

**Custom Reports Guide**

**PowerEditor Version 5.10**

Last Revised: October 25, 2007

Re-Released: September 07,2018

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

*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.

Table of Contents

[1. Crystal Reports™ Tutorial: Single Template 4](#_Toc181088311)

[1.1 Install Crystal Reports XI Developer Edition 4](#_Toc181088312)

[1.1.1 Installation Notes 5](#_Toc181088313)

[1.2 Create Report: Specify Data Sources 7](#_Toc181088314)

[1.2.1 Create New Report 7](#_Toc181088315)

[1.2.2 Define Data Source 7](#_Toc181088316)

[1.3 Create Report: Specify Data Elements 13](#_Toc181088317)

[1.4 Create Report: Report Layout 19](#_Toc181088318)

[1.4.1 Blank Layout Screen 19](#_Toc181088319)

[1.4.2 Step 1: Add Group For Template ID and Activation ID 20](#_Toc181088320)

[1.4.3 Step 2: Add Grid Data 23](#_Toc181088321)

[1.4.4 Step 3: Add Message Text 24](#_Toc181088322)

[2. Crystal Reports™ Tutorial: Multiple Templates 26](#_Toc181088323)

[2.1 Multiple Template Reports: Overview 26](#_Toc181088324)

[2.2 Create Report: Specify Data Sources 27](#_Toc181088325)

[2.2.1 Define Data Source 27](#_Toc181088326)

[2.3 Create Report: Specify Data Elements 30](#_Toc181088327)

[2.4 Create Report: Current Rate Adjusters 33](#_Toc181088328)

[2.4.1 Step 1: Add Group For Template and Activation 33](#_Toc181088329)

[2.4.2 Step 2: Add Grid Data 34](#_Toc181088330)

[2.4.3 Step 3: Add Filter Formula 35](#_Toc181088331)

[3. Crystal Reports Tutorial: Audit REport 36](#_Toc181088332)

[3.1 Audit Report Overview 36](#_Toc181088333)

[4. Crystal Reports™ Interface to PowerEditor: Technical Reference 38](#_Toc181088334)

[4.1 Crystal Report Version Information 38](#_Toc181088335)

[4.2 Making Report Available to PowerEditor GUI 38](#_Toc181088336)

[4.3 Refreshing PowerEditor Data Within Crystal Reports Developer 38](#_Toc181088337)

[4.4 URL Specifications 39](#_Toc181088338)

[4.4.1 XML Data Source Data URL 39](#_Toc181088339)

[4.4.2 URL for XML Schema URL 40](#_Toc181088340)

[4.5 Parameter Arguments 41](#_Toc181088341)

[4.5.1 Parameter Arguments: All Report Types (Except Entity Report) 41](#_Toc181088342)

[4.5.2 Parameter Arguments: Guideline Reports 43](#_Toc181088343)

[4.5.3 Parameter Arguments: Audit Reports 43](#_Toc181088344)

[4.5.4 Parameter Arguments: Entity Reports 44](#_Toc181088345)

[4.6 Database Tables and Fields 45](#_Toc181088346)

# Crystal Reports™ Tutorial: Single Template

Custom reports using data stored in the PowerEditor can be developed using Business Object’s Crystal Reports XI. Specifically, PowerEditor provides an application interface that can be used from Crystal Reports. This document provides two brief tutorials for creating custom reports using PowerEditor data. The tutorials are followed by a technical reference that summarizes the technical details that are buried in the tutorials.

Crystal Reports XI Release 2 Developer Edition is required for building PowerEditor reports. PowerEditor Version 4.4.0 has been tested with Crystal Reports Version 11.0.0.2002.

The two tutorials walk through the creation of two types of reports. The first is a single template report (or *template-specifi*c report), which provides very explicit control over the specific columns that appear in the template report. The second tutorial is for a multiple template report (or *generic-template*), which reports across multiple templates, thereby providing less control over template-specific column manipulation. The second tutorial also shows how to limit the activations reported to ones that are currently active.

Note that all figures in these tutorials are from the Crystal Reports XI Developer Edition, unless otherwise noted. Do not confuse these screens with PowerEditor figures. Let’s get started.

## Install Crystal Reports XI Developer Edition

Crystal Reports XI Release 2 Developer Edition is required for building PowerEditor reports. Specifically, PowerEditor Version 4.4.0 has been tested with Crystal Reports Version 11.0.0.2002. As of the writing of this document, a trial version of Crystal Reports XI can be obtained at [www.businessobjects.com](http://www.businessobjects.com). Make sure to download the Release 2, as is shown below in *Figure 1.*

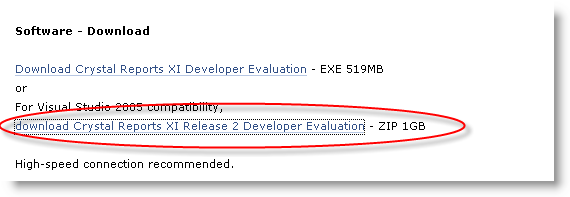


Figure 1: Business Objects Website Download

⮱ Note: This is a 1 gigabyte file. Yes, that’s right – the installer alone is a *Gigabyte*.

### Installation Notes

Two known PE/Crystal Reports installation issues have been identified, the problems and their workarounds are documented in the PowerEditor 4.5 *Release Notes,* and repeated here for ease of reference.

1. **Issue:** Crystal Reports LIBEAY32.DLL Incompatibility

If you have installed Crystal Reports XI and are running a version of ART*Enterprise* earlier than 10.1.1, you may run into plugin warnings such as:

WARNING: The following plugins failed to load:

Plugin http.aex (failed)

File: C:/mindbox/ae/10.0.4/dist/plugins\http.aex

Failed because: Unknown system error...(more)

Followed later by:

Loading from file "command.art"

WARNING: While compiling the DEFINE-METHOD CMD:PING, calling the function NET:LOCAL-HOSTNAME before it is defined...(more)

      ...ending with:

ERROR: NET:LOCAL-HOSTNAME is not the name of a function.

ART*Enterprise* and Crystal Reports XI need to load different versions of libeay32.dll.  Crystal Reports XI installs a version of libeay32.dll in the C:\windows\system32 directory.  When the ART*Enterprise* engine is started the C:\windows\system32 version is loaded first because C:\windows\system32 is included in the PATH environment variable.

**Work-Around:**

Set up your system so both ART*Enterprise* and Crystal Reports XI can use the correct shared libraries:

1. Copy the libeay32.dll from the C:\Mindbox\ae\10.0.4\dist\plugins directory into the C:\Mindbox\ae\10.0.4\dist\bin directory.
2. Create an empty file (0 bytes) in the C:\Mindbox\ae\10.0.4\dist\bin directory called aeserver.exe.local.
3. **Issue:** MSVC\*.DLL Incompatibility

On Windows Server 2003, once the LIBEAY32.DLL issue has been resolved, you may notice that the ODBC Data Source Administrator no longer allows you to create new DSNs, and when running the ART*Enterprise* engine you get errors such as:

ERROR: Terminal Error: Unable to connect to database DSN=<DSN>;uid=<UID>;pwd=<PWD>;srvr=<SRVR>

ARTEnterprise was at the top-level in function "DB::CONNECT-TO-DATABASE",

 when the error occurred.

Enter :H for help with the debugger.

Break>

**Work-Around:**

These MSVC\*.DLL libraries are needed if user-defined C++ plugins need to be built, or if ART*Enterprise* needs to be embedded into another C++ application.  This work-around assumes you will not be using Visual Studio C++ 6.0.

1. Shut down the ART*Enterprise* engine.
2. CD to the ART*Enterprise* dist\bin directory.
3. Rename the libraries:

MSVCIRT.DLL

MSVCRT.DLL

To:

MSVCIRT.DLL.OLD

MSVCRT.DLL.OLD

1. If you are not using the latest version of MDAC, download it from Microsoft and install it.

<http://www.microsoft.com/downloads/details.aspx?DisplayLang=en&FamilyID=6c050fe3-c795-4b7d-b037-185d0506396c>

1. Your original DSNs may no longer work.  If this is the case, recreate them in the ODBC Data Source Administrator (accessed via *Control Panel / Administrative Tools / Data Sources (ODBC)*).
2. Start the ART*Enterprise* engine.

## Create Report: Specify Data Sources

Once installed, start the Crystal Reports Developer application.

### Create New Report

A report that uses PowerEditor data can be created with one of the following two commands:

* Standard Report Wizard
* Blank Report

This document focuses on the Blank Report command.

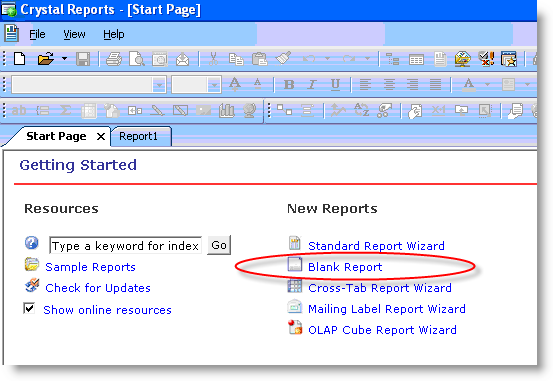


Figure 2: Crystal Reports Start Page

### Define Data Source

After selecting Blank Report from the Getting Started screen, you need to select all the data sources for your report. The interface to the PowerEditor database will be one of these data sources. You might have other data sources as well. For example, a Rate Sheet Report might interface with a base pricing database as well the PowerEditor interface. This document describes how to interface with the PowerEditor data source.

The PowerEditor reporting interface will be accessed via a URL based XML data source. To specify the PowerEditor as a data source, select Create New Connection, then the XML folder, from the Database Expert dialog, as shown below.

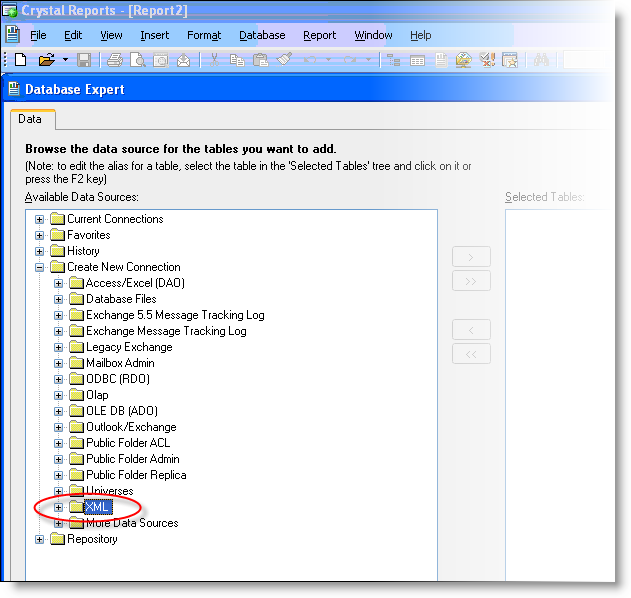


Figure 3: Database Expert Dialog

After selecting XML as the data source type, you will be presented with a series of five dialogs. In this series of dialogs, you will specify the URL that points to the PowerEditor data, as well as the URL which points to the XML schema. This series of five dialogs is presented below.

#### URL for XML Data

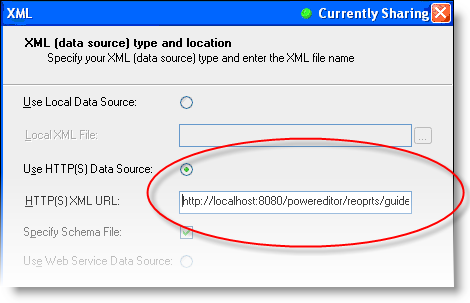


Figure 4: Dialog 1 in XML Specification

In the first dialog, select the radio button labeled “Use HTTP(S) Data Source”. Then, you will need to type in the URL for the data you want. An example URL is as follows:

http://localhost:8080/powereditor/reports/specific-template.jsp?usage=MMS-Create-Stipulations

The syntax of this URL is as follows. The 3 highlighted pieces of the URL are options that are described below.

http://*WebserverPort*/powereditor/reports/specific-template.jsp?*ParamName*=*ParamArg(s)*

*WebserverPort* This is the location of the web-server that hosts the PowerEditor. This portion of the URL is exactly the same as the URL that you use to launch the PowerEditor. Examples include localhost:8080 or 172.168.1.1:8080

*ParamName* This specifies the group of PowerEditor data that you will be drawing from in the report. The group of data that you specify here will probably be a superset of the data that actually gets included in the report. You will have a chance to refine the set of data later in the report specification process. The following are some of the valid values for *ParamName*. For a full list of valid values, see the technical specifications.

* usage – Specifies that you would like to report on data from the PowerEditor templates within the specified usage-type(s).
* templateid – Specifies the ID(s) of the template(s) that you would like to include in your report.
* template – Specifies the name(s) of the template(s) that you would like to include in your report.

*Examples of other parameter names include* context-elements, date, status *and more! These parameters help filter the set of guidelines included in the report. See the technical specifications for details.*

*ParamArg(s)* A comma delimited list of arguments to the given parameter.

* usage - The argument(s) to the usage parameter must be usage-type names as defined in your PowerEditor configuration (not to be confused with the usage-type display names). Typical examples include Pricing-Adjustments and MMS-Create-Stipulations. You would use this parameter if you intend your report to contain more than one template-specific report.[[1]](#footnote-1)
* templateid – The argument(s) to the templateid parameter must be numerical template IDs, e.g. 31024.
* template – The argument(s) to the template must be template names as they appear in the PowerEditor Manage Templates screen. These names can contain spaces.

Example URLs:

http://172.168.0.0:8080/powereditor/reports/specific-template.jsp?usage=MMS-Create-Stipulations, Pricing-Adjustments

http://amx-uat01:8080/powereditor/reports/specific-template.jsp?templateid=35225,35220

http://amx-uat01:8080/powereditor/reports/specific-template.jsp?templateid=31660

http://172.168.0.0:8080/powereditor/reports/specific-template.jsp?template=CORE:Units-Property,STIP-CREDIT: Max DTI

⮱ Hint: After you have entered your URL, you might want to copy it for use in Dialog 3 below. To copy the URL text, select the text, and type Ctrl-C.

⮱ Hint: You might even want to save this URL into a separate text file so you can copy and paste it when creating subsequent reports.

After entering a valid URL in Dialog 1 (Figure 4), select the *Next>* button.

Dialog 2 (Figure 5) can be ignored – no authentication is needed. Simply hit the *Next>* button.

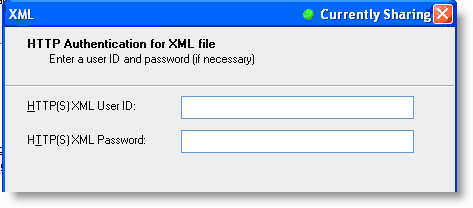


Figure 5: Dialog 2 in XML Specification

#### URL for XML Schema

The third dialog in the XML specification process prompts for the XML schema.

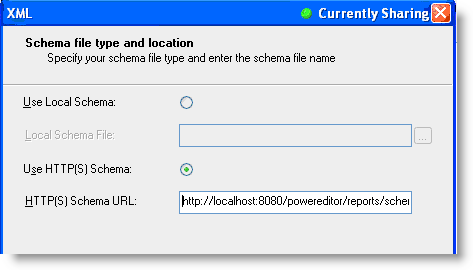


Figure 6: Dialog 3 in XML Specification

Make sure that the radio button labeled “Use HTTP(S) Schema” is selected. Enter the URL of the XML schema. The URL to be entered here is *exactly* the same as the one you entered in Dialog 1, with the following exception: replace /reports/ with /reports/schema/

Example URLs:

http://172.168.0.0:8080/powereditor/reports/schema/specific-template.jsp?usage=MMS-Create-Stipulations, Pricing-Adjustments

http://amx-uat01:8080/powereditor/reports/schema/specific-template.jsp?templateid=35225,35220

http://amx-uat01:8080/powereditor/reports/schema/specific-template.jsp?templateid=31660

http://172.168.0.0:8080/powereditor/reports/schema/specific-template.jsp?template=CORE:Units-Property,STIP-CREDIT: Max DTI

After entering a valid URL in Dialog 3 (Figure 6), select the *Next>* button. Alternately, you can just select the *Finish* button, thereby skipping Dialogs 4 and 5.

Dialog 4 (Figure 7) can be ignored – no authentication is needed. Simply hit the *Next>* button.

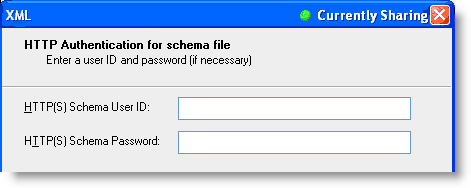


Figure 7: Dialog 4 in XML Specification

Dialog 5 can also be ignored – it simply verifies what you’ve already entered.

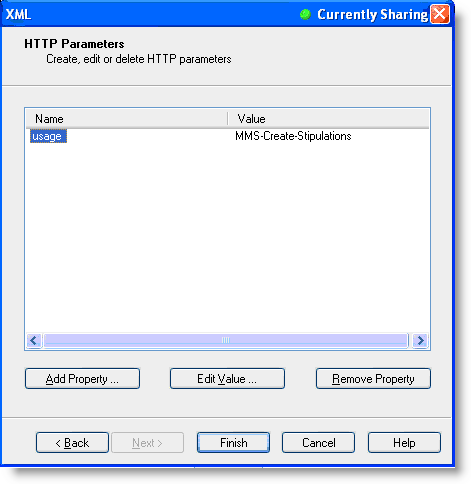


Figure 8: Dialog 5 in XML Specification

#### XML Specification Results

If the URLs were correctly entered, the system will pause for a bit at this point while Crystal Reports queries the PowerEditor server application. You will know that you have succeeded if you see the dialog presented in Figure 11.

If you receive the following error, the URL was entered incorrectly. When you click OK, Crystal Reports graciously takes you back to the dialog in which the error was generated.

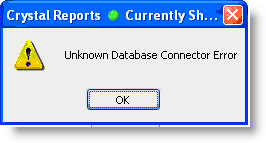


Figure 9: XML Specification Error

The following dialog indicates that there is a translation error in the template name. This is an internal PowerEditor error, and should be reported to your PowerEditor system administrator.

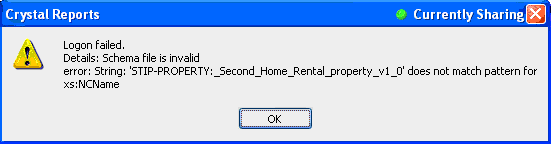


Figure 10: Internal PowerEditor Error

## Create Report: Specify Data Elements

When you successfully selected your data sources that will be included in the report, you are ready start selecting specific data elements from the data source. You will be in the Report layout tab, and will be presented with a couple more dialogs.

Many times when Crystal Reports accesses the PowerEditor database, it prompts for parameter values for the URL. Just select the parameter values that you specified in your URL (e.g. MMS-Create-Stipulations) and choose OK.

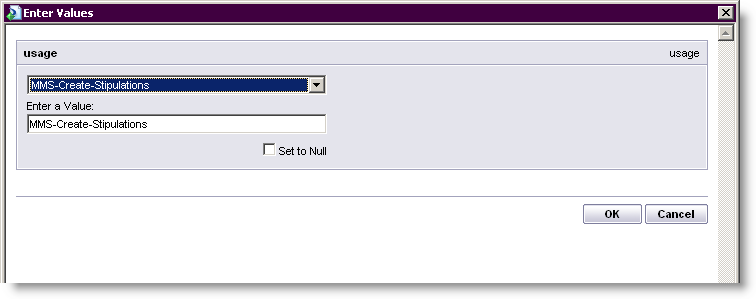


Figure 11: Parameter Values Dialog

You will then be presented with the Database Expert dialog that you’ve seen before. However, now the templates you selected will appear under the XML element. Expand the guideline-activations box to see your templates. A sample expanded list is shown in below in Figure 12.

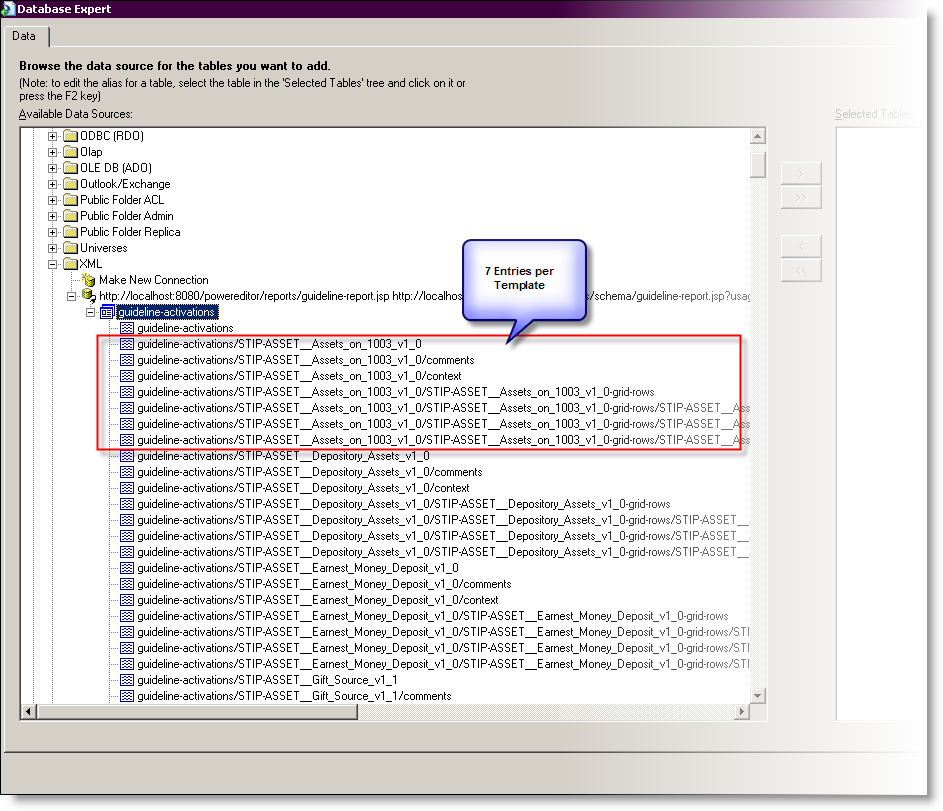


Figure 12: Database Expert Dialog After Data Source Selection

In this dialog, choose the data elements that you would like to make available in your report. Move selected data elements to the *Selected Tables* portion of the dialog by using the button.



⮱ Note: If your data source selections contain multiple templates, you should select the first item in the list: guideline activations. This is the *container element* that links together all other template elements.

The remainder of this section describes the contents of this listing. The following diagram shows a PowerEditor view of templates listed above. The PowerEditor template highlighted in the figure below is represented by the 7 lines highlighted in the figure above.

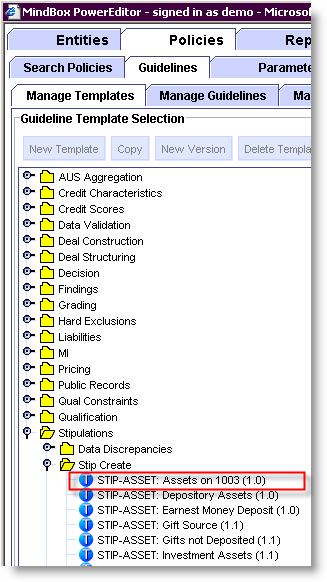


Figure 13: PowerEditor View of Templates

The names of the PowerEditor templates get somewhat digested in this Crystal Reports listing. In particular, special characters in the template name, such as colons, periods, and spaces, are replaced with underscores in the Crystal Reports listing. Furthermore, the template version name get appended to the template name. Therefore, a template in the PowerEditor that is named “STIP-ASSET: Assets on 1003” version “1.0” appears in Crystal Reports as STIP-ASSET\_\_Assets\_on\_1003\_v1\_0.

Furthermore, this listing includes 7 entries for each PowerEditor template. Each of the 7 entries is represents a Crystal Reports table, each of whichrepresents a structural component of PowerEditor guidelines that you might be interested in displaying. For example, in Template A, you might be interested in displaying the guideline grids, and for Template B, you might be interested in displaying a verbal representation of the guidelines rules. The following diagram shows an expanded listing of the 7 table names available for one template.

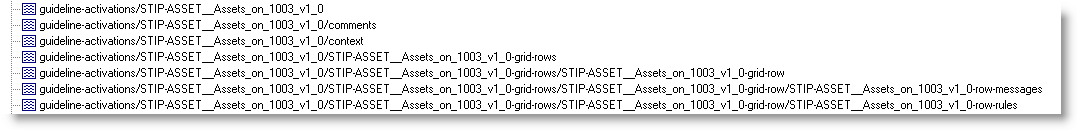


Figure 14: Table Entries for Template STIP-ASSET\_\_Assets\_on\_1003\_v1\_0

The semantics of these 7 table entries are as follows:

*<template>*  STIP-ASSET\_\_Assets\_on\_1003\_v1\_0

This contains template-level data, such as template ID and template name.

⮱ Note: If you intend to select any other entries for this template, you **must** select this entry as well, since this is the *container element* that links together all the other elements for this template.

*comments* STIP-ASSET\_\_Assets\_on\_1003\_v1\_0/comments

This table references the comments associated with each activation of the template.

*context* STIP-ASSET\_\_Assets\_on\_1003\_v1\_0/context

This table references the context associated with each activation of the template.

*grid-rows* STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-grid-rows

This references the data in the guideline grids that are associated with each activation of the template.

*grid-row* STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-grid-rows / STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-grid-row

This references the data in a single row of the guideline grid.

*row-messages* STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-grid-rows / STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-row-messages

This table references the rule message generated for a single row of the guideline grid.

*row-rules* STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-grid-rows / STIP-ASSET\_\_Assets\_on\_1003\_v1\_0-row-rules

This table references the textual representation of rule message associated with single row of the guideline grid.

The following PowerEditor screen display shows the data that is referenced by most of these 7 entries.

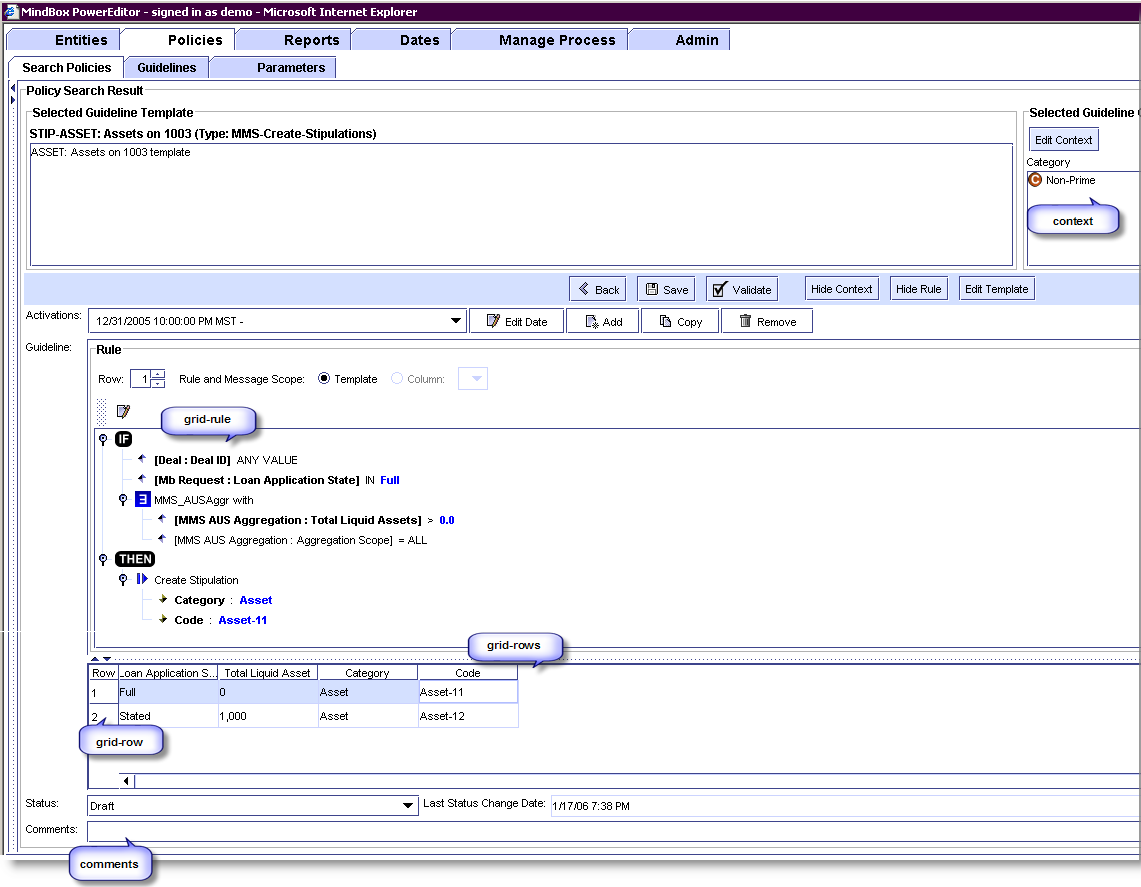


Figure 15: PowerEditor Display Showing Source of Some Template Entries

After you have selected the entries that you are interested in, you will be presented with a Link Selection dialog. This dialog will be used for merging multiple data sources. If the PowerEditor database is the only data source included in your report, you should never need to touch this dialog, and can simply select the *OK* button. However, if you are merging PowerEditor data with another data source (e.g. a base pricing database), you will probably need to do something here. That discussion is outside of the scope of this document.

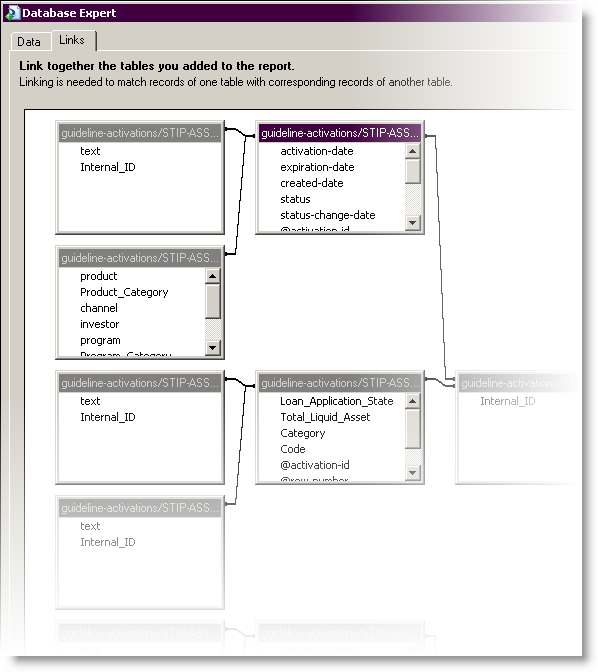


Figure 16: Database Expert: Link Selection

You will probably see the following dialog: don’t worry about it.

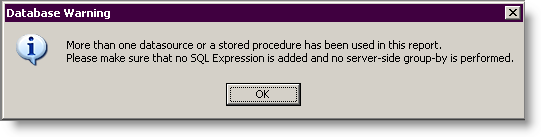


Figure 17: Database Expert: Warning Dialog

⮱ Note: After selecting OK here, it would be a good idea to save the work you have done so far. And maybe take a little coffee break.

## Create Report: Report Layout

Now you are ready to layout your report elements. This is about to start getting fun. Crystal Reports is a very robust report layout tool; this document presents the tip of the iceberg. This section will walk through the creation of a sample report using PowerEditor data. However, you will quickly find that you will need to refer to Crystal Report documentation.

This section of the document walks through an example of laying out a report consisting of grid data from a single template. Note that in this document, the word *template* refers to a PowerEditor template. This is not to be confused with a Crystal Reports template, a term that you might find as you peruse the Crystal Reports GUI.

After having completed the steps outline in the previous sections, you should be presented with a screen that is similar, but not quite the same as, the figure below. The subsequent notes describe how to get your screen to look more like this.

### Blank Layout Screen

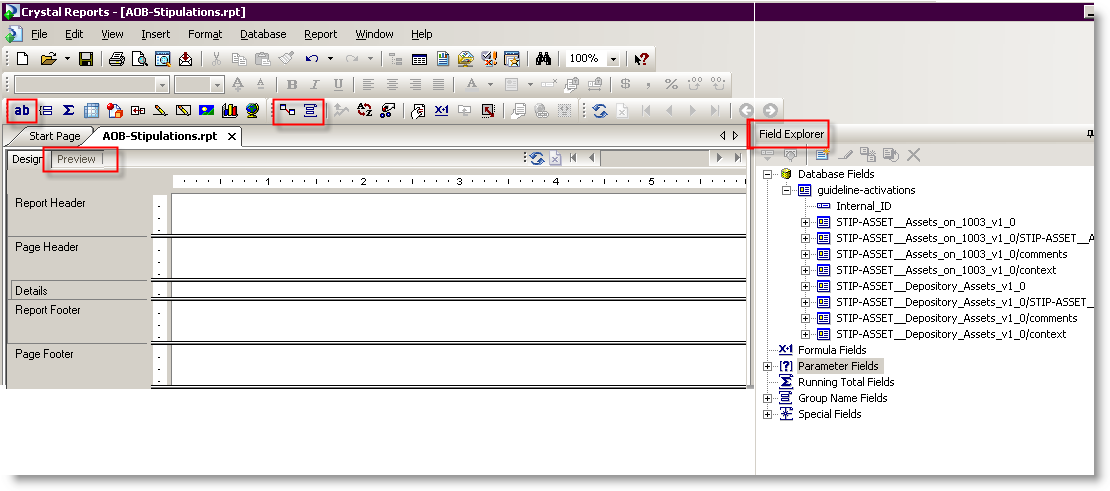


Figure 18: Initial Report Layout Screen

The following lists some elementary screen elements that will be useful as we proceed through the exercise of creating a report.

The Insert Text Object button is used to insert static text in the report.



The Database Expert button is used to add more database elements to the report. This takes you back to the dialog presented in *Figure 12: Database Expert Dialog After Data Source Selection*.



The Group Expert button allows you to order groups of items, like multiple activations for a single template. This is explained below.



The Field Explorer pane shows your selected data elements. See the note regarding this pane below.



The Refresh button refreshes data that was generated from a database query. Use this if the contents of your PowerEditor database has changed since you opened this report.



The Start Page tab takes you back to the page that allows you to create a new report (the page that is shown in *Figure 2: Crystal Reports Start Page.*)



The Preview tab allows you to toggle to a view that shows data being populated in your report. When Crystal Reports displays the Preview View, it queries the PowerEditor database for the actual data values. Toggling between the Preview View and the Design View is very useful during report development.



⮱ Note: If your Field Explorer Pane is not visible, select View->Field Explorer. Then expand the Database Fields and Guideline Activations in the Field Explorer hierarchy.



⮱ Note: If your Preview tab is not visible, select View->Preview Sample. Then select the design tab to get back to the view presented in *Figure 18: Initial Report Layout Screen.*

### Step 1: Add Group For Template ID and Activation ID

To list all the activations for a given template as sorted by activation date, the activations should be defined as part of a Crystal Reports *Group*. From the Crystal Reports documentation:

Use the Group command to sort your data and break it into groups. For example, you could group the information in a customer list into state or zip codes. When you choose the Group command, the Insert Group dialog box appears.

Choose the Group Expert icon . Selected the template-id as the primary sort for your group, then the activation-date as the secondary sort. Your dialog should appear as follows:

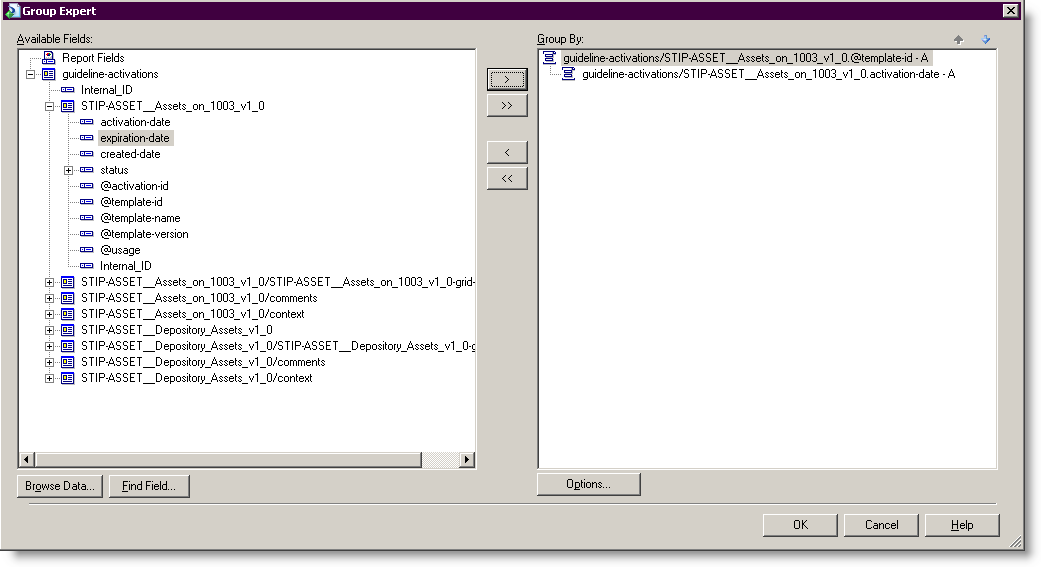


Figure 19: Select Template ID and Activation Date as Sort Values

After you’ve selected your template-id and activation date, the report screen should appear as below. Note that the figure below presents both the Design View and the Preview View.

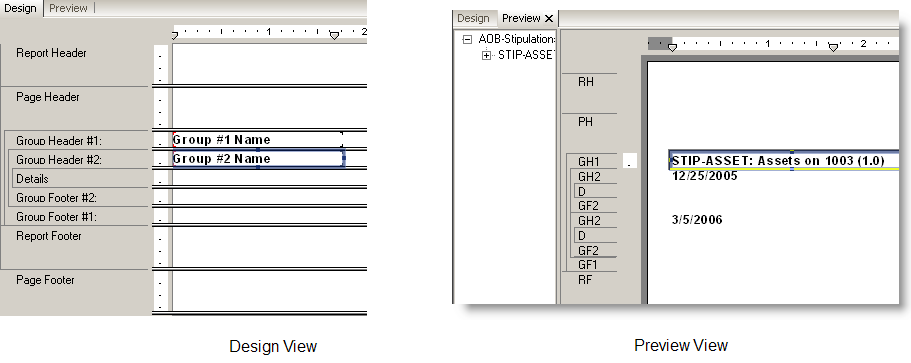


Figure 20: Add Group for Template 1

Try adding static text, changing fonts, and adding other data elements to the Group Heading fields. Note that the layout of fields can be manipulated in both the design view and the layout view.

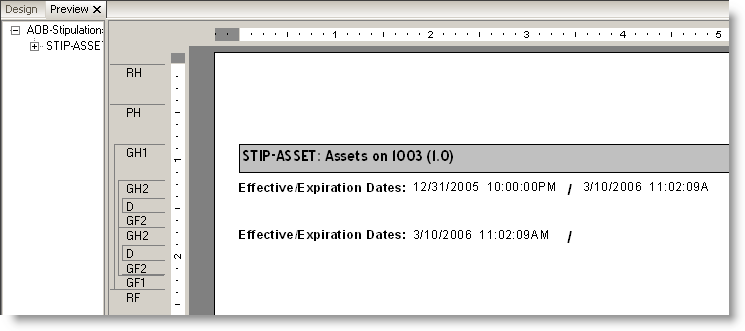
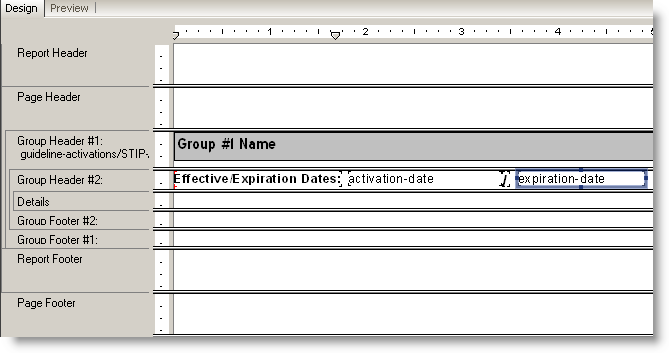


Figure 21: Adding Text -- Design and Preview Views

### Step 2: Add Grid Data

To add grid data, expand the grid-rows and grid-row database fields. You’ll see your PowerEditor columns listed, as shown in the right in the figure below. Click on these, and drag them to the Details section of the Design View. The column headings will appear by default in the Page Header section. Move column headings into the Group Header #2 section. Figure 22 shows the results of the actions.

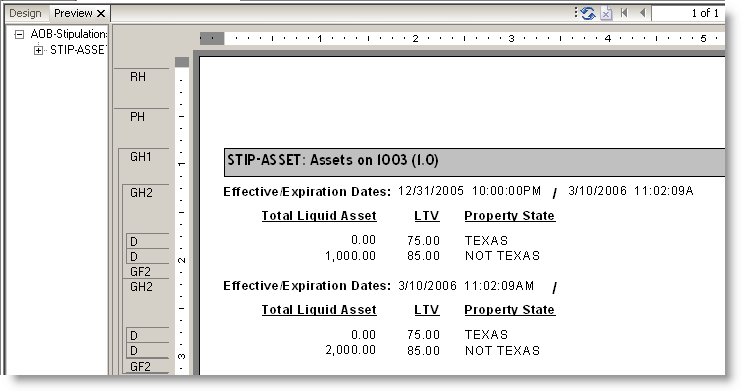
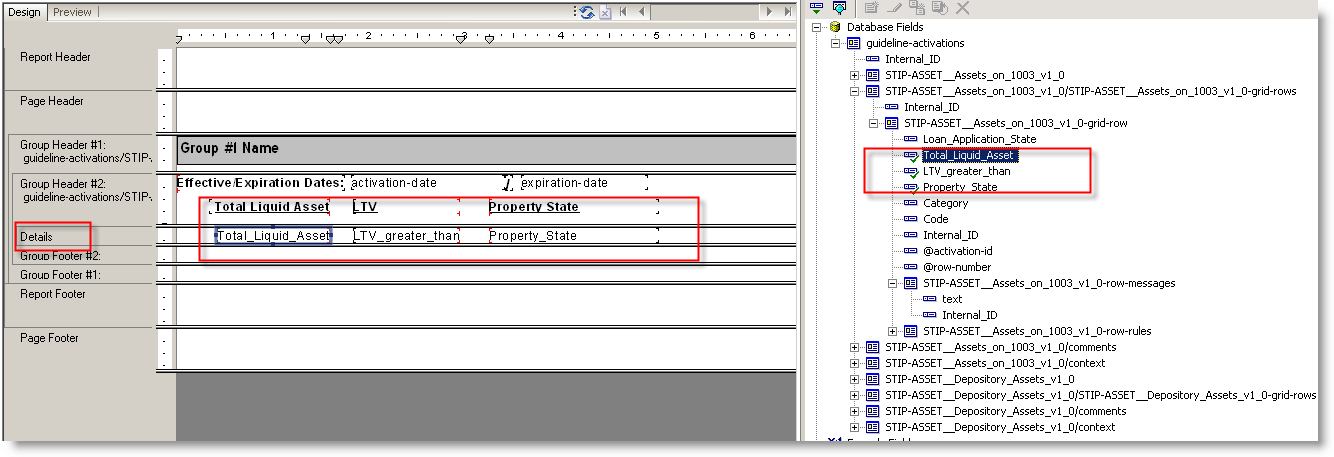


Figure 22: Adding Grid Data -- Design and Preview Views

### Step 3: Add Message Text

To add the message that will be generated for each row, select the row-messages database field, as shown in the figure below. Drag this into the Details section. Again, drag the header into the Group Header #2 section.

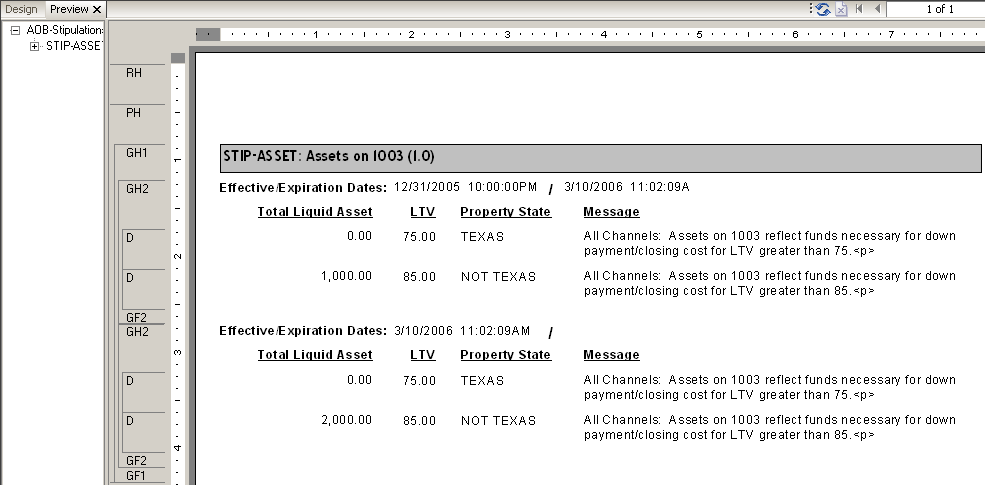
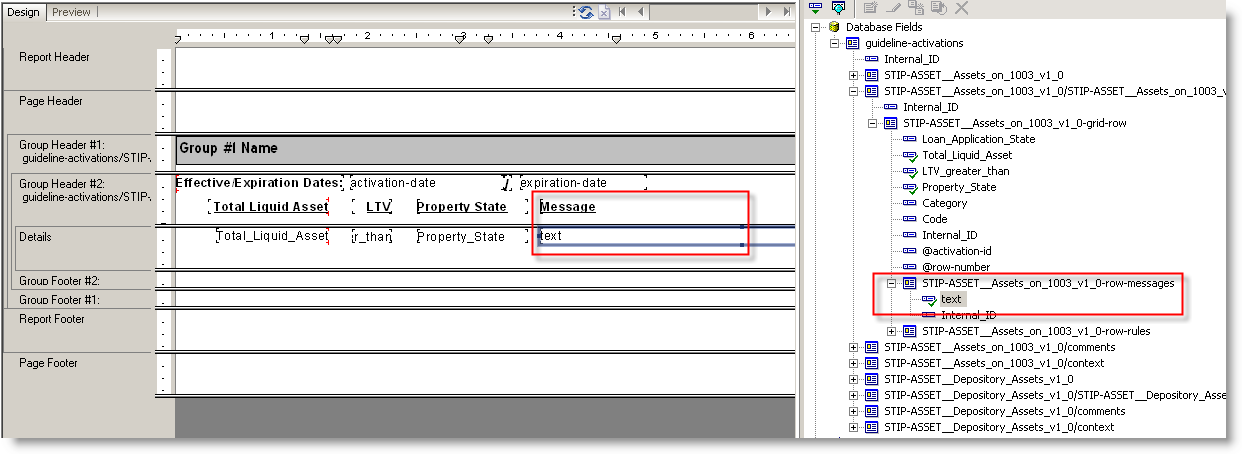


Figure 23: Adding Message Text -- Design and Preview Views

⮱ Hint: To get your message text to be automatically sized, click right on the text item in the details screen, select Format Field, the click on the Can Grow checkbox.

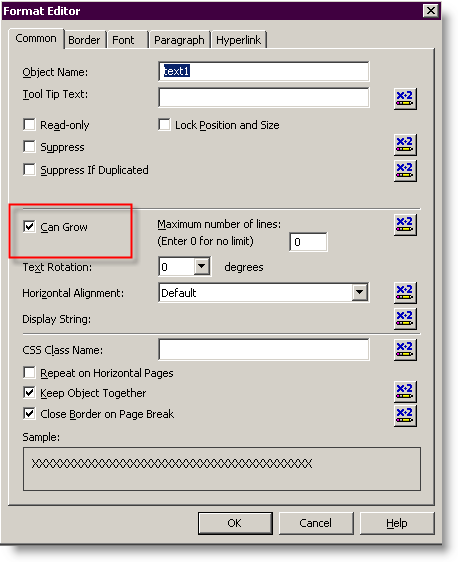
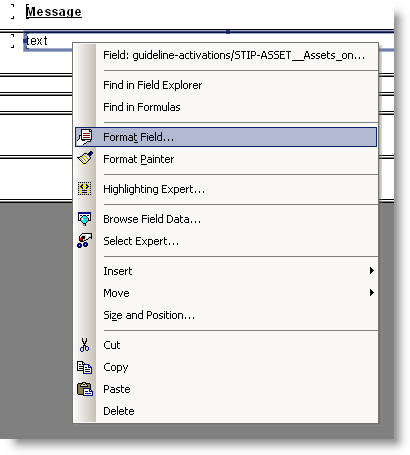


Figure 24: Auto-Sizing Message Text

Congratulations, you’ve created your first template-specific Crystal Report! Your report can be made available for viewing via the PowerEditor GUI. Please see the *PowerEditor User’s Guide* for information about viewing custom reports.

The following section summarizes, and sometimes elaborates on, some of the technical details contained in the tutorials above. The section can serve as a reference once the fundamentals that are presented in the tutorials are understood.

# Crystal Reports™ Tutorial: Multiple Templates

This section walks you through the creation of a multiple template report (or *generic-template* report), which shows all the activations within a specific usage-type or set of usage types. It is assumed that the reader has already created a single-template report, as described above.

## Multiple Template Reports: Overview

A conceptual overview of a multiple template report is needed before launching into the details. The idea of a multiple template report, or a *generic template* report, is that the report writer would like to display all the grid data for a set of templates, without individually specifying each template and column. For example, the report writer would like to see all the grid data in the templates whose usage-type is “Pricing-Adjustments”.

The difficulty with generic template reports is that Column 1 for the “Property Pricing Adjustment” template is different than Column 1 of the “Doc Type Pricing Adjustment” template. Furthermore, it is often the case that all templates to be reported have a certain column in common that should be treated specially. For example, all Pricing-Adjustment templates might have a Rate Adjustment column. This column should always appear in the same place in the report, in spite of the fact that they have different column numbers in the PowerEditor. The figure below shows these examples.

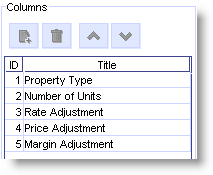
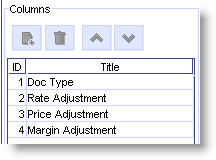


Figure 25: PowerEditor: Sample Pricing Adjustment Templates

For the purpose of this discussion, columns that are common to each template will be referred to as *Common Columns*, while columns whose inclusion vary by template will be referred to as *Independent Columns.* The remainder of this section describes how to create generic template reports.

Common Columns can be relevant to the report generation process under any of the following conditions:

* The data from your common columns needs to appear in the same location on the report (e.g. pricing adjusters).
* Data needs to be sorted by column value (e.g. stipulation category).

## Create Report: Specify Data Sources

To create a multiple template report, first create a new blank report. This can be achieved by selecting the Start Page tab, or selecting the menu item File->New->Blank Report.

### Define Data Source

To specify the PowerEditor as a data source, once again select Create New Connection, then XML, from the Database Expert dialog, as shown below.

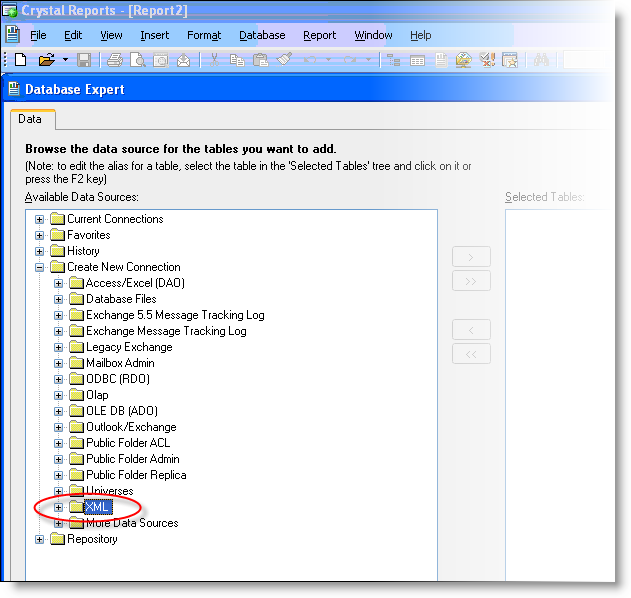


Figure 26: Database Expert Dialog

After selecting XML as the data source type, you will be presented with a series of five dialogs. In this series of dialogs, you will specify the URL that points to the PowerEditor data, as well as the URL which points to the XML schema. This series of five dialogs is described below.

#### URL for XML data

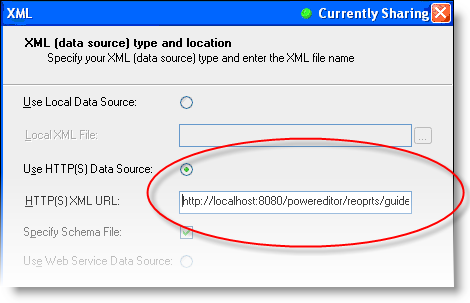


Figure 27: Dialog 1 in XML Specification

In the first dialog, select the radio button labeled “Use HTTP(S) Data Source”. Then, you will need to type in the URL for the data you want[[2]](#footnote-2). An example URL is as follows:

http://localhost:8080/powereditor/reports/**generic-templates.jsp**?usage=MMS-Create-Stipulations&columns=Code,Category

The syntax of this URL is as follows. The 3 highlighted pieces of the URL are options that are described below.

http://*WebserverPort*/powereditor/reports/**generic-templates.jsp**?*ParamName*=*ParamArg(s)*& *Param2Name*=*Param2Arg(s)*

*WebserverPort* This is the location of the web-server that hosts the PowerEditor. This portion of the URL is exactly the same as the URL that you use to launch the PowerEditor. Examples include localhost:8080 or 172.168.1.1:8080

*ParamName* This specifies the group of PowerEditor data that you will be drawing from in the report. The group of data that you specify here will probably be a superset of the data that actually gets included in the report. You will have a chance to refine the set of data later in the report specification process. There are four valid values for *ParamName*:

* usage – Specifies that you would like to report on data from the PowerEditor templates within the specified usage-type(s).
* columns – Specifies the names of the Common Columns (described in *Section 2.1 Multiple Template Reports: Overview*). The columns specified here should be columns that appear in all (or most) of the templates that are in the report.
* templateid – Specifies the ID(s) of the template(s) that you would like to report on. Although this argument is valid syntax for multiple-template reports, its usefulness is limited.
* template – Specifies the name(s) of the template(s) that you would like to report on. Although this argument is valid syntax for multiple-template reports, its usefulness is limited.

*ParamArg(s)* A comma delimited list of arguments to the given parameter.

* usage - The argument(s) to the usage parameter must be usage-type names as defined in your PowerEditor configuration (not to be confused with the usage-type display names). Typical examples include Pricing-Adjustments and MMS-Create-Stipulations. You would use this parameter if you intend your report to contain more than one template-specific report.
* columns –The argument(s) to the columns parameter must be the value of the column’s *Title* field as specified in the PowerEditor template, e.g. Category,Rate Adjustment.
* templateid – The argument(s) to the templateid parameter are numerical template IDs, e.g. 31024.
* template – The argument(s) to the template are template names as they appear in the PowerEditor Manage Templates screen. These names can contain spaces.

*ParamName2* If there are two *ParamNames* specified in a single URL, one of the parameters must be columns. In other words, usage, templateid, and template are mutually exclusive arguments: they cannot be used in conjunction with each other.

**Example URLs:**

http://172.168.0.0:8080/powereditor/reports/generic-templates.jsp?usage=MMS-Create-Stipulations, Pricing-Adjustments

http://localhost:8080/powereditor/reports/generic-templates.jsp?usage=MMS-Create-Stipulations&columns=Code,Category

http://amx-uat01:8080/powereditor/reports/generic-templates.jsp?usage=Pricing-Adjustments&columns=Rate Adjustments,Margin Adjustments,Price Adjustments

In Dialog 1, after entering a URL, select the *Next>* button.

Dialog 2 can be ignored – no authentication is needed. Simply hit the *Next>* button.

Dialog 3 prompts for the XML schema. Make sure that the radio button labeled “Use HTTP(S) Schema” is selected. The URL to be entered here is *exactly* the same as the one you entered in Dialog 1, with the following exception: replace /reports/ with /reports/schema/

After entering a valid URL in Dialog 3, simply select the *Finish* button, or select *Next>* for the remainder of the dialogs.

## Create Report: Specify Data Elements

When you successfully selected your data sources that will be included in the report, you are ready start selecting specific data elements from the data source. You will be in the Report layout tab, and will be presented with a couple more dialogs.

After being presented with the Parameter Values dialog, you will then be presented with the Database Expert dialog that you’ve seen before. Expand the guideline-activations box to see available table entries. A sample expanded list is shown in below.

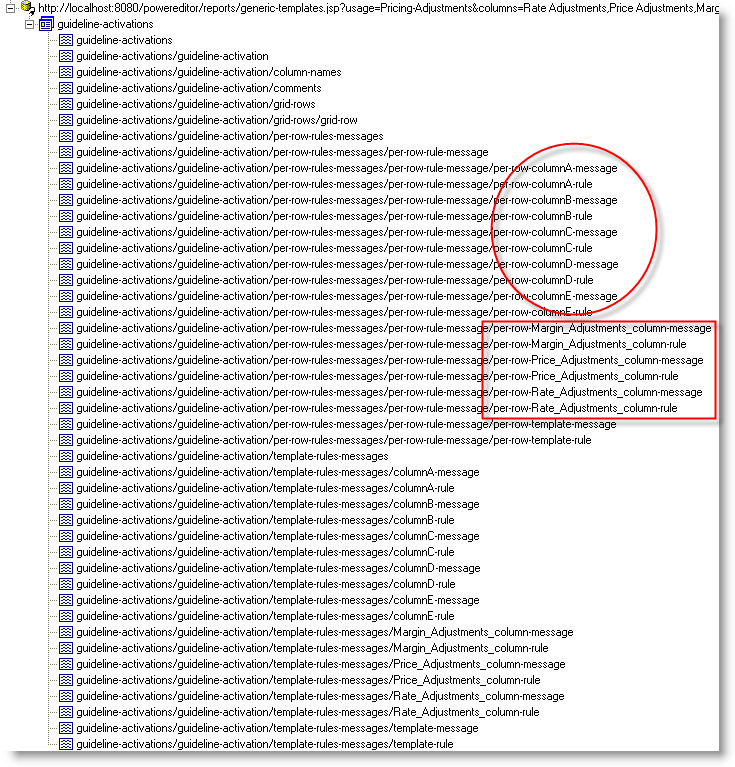


Figure 28: Database Expert Dialog After Data Source Selection

Note in the figure above that Common Columns are listed differently than Independent Columns. Independent Columns (highlighted by a circle above) are named columnA, columnB, etc. Common Columns (highlighted by a square above) are named by their PowerEditor column title.

The Independent Columns listed here represent all of the columns that could appear in any of the specified templates. Therefore, the number of Independent Columns listed is max number of columns in any of the templates of the given usage-type(s). For example, in the two templates in *Figure 25: PowerEditor: Sample Pricing Adjustment* Templates, the number of Independent Columns would be 5.

The remainder of this section describes the contents of the listing above.

*column-names* This table provides a reference to the title (or column heading) for each column.

*comments* This references the comments associated with each activation of the template.

*grid-rows* This references the data in the guideline grids that are associated with each activation of the template. This is just a grouping of grid-row elements (described next)

*grid-row* This references the data in a single row of the guideline grid.

*per-row-rules-messages* References textual forms of rules or messages that are generated for each row. For the each rule or message, row data is substituted in the template rule or message (e.g. “Property Type = Attached”).

*template-rules-messages* References textual forms of rules or messages that are generated for a template. These are template level or column-level rules and messages, whose cell data is not filled in. (e.g. “Property Type = %column 1%”)

To proceed from this dialog, select the data that you will use in your report. There is no harm in selecting all the tables listed in this dialog – this is actually recommended.

Then, proceed through the remainder of the dialogs to get to the Report Layout screen.

⮱ Note: It would be a good idea to save the work you have done so far. And maybe watch the birds for a bit.

## Create Report: Current Rate Adjusters

The following shows how to create a report of all price adjusters that are currently active.

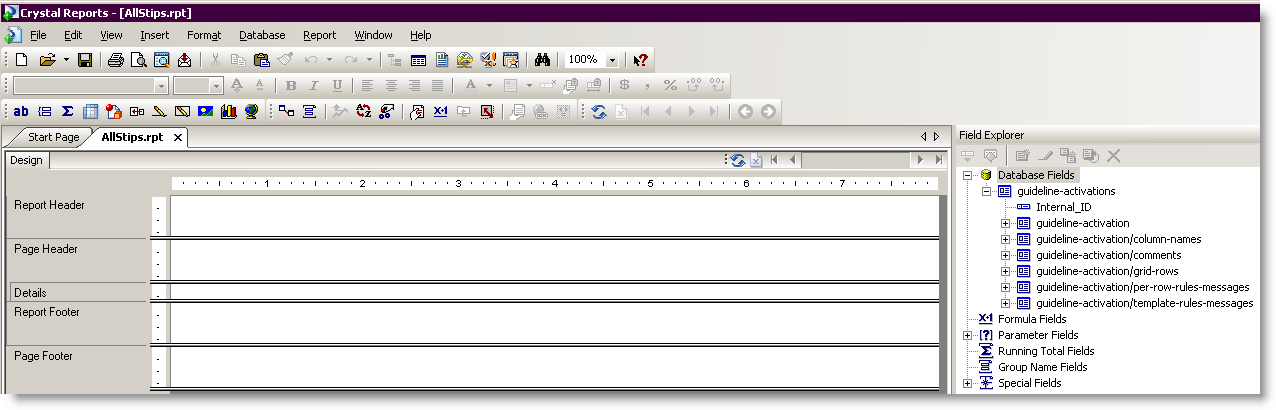


Figure 29: Initial Report Layout Screen

### Step 1: Add Group For Template and Activation

For Generic Template reports, a group must be specified for a template ID or name. Without this group, column references will be meaningless.

In this example, we are filtering data by activation data, so it is helpful to put the activation date in the group too.

Choose the Group Expert icon . Selected the template-id as the primary sort for your group, then the activation-date as the secondary sort. Your dialog should appear as follows:

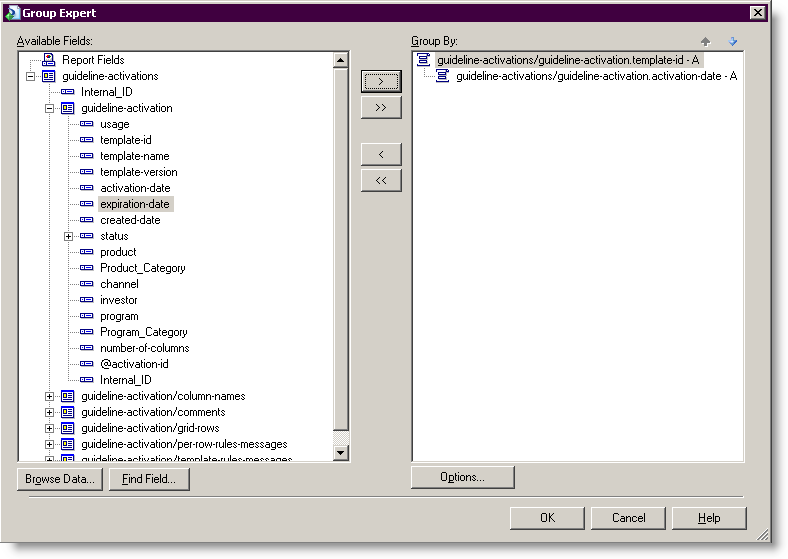


Figure 30: Select Template ID and Activation Date as Sort Values

### Step 2: Add Grid Data

To add grid data, expand the grid-rows, grid-row and column-names database fields. The column headings will appear by default in the Page Header section. The Crystal generated column headers for Common Columns can be used as is. However, for Independent Columns, the Crystal column headers must be deleted, and replaced with the columns in guideline-activation/column-names. This will provide template-specific column names. Note that guideline-activation/column names must be placed within the template group, or another group within the template group.

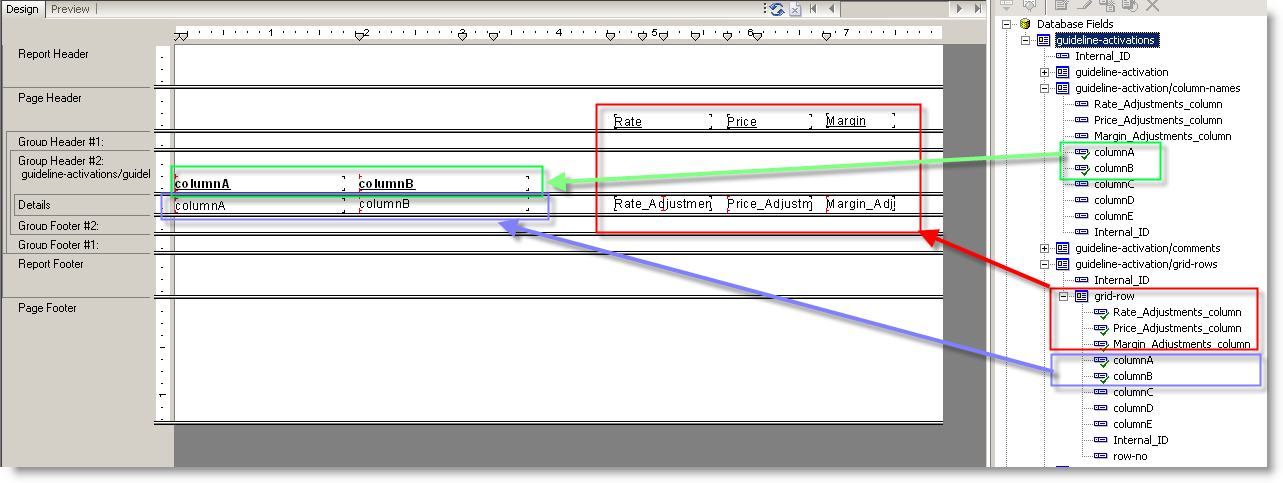


Figure 31: Adding Grid Data With Column Headings

### Step 3: Add Filter Formula

To filter out activations that are currently expired, the Crystal Reports formula editor can be used. Select the menu Report->Formula Workshop (or the icon). You’ll be presented with similar to the dialog below. Note that this figure has the formula used for selecting activations that are currently active.

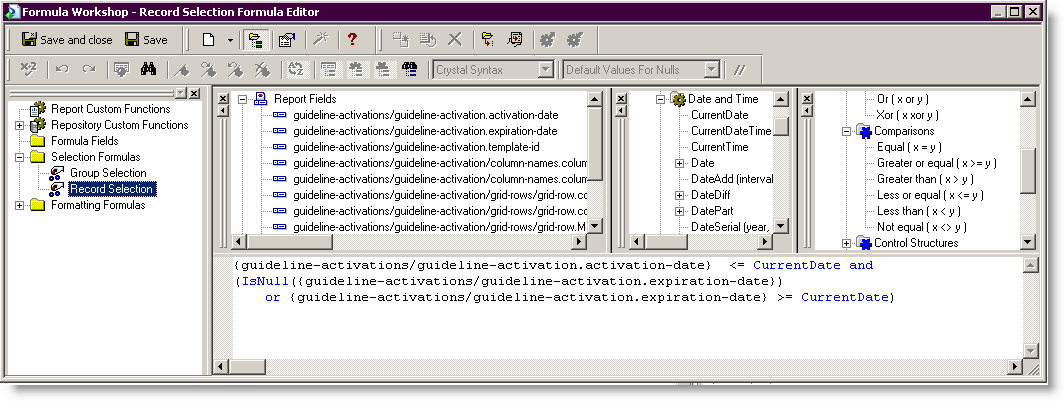


Figure 32: Formula Editor – Selecting Currently Active Activations

# Crystal Reports Tutorial: Audit REport

This section assumes that you are familiar with the material presented in Chapter 1.

## Audit Report Overview

The audit reports are different from the guideline reports in that you can write audit reports in a way that can be run against any PowerEditor database. This means you could potentially reuse a single report for use with all of your PowerEditor projects. This portability is possible because structure of the "Database Fields" (XML schema) for an audit report can be independent of the data in the Knowledge Base. (Note: You need to follow certain guidelines to achieve this portability, which will be highlighted below).

An *audit-report* is a collection of a*udit-event*s. Each audit-event represents a single user action. For example, logging into the PE is an audit-event. Saving a guideline that has 3 edited cells is a single audit-event. The audit-event contains the event date, type, name of the user who made the change, as well a section that contains the details of what changed. The section that contains the details is the *audit-event/change-details* (Note: that's change-details *plural* – with an "s" on the end).

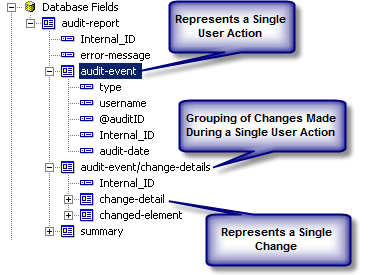
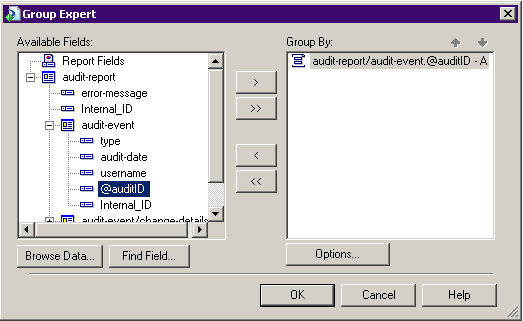
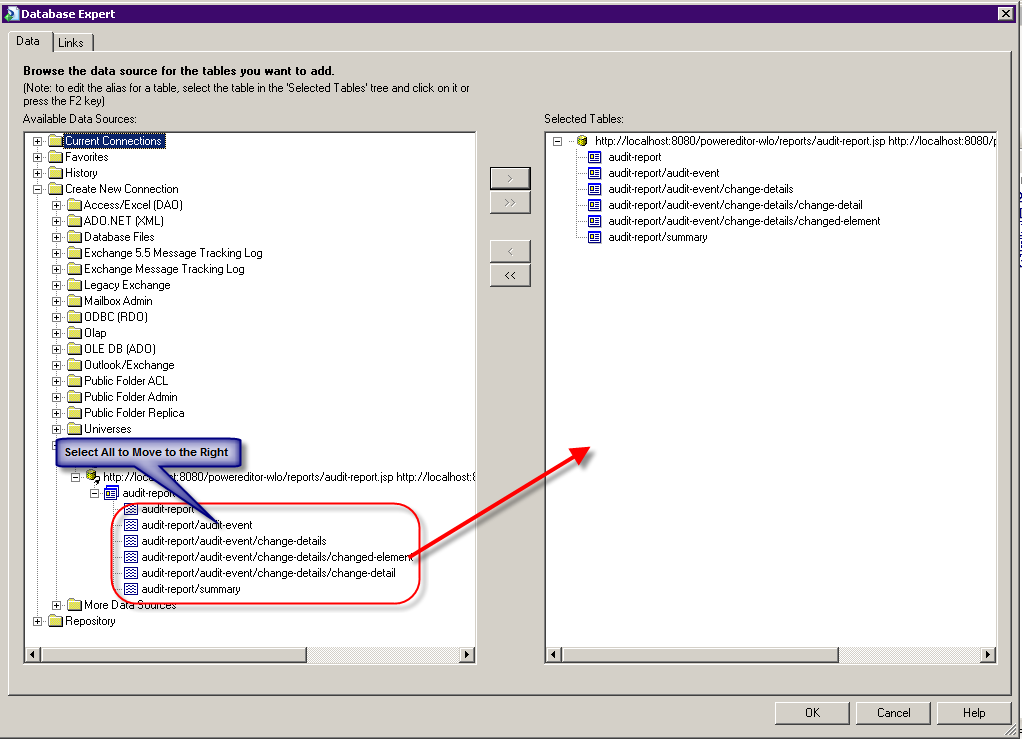


Figure 33: Composition of an *audit-event*

Let's look now at the details of an event. The *audit-event/change-details* is a single section that contains a list of *change-detail* elements, and a single *changed-element*. For example if a user saves a guideline that had 3 cell edits, the single *changed-element* will describe what guideline changed, and there will be 3 *change-detail* elements, each describing a single cell change. The change-detail element contains the row number, column number, previous value, and new value.



# Crystal Reports™ Interface to PowerEditor: Technical Reference

The following section summarizes, and sometimes elaborates on, some of the technical details contained in the tutorials above. The section can serve as a reference once the fundamentals that are presented in the tutorials are understood.

## Crystal Report Version Information

Crystal Reports XI Release 2 Developer Edition is required for building PowerEditor reports. Specifically, PowerEditor Version 4.4.0 has been tested with Crystal Reports Version 11.0.0.2002.

## Making Report Available to PowerEditor GUI

Reports can be made available for viewing from the PowerEditor GUI. Simply move the .rpt file created in Crystal Reports into the PowerEditor web-server directory, under webapps\powereditor\WEB-INF\classes.

Once the .rpt file is moved, go to the PowerEditor GUI, select the Report Tab then the Refresh button. The new custom report should now appear in the list of available reports.

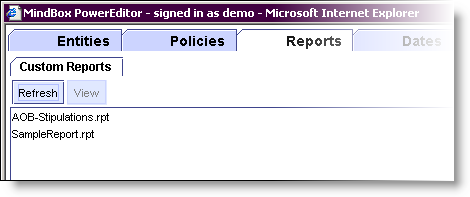


Figure 34: PowerEditor Reports Tab

⮱ Note: Crystal Reports files that contain sub-reports cannot be viewed from the PowerEditor. However, these can be viewed in the Crystal Reports Developer software.

## Refreshing PowerEditor Data Within Crystal Reports Developer

When in Crystal Reports Developer, the contents of the PowerEditor database may have been changed since the report was created. For example, someone may have added a new activation or edited a template since the report was created. In this case, you will need to refresh your report. Similarly, if you have opened a saved report, you may also need to refresh your report. This can be achieved via the following steps. *Note: these steps are rules of thumb that have been discovered by trial and error, and may not be the most efficient means of achieving the refresh.*

1. Choose the Database -> Verify Database menu item. This could take a few seconds.
2. Choose the Report -> Refresh Report Data menu item.

If this does not work, try the following measures:

1. Open Database Expert. Expand Current Connections. Select the used data source and right click. Choose "Refresh" from the popup menu. Then click OK to dismiss Database Expert.
2. In the Field Explorer, select a top level table. Right-click, and choose "Set Datasource Location".
3. In the Set Database Location dialog, you should see the Current Connections in the bottom panel (under "Replace with:"). Choose the appropriate table and click on the Update button to the right. Then, click OK to dismiss the dialog.

## URL Specifications

This section summarizes the syntax of the URL that is specified in the Source Data specification. The highlighted pieces of the URL are options that are described subsequently.

### XML Data Source Data URL

The following is the syntax for the HTTP(S) XML URL specified in the dialog entitled: “XML (data source) type and location”.

[http://*WebserverPort*/powereditor/reports/*JspPageName*.jsp?*ParamName*=*ParamArg(s)*&*Param2Name*](http://WebserverPort/powereditor/reports/JspPageName.jsp?ParamName=ParamArg(s)&Param2Name)*=Param2Arg*

*WebserverPort* This is the location of the web-server that hosts the PowerEditor. This portion of the URL is exactly the same as the URL that you use to launch the PowerEditor. Examples include localhost:8080 or 172.168.1.1:8080

*JspPageName* One of the following:

* specific-template Guidelines for Single Template
* generic-templates Guidelines for Multiple Template
* audit-report View change history
* entity-report Details of entities and categories
* guideline-template-report Contents of a Guideline Template [[3]](#footnote-3)
* parameter-template-report Structure of a Parameter Template [[4]](#footnote-4)

Each of these will be documented in the following sections

*ParamName* (Optional) This specifies the group of PowerEditor data that you will be drawing from in the report. The group of data specified here could be a superset of the data actually included in the report. You will have a chance to refine the set of data later in the report specification process. If there are no *ParamName*s in the URL, all templates will be included. *ParamArgs* consist of a comma-delimited list of arguments to the given parameter. The valid values for *ParamName* and *ParamArgs* are outlined in the following section.

Example URLs:

http://172.168.0.0:8080/powereditor/reports/specific-template.jsp?usage=MMS-Create-Stipulations, Pricing-Adjustments

http://amx-uat01:8080/powereditor/reports/specific-template.jsp?templateid=35225,35220

http://amx-uat01:8080/powereditor/reports/specific-template.jsp?templateid=31660

http://172.168.0.0:8080/powereditor/reports/specific-template.jsp?template=CORE:Units-Property,STIP-CREDIT: Max DTI

http://172.168.0.0:8080/powereditor/reports/generic-templates.jsp?usage=MMS-Create-Stipulations, Pricing-Adjustments

http://localhost:8080/powereditor/reports/generic-templates.jsp?usage=MMS-Create-Stipulations&columns=Code,Category

http://amx-uat01:8080/powereditor/reports/generic-templates.jsp?usage=Pricing-Adjustments&columns=Rate Adjustments,Margin Adjustments,Price Adjustments

http://172.168.0.0:8080/powereditor/reports/entity-report.jsp?entity-types=product

### URL for XML Schema URL

The following is the syntax for the HTTP(S) Schema URL specified in the dialog entitled: “Schema file type and location”.

The URL to be entered here for the Schema specification is *exactly* the same as the one you entered for above, with the following exception: replace /reports/ with /reports/schema/

**Example URLs:**

http://172.168.0.0:8080/powereditor/reports/schema/specific-template.jsp?usage=MMS-Create-Stipulations, Pricing-Adjustments

http://amx-uat01:8080/powereditor/reports/schema/specific-template.jsp?templateid=35225,35220

http://amx-uat01:8080/powereditor/reports/schema/specific-template.jsp?templateid=31660

http://172.168.0.0:8080/powereditor/reports/schema/specific-template.jsp?template=CORE:Units-Property,STIP-CREDIT: Max DTI

http://192.168.0.0:8080/powereditor/reports/schema/entity-report.jsp

## Parameter Arguments

The following is a table of parameters available by each report type:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **audit-report** | **generic-template** | **specific-template** | **entities** |
| status | ☑ | ☑ | ☑ |  |
| context-elements | ☑ | ☑ | ☑ |  |
| include-parents | ☑ | ☑ | ☑ |  |
| include-children | ☑ | ☑ | ☑ |  |
| include-empty-context | ☑ | ☑ | ☑ |  |
| template | ☑ | ☑ | ☑ |  |
| template-id | ☑ | ☑ | ☑ |  |
| usage | ☑ | ☑ | ☑ |  |
| filter-column-data | - | ☑ | ☑ |  |
| columns | - | ☑ | - |  |
| date | - | ☑ | ☑ | ☑ |
| types | ☑ | - | - |  |
| begin-date | ☑ | - | - |  |
| end-date | ☑ | - | - |  |
| entity-type |  |  |  | ☑ |

Each of these parameters is described in detail in the sections below. The parameters are grouped into four sections as indicated by the groupings in the table above: All Report Types, Audit Reports, Guideline Reports, and Entity Reports.

### Parameter Arguments: All Report Types (Except Entity Report)

**Parameter Name: Usage**

Description

Specifies that you would like to report on data from the PowerEditor templates within the specified usage-type(s).

Parameter Arguments

Usage-type names, as they are defined in your PowerEditor configuration file (not to be confused with the usage-type display names). Typical examples include Pricing-Adjustments and MMS-Create-Stipulations. Use this parameter if you intend your report to contain more than one template-specific report.

**Parameter Name: templateid**

Description

Specifies the ID(s) of the template(s) you’d like to report on.

Parameter Arguments

Numerical template IDs, e.g. 31024.

**Parameter Name: template**

Description

Specifies the name(s) of the template(s) that you would like to report on.

Parameter Arguments

Template names, as they appear in the PowerEditor Manage Templates screen. The names can contain spaces.

In any given report, except Entity reports, you can use at most one of the following:

usage, templateid, or template.

**Parameter Name: context-elements**

Description

Narrows the list of guidelines returned to those whose context includes the specified entities/categories.

Parameter Arguments

A list of comma-delimited entities or categories. If categories are specified the guidelines returned will be those that appear in *all* the categories specified (i.e. AND logic is used). Each comma-separated value must be one of the following formats:

<entity-type>:entity:<entity-name>

   <entity-type>:category:<category-name>

Example:

context-elements=product:entity:FNMA Conforming Fixed 15

context-elements=product:category:Fixed

context-elements=product:category:Fixed,product:category:Conforming

If the category names in your application are not unique they can be fully qualified with the “->” separator. For example:

context-elements=product:category:Root->AmTerm->Fixed

**Parameter Name:** **include-children**

Description

If context-elements are specified, this indicator specifies whether the resulting guidelines should include ones whose context contain children categories/ entities of the provided context elements.

Parameter Arguments

Must be one of true or false. If not specified, defaults to false.

Example:

context-elements=product:category:Fixed&**include-children**=true

**Parameter Name:** **include-parents**

Description

If context-elements are specified, this indicator specifies whether the resulting guidelines should include ones whose context contain parent categories of the provided context elements.

Parameter Arguments

Must be one of true or false. If not specified, defaults to false.

Example:

context-elements=product:category:Fixed&include-parents=true

context-elements=product:category:Fixed&include-parents=true&include-children=true

**Parameter Name:** **include-empty-contexts**

Description

If context-elements are specified, this indicator specifies whether guidelines with blank contexts should be included in the results.

Parameter Arguments

Must be one of true or false. If not specified, defaults to false.

For example:

context-elements=product:category:Fixed&include-empty-contexts=true

context-elements=product:category:Fixed&include-empty-contexts=true&include-parents=true

**Parameter Name:** **status**

Description

Narrows the list of resulting guidelines returned to those whose status is the specified value.

Parameter Arguments

### Parameter Arguments: Guideline Reports

This section describes the following two report types:

* specific-template Guidelines for Single Template
* generic-templates Guidelines for Multiple Template

When you are generating either of these report, you can use any of the parameters listed in *Section 4.1.1 Parameter Arguments: All Report Types*. In addition, you can use any of the following parameter arguments that are specific to guideline reports.

**Parameter Name: columns**

Description

For Generic-template reports only. Specifies the names of the Common Columns (described in *Section 2.1 Multiple Template Reports: Overview*). The columns specified here should be columns that appear in all (or most) of the templates that are in the report.

Parameter Arguments

The value of the column’s Title field, as specified in the PowerEditor template

Example

columns=Category, Rate Adjustment.

**Parameter Name:** **date**

Description

Narrows the list of resulting guidelines returned to those that are active on the given date.

Parameter Arguments

The value must be in the format: MM/dd/yyyy HH:mm:ss.

Example

date=11/01/2006 01:01:01

**Parameter Name: filter-column-data**

Description

Used to specify whether context columns grid data should be filtered based on the parameters.

Parameter Arguments

One of the following: true, false. If not specified, defaults to false.

### Parameter Arguments: Audit Reports

When you are generating an audit report, you can use any of the parameters listed in *Section 4.5.1 Parameter Arguments: All Report Types*.

In addition, you can use any of the following parameter arguments that are specific to the audit report.

**Parameter Name: types**

Description

Specifies the types of events to report upon.

Parameter Arguments

Must be a comma separated list of valid integer values, which are

* 1 : Logon events
* 2 : Logoff events
* 3 : Server Startup events
* 4 : Server Shutdown events
* 5 : KB Modification events

Example

types=5

types=1,2

**Parameter Name: begin-date**

Description

The date and time of audit logs to retrieve from. If set, only the audit logs that are logged at or after the specified date are returned.

Parameter Arguments

In the format of “mm/dd/yyyy hh:mm:ss.”

Example

begin-date=01/28/2007 00:00:00

**Parameter Name: end-date**

Description

The date and time of audit logs to retrieve from. If set, only the audit logs that are logged at or before the specified date are returned.

Parameter Arguments

In the format of “mm/dd/yyyy hh:mm:ss.”

Example

end-date=01/28/2007 00:00:00

### Parameter Arguments: Entity Reports

You can use any of the following parameter arguments that are specific to the entity report.

**Parameter Name:** **date**

Description

Date used for determining the list of categories to which resulting entities are associated. If not specified, the current date and time is used.

Parameter Arguments

The value must be in the format: MM/dd/yyyy HH:mm:ss.

Example

date=11/01/2006 12:00:00

**Parameter Name:** **entity-type**

Description

Specifies the types of entities to report upon

Parameter Arguments

Must be a comma separated list of valid entity type names, which are defined in the PowerEditor Configuration file.

Example

Entity-type=product,channel

## Database Tables and Fields

This section summarizes some of the Database Tables and Fields that can be accessed in the PowerEditor reports.

*<template>*  For specific-template reports only. This contains template-level data, such as template ID and template name.

*context* For specific-template reports only. This table references the context associated with each activation of the template.

*comments* This references the comments associated with each activation of the template.

*grid-rows* This references the data in the guideline grids that are associated with each activation of the template.

*grid-row* This references the data in a single row of the guideline grid.

*row-messages* For specific-template reports only. This table references the rule message generated for a single row of the guideline grid.

*row-rules* For specific-template reports only. This table references the textual representation of rule message associated with single row of the guideline grid.

*per-row-rules-messages* For generic-templates reports only. References textual forms of rules or messages that are generated for each row. For the each rule or message, row data is substituted in the template rule or message (e.g. “Property Type = Attached”).

*template-rules-messages* For generic-templates reports only. References textual forms of rules or messages that are generated for a template. These are template level or column-level rules and messages, whose cell data is not filled in. (e.g. “Property Type = %column 1%”)

1. Note, that defining a report that contains more than one specific-template report requires the use of Crystal Reports sub-report mechanism. This feature is not described in this document, but is described in the Crystal Reports documentation. [↑](#footnote-ref-1)
2. Hint: You will find that you want to store your commonly used URLs in a text file that you can easily copy and paste from – you’ll be typing this URL many times as you are working with Crystal Reports. [↑](#footnote-ref-2)
3. This report type is not documented here. If you are interested in this report type, please contact technical support. [↑](#footnote-ref-3)
4. This report type is also not documented here. [↑](#footnote-ref-4)