilege attribute of the <UsageType> element (in the config file) can be used to group multiple usage types into a single permission category. For example, the usage types CRA-Initialize-Liabilities, CRA-Ignore-Liabilities, and CRA-Preprocess-Liabilities could be consolidated under the privilege “CreditData:”

<UsageType name="CRA-Initialize-Liabilities" displayName="LB Initialize" privilege="CreditData"/>

<UsageType name="CRA-Ignore-Liabilities" displayName="LB Include" privilege="CreditData"/>

<UsageType name="CRA-Preprocess-Liabilities" displayName="LB Exclude" privilege="CreditData"/>

On the Manage Roles screen the three usage types would then appear under a single set of permission controls:

 Edit CreditData Guidelines

 Edit CreditData Templates

 View CreditData Guidelines

 View CreditData Templates

The following three tables illustrate how to use the settings on the Manage Roles screen to grant different kinds of privileges and the effects privilege settings will have on PowerEditor screens and operations. Note that when Edit permission is given, permission to View is granted automatically whether the View box is checked or not.

| **View Only Permissions for Usage Type** | | |
| --- | --- | --- |
| **Settings** | **Screen** | **Effect** |
|  View Guidelines   Edit Guidelines | Manage Guidelines screen | Edit button disabled. |
| Edit Guideline screen | Information on this screen will not be editable. |
| Search Policies screen,  Search Results pane | Edit button disabled. |
|  View Templates   Edit Templates | Manage Templates screen | The following buttons will be disabled: New Template, Copy, New Version, Delete Template, Edit, and Save. |
| Edit Guideline screen | The Edit Template button is labeled “View Template.” |
| New Template wizard | Usage types for which user does not have edit privileges will not appear in the dropdown list. |

Table 11: View Only Permissions for Usage Type

|  |  |  |
| --- | --- | --- |
| **Edit Permissions for Usage Type** | | |
| **Settings** | **Screen** | **Effect** |
|  Edit Guidelines   View Guidelines  Alternate setting:   Edit Guidelines   View Guidelines | Manage Guidelines screen | Edit button enabled. |
| Edit Guideline screen | Screen data editable. |
| Search Policies screen,  Search Results pane | Edit button enabled. |
|  Edit Templates   View Templates  Alternate setting:   Edit Templates   View Templates | Manage Templates screen | The following buttons will be enabled: New Template, Copy, New Version, Delete Template, Edit, and Save. |
| New Template wizard | Usage types for which user has edit privileges will appear in the dropdown list. |

Table 12: Edit Permissions for Usage Type

| **Disabling Usage Type Permissions (No Access)** | | |
| --- | --- | --- |
| **Settings** | **Screen** | **Effect** |
|  View Guidelines   Edit Guidelines | Manage Guidelines screen | No Load button or Phase Group tabs for that usage type. |
| Search Policies screen,  Template Selection pane | Phase folder for usage type does not appear. |
| Search Policies screen,  Search Results pane | Guidelines for usage type do not appear in search results. |
|  View Templates   Edit Templates | Edit Guideline screen | The Edit Template button is disabled. |
| Manage Templates screen | The Edit and View buttons are disabled. |

Table 13: No Privileges for Usage Type

## Manage Process

The Manage Process module is used to control business process flow of the rule engine.

### Concepts

In this contexta taskrefers to a particular set of application code (i.e. business rules) which performs a specific set of work towards a goal, for example pricing a loan. Each rule within the task will have a condition (described below) containing the name of the task. The rules within a task will have no direct reference to phases only to tasks. Tasks are defined to be a guideline type (or usage type).

A phase is an rule engine object which controls the activation of a particular task. A network of phases implements the flow of control in the application. More than one phase can activate the same task.

Normally phases are activated sequentially with one ending before the next begins. However, in some cases phases can be active concurrently. A phase can have one or more sub-phases. Sub-phases can start as soon as the parent phase starts and the parent phase is not complete until its sub-phases have completed. The single root phase represents the entire set of phases in a business process. All the other phases will be sub-phases of the root phase.

### Managing Phases

Phases are maintained in the Manage Phases screen (Figure 83).

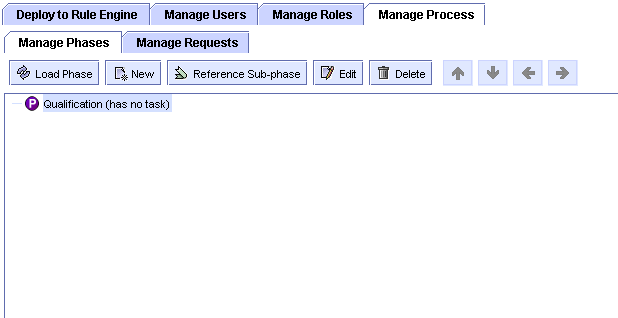


Figure 83: Manage Phase Screen

#### Create New Phase

1. Login to PowerEditor as a user that has the Manage Phase privilege.
2. Click the Admin tab, Manage Process tab, and then Manage Phases tab.
3. Click the Load Phase button to load existing phases.
4. Select the parent Phase, if desired.
5. Click Newto display the New Phase dialog appears (Figure 84).
6. Fill in the details as follows:

* Name - name of the phase used as the identifier in the generated rule engine file
* Display Name - display name
* Parent Phase - display name of the parent of the new phase. Cannot be changed. Note: the parent of a phase can be modified by using the left and right arrow buttons, once a phase has been created.
* Prerequisite Phase - a list of prerequisite phases. Can be empty.
* Disjunctive Prerequisites - If checked, only one of the prerequisite phases needs to be completed before this phase can run. If unchecked, the entire prerequisite phases in the prerequisites list must be completed.
* Task - Select the task (the same as usage) to perform for this phase. Can be empty.

1. Click Accept. The newly create phase should now be visible in the Phase Tree pane (just below the buttons).

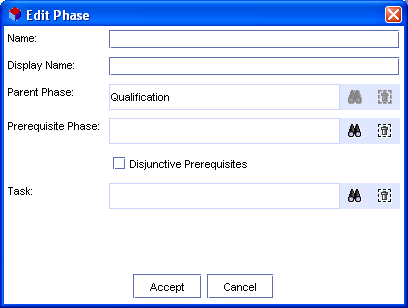


Figure 84: Edit Phase Dialog

#### Edit Phase

1. In the Phase Management screen, select the Phase to edit
2. Click the Edit button
3. Modify any of the details. See Step 6 of Section 10.7.2.1 for details.
4. Click the Accept button.

#### Move Phase

Phases can be moved using the arrow buttons as follows:

1. The Left button “outdents” the selected phase, making it a child of its grand-parent phase (the parent phase of its parent phase).
2. The Right button “indents” the selected phase, making it a child of its sibling that appears just before the selected phase
3. The Up button moves the selected phase up in the tree, allowing reordering of sibling phases. This does NOT modify the phase in any way. It simply rearranges the sibling phases so that the Left and Right button produces desired effects.
4. The Down button moves the selected phase down in the tree, allowing reordering of sibling phases. This does NOT modify the phase in any way. It simply rearranges the sibling phases so that the Left and Right button produces desired effects.

#### Delete Phase

To delete an existing phase,

1. Select the phase to delete. If the Delete button becomes inactive (grayed-out), you cannot delete the selected phase. You can only delete a phase that has no child phase.
2. Click the Delete button.
3. If the selected phase is a root phase and is used by at least one request object, you will be prompted as such and to confirm the deletion. If not, you will be prompted to confirm the deletion. Click Yes to delete, or No to cancel the operation.

#### Refresh Phases

If at any time, you want to refresh the phase tree from the server, click the Load Phase button.

### Managing Requests

Requests are maintained in the Manage Request screen, depicted in Figure 85.

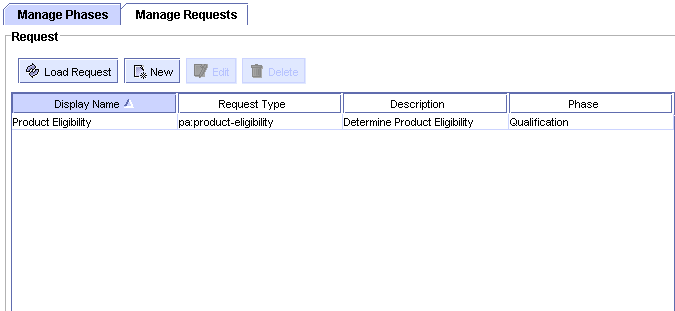


Figure 85: Manage Request Screen

#### Create New Request Object

1. Login to PowerEditor as a user that has the Manage Request Type privilege.
2. Click Admin tab, Manage Process tab, and then Manage Requests tab.
3. To load existing request objects, click the Load Request button.
4. Click the New button to display the New Request dialog (Figure 86).
5. Fill the details as follows:

* Name - the name of the request object, used as the ARTScript object instance name; must be a valid AE symbol
* Display Name - display name
* Description - a text description of the request object
* Request Type - the request type; must be a valid AE symbol
* Init-Function - the init-function; must be a valid AE function name
* Purpose - the purpose; must be a valid AE symbol
* Phase - select the root-level phase (a phase that does not have a parent) for the request object.

Note that request type, init-function, and purpose must be a symbol or a function recognized by the MMS engine. Consult the MMS engine documentation or code for valid values for these fields.

1. Click the Accept button. The newly created request object should now be visible in the Request object table.

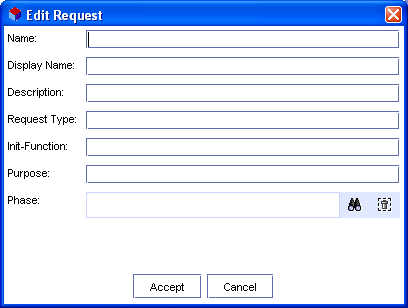


Figure 86: Edit Request Dialog

#### Edit Request Object

1. Click the request object to edit.
2. Click the Edit button.
3. Modified the fields; see Step 5 of Section 10.7.3.1 for details on the fields.
4. Click the Accept button.

#### Delete Request Object

To delete an existing request object,

1. Select the request object to delete.
2. Click the Delete button.
3. You will be prompted to confirm the deletion. Click Yes to delete, or No to cancel the operation.

#### Refresh Phases

If at any time, you want to refresh the request objects from the server, click the Load Request button.Advanced Knowledge Base Concepts

The following section includes information that is particularly applicable to PowerEditor users who will be creating new templates, and maintaining the domain model. Many of these concepts were introduced in *Section 1 Introduction*.

## Domain Model

The Domain model provides a definition of the object model used by the rule engine. The model defines the hierarchy of classes and their associated attributes and is typically used to model both input data and derived data. The maintenance of the domain model is achieved via the Domain Editor tool, which is external to the PowerEditor itself.

### Classes

At the highest level the domain model consists of classes. A class is used for containing a set of related attributes. For example, the class “Borrower” would be used to contain information (e.g. name, address, credit score, etc) about a specific borrower. Classes can also have multiplicity. For example this means that there could be multiple borrowers (e.g. primary, secondary) resulting in more than one instantiation of the “Borrower” class at runtime. Often times, rules may need to be written to match a specific instance of the class “Borrower”.

A class can also have child classes. This can be represented as a class contained within a class. The class in which the child class belongs is referred to as the parent class. A child class is typically defined for the purpose of containing some subset of data related to the higher-level parent class. For example, a class named “Borrower” may contain child classes called “Credit-Report” and “Residence History”. This provides a means of logically grouping like data within the class itself. There is no limit to the hierarchy of classes in the domain or to the number of child classes one can have.

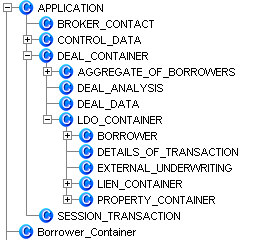


Figure 87: Example Parent Class/ Child Class Hierarchy

The symbol is used to represent classes throughout PowerEditor.



### Attributes

An attribute defines a specific piece of data contained within a class. Examples of attributes could include Loan-Amount, Credit-Score, etc. An attribute is always accessed through the class in which it is a member. For the purpose of illustration the previously mentioned attributes can be represented as LOAN.Loan-Amount and BORROWER.Credit-Score. The term to the right of the period (“.”) indicates the attribute name. The term to the left of the period in all CAPS represents the name of the class in which the attribute belongs. Joined together, they provide a unique identifier for the attribute within the knowledge base.

Attributes are defined with properties, such as name and data type. As an example, an attribute named “Term” representing the number of months in which a loan is amortized would be defined with a data type of Integer.

A complete list of the data types for attributes supported by PowerEditor include:

**Data Types**

* *Boolean* - Used to indicate true or false.
* *Code*. Alphanumeric value used as a unique identifier. Typically associated with legacy systems.
* *Currency* - Used to represent monetary amounts (e.g. $10.00).
* *Date* - Used for dates. Formats include Julian or Gregorian date formats.
* *Integer* - Positive or negative numeric values (e.g. –1, 0, 1).
* *Percent* - Used to represent non-monetary floating-point numbers.
* *String* - A group of characters surrounded by double quotes (e.g. “String”). A string can also have embedded spaces.
* *Symbol* - A specialized term used by ART\*Enterprise. A symbol is a group of characters not surrounded by quotes with no embedded spaces (e.g. SYMBOL). Specifically, a symbol can be comprised of any of these characters:

ABCDEFGHIJKLMNOPQRSTUVWXYZ  
abcdefghijklmnopqrstuvwxyz  
0123456789  
!@$^\*\_-+={}[:<>.?/

The figure below illustrates a set of attributes that have been defined for a class called “PROPERTY\_ADDRESS”.

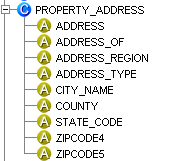


Figure 88: “PROPERTY\_ADDRESS” Class

The symbol is used to represent attributes throughout PowerEditor.



## Rules

A rule is a means of systematically expressing all or part of a specific business policy or procedure. A rule contains one or more conditions along with an action(s). The set of condition(s) are sometimes referred to as the **Left-Hand-Side** (**LHS**) of the rule while the action(s) is sometimes referred to as the **Right-Hand-Side** (**RHS**). The logic of a rule is such that whenever the conditions on the Left-Hand-Side are true then the actions on the Right-Hand-Side are performed. This can also be thought of as an “IF – THEN” logical statement where “IF” represents the conditions (LHS) and “THEN” refers to the action(s) (RHS) to be performed:

IF

Occupancy = Primary

Loan-Amount >= 100

Loan-Amount <= 400,000

LTV <= 95

THEN

Qualify-the-Loan

The figure above represents a LTV Guideline Rule in the mortgage domain. In the example, whenever all of the conditions defined in the “IF” section are true then the associated action is performed which in this case is to qualify the loan application.

### Conditions

Each condition within a rule represents an evaluation of a specific attribute in the domain model. Examples of conditions could include Loan-Amount > 50000, Loan-Purpose = Purchase, Credit-Score <= 600, etc. Conditional statement logic can take several forms depending upon the attribute being evaluated. The following examples represent the general types of conditional logic that can be performed and are provided in this section for demonstration purposes only.

* *Equality/Inequality*. This simply checks to determine if the value of the attribute is equal to the guideline value.

Property-Type = “Single Family”

Loan-Purpose != Purchase

* *Membership*. This checks to determine if the value of the attribute is equal to any one of multiple values stated in the guideline. There is no limit to the number of comparison values for the guideline.

Property-Type IN (“Single Family”, “Duplex”, “Triplex”)

* *Numeric Range*. This is for comparing numeric attributes to guidelines. Numeric checks can be performed against a single value as with the first example or range of minimum and maximum values as with the second example. The list of allowable operations includes <,>,<=, >= and between.

Loan-Amount >= 50000

Loan-Amount BETWEEN (50000, 100000)

Table 14 summarizes the relationship between data attribute types, allowable operators and comparison values.

|  |  |  |
| --- | --- | --- |
| Attribute Type | Supported Operators | Valid Comparison Values |
| Boolean | ==, != | true, false |
| Code | ==, !=, in | Any valid symbol |
| Currency | ==, !=, <, >, <=, >=, between, in | Any number |
| Date | ==, !=, <, >, <=, >=, between | Any valid date |
| Integer | ==, !=, <, >, <=, >=, between, in | Any integer |
| Percent | ==, !=, <, >, <=, >=, between, in | Any number |
| String | ==, !=, in | Any string |
| Symbol | ==, !=, in | Any valid symbol |

Table 14: Conditional Operators and Compare Values by Attribute Data Type

### Logical Operators

Logical operators can be applied to one or more conditions within a rule. Their purpose is to determine how the conditions are to be evaluated in order to establish whether or not the Left-Hand-Side of the rule has been satisfied. The logical operators and their evaluation method are described below:

* **AND**. All conditions defined under the **AND** logical operator must be true in order for the rule to be satisfied. If no logical operator is present within the rule definition, then AND is the implied operator.



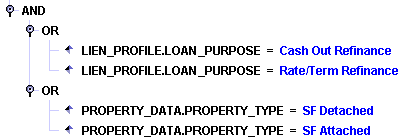
* **OR**. At least one of the conditions under the **OR** logical operator must be true in order for the rule to be satisfied.



* **NOT**. This logical operator negates the condition(s) it is applied to. For example, a test for equality becomes a test or inequality. The NOT logical operator applied below results in the rule evaluating that the result of the AND operations is not true. This would mean that either one or none of the conditions would be true but not both.



Logical Operators can also be nested to achieve more complicated logic. For instance, in the illustration below, an AND operator is used to apply logic to the results of two OR clauses. As a result, both of the OR clauses must evaluate to be true in order for the rule’s Left-Hand-Side to be satisfied.



### Actions

If the conditions (Left-Hand-Side) of the rule are satisfied then the action (Right-Hand-Side) associated with the rule is performed. See Section 2.6 for more details.

## Messages

A message is a parameterized text explanation used to provide a description to end users of a rule or guideline that has executed. The message will typically include the identity/purpose of the rule or guideline and the corresponding action taken by the rule engine. For example, in the mortgage domain, a price adjustment guideline will describe an adjustment that has been made to the rate or price of the loan; and a qualification guideline will describe why a loan is not eligible for a particular product

Messages can be made more specific to the request through the use of Message Arguments. Message Arguments are implemented as placeholders within a message that can be dynamically populated during the processing of the request.

There are two types of message arguments that can be used: Guideline Message Arguments and Input Message Arguments. Pursuant to their name Guideline Message Arguments only apply to guidelines. They get their value from the guideline rule that executes. It’s the value used by the rule to compare against the input data. For example, in a rule comparing Loan Amount > 50000, the Guideline Argument would be 50000. Input Message Arguments are obtained or derived from the input request data. The Input Argument would be the actual Loan Amount being requested by the borrower.

The following message provides an example of a complete guideline message with arguments:

“DTI of |deal\_calc\_a.dti| exceeds limit of %column 4% for doc type %column 2%, credit grade %column 1% and LTV of |lien\_calc.LTV|”

In this example, the message describes the reason that a product is ineligible. The use of message arguments allows the message be very specific as to why it is not eligible. Arguments surrounded by the percent sign (‘%’) are Guideline Message Arguments while those surrounded by the pipe symbol ‘|’ are Input Message Arguments.

# troubleshooting

## Read-Only Mode

If Edit buttons on all screens is disabled or not visible, the PowerEditor you logged into is running in read-only mode. This means that the PowerEditor is configured as a historical instance with date-based knowledge base (KB) filtering. Typically, there is another PowerEditor instance that allows editing. See PowerEditor administrator for more information on how to access such instance.

## Message Validation Errors

### Missing “%” Symbol

This error occurs when a “%” symbol is missing from a Guideline Message Argument placeholder. There should be one in front and back of the argument value.

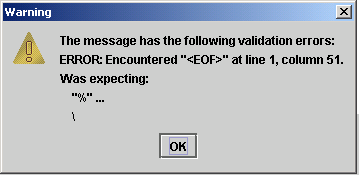


Figure 89: Missing % symbol Error Message

### Missing “|” Symbol

This error occurs when a “|” symbol is missing from a Input Message Argument placeholder. There should be one in front and back of the argument value.

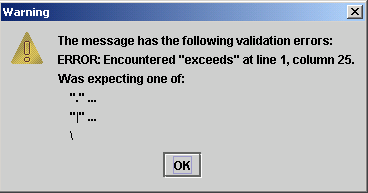


Figure 90: Missing | symbol Error Message

### Bad Value

This error can occur for a variety of reasons. Most commonly it is associated with a malformed argument placeholder.

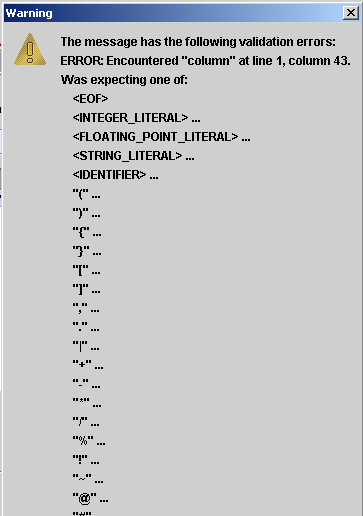


Figure 91: Bad Value Error Message

### Class Not Found

This error occurs when an incorrect class name is used with an Input Message Argument.

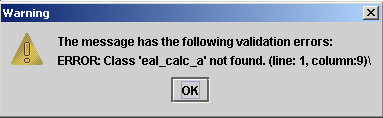


Figure 92: Class not Found Error Message

### Attribute Not Found

This error occurs when an incorrect attribute name is used with an Input Message Argument.

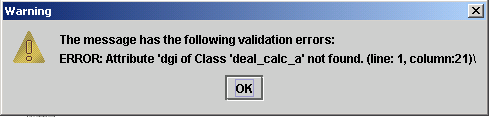


Figure 93: Attribute not Found Error Message

### Invalid Column Number

This error occurs whenever an invalid column number is used for a Guideline Message Argument.

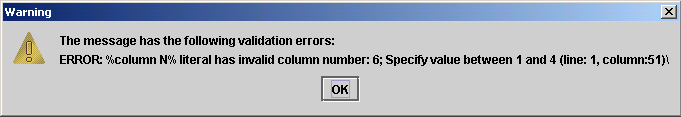


Figure 94: Invalid Column Error Message

## Deploy Script Issues

If the deployment error message indicates that PE failed to execute the deploy script, review the PowerEditor server log for details. As of PE 5.8.0, the output from the deploy script will be logged in the server log.

## Audit Trail

PowerEditor provides audit trail that captures changes to various aspects of knowledge base.

### Captured Changes

The following is a list of changes captures in the PowerEditor audit trail.

* Server Startup and Shutdown
* PowerEditor keeps audit trail of deployment history
* Capture the changes in the policy grids, including inserted rows and deleted rows in Audit PowerEditor captures all changes made to individual cells in a policy grid.

### View Audit Reports

PowerEditor can display audit reports that can be viewed in the Reports tab. This requires building a customer report. Refer to *PowerEditor Customer Report Guide* for details.