InterSystems® Caché

Using DeepSee Visual Reporting

Version 2017.2
2020-06-25

Using DeepSee Visual Reporting
Caché   Version 2017.2   2020-06-25
Copyright © 2020 InterSystems Corporation
All rights reserved.

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Response Center (WRC)

Tel:	+1-617-621-0700
Tel:	+44 (0) 844 854 2917
Email:	support@InterSystems.com

Table of Contents

List of Figures

About This Book

This book explains how to generate reports using DeepSee Visual Reporting. This book contains the following chapters:

- "Introducing DeepSee Visual Reporting"
- "Editing the Report Content"
- "Editing the Report Layout"
- "Security"

With this appendix:

- "Customizing Paper Options"

1
Introducing DeepSee Visual Reporting

DeepSee Visual Reporting is a graphical user interface to Zen reports, which allows you to create reports from data stored in the Caché database. It consists of a suite of web-based tools including graphical query generators and layout editors. It uses the same core engine as Zen reports and is built on top of the existing Zen reports architecture. The GUI makes the report authoring process easier, and hides much of the complexity of the underlying technology.

Creating a report in DeepSee Visual Reporting is a three-step process:

1. The first step is to specify the data set that drives the report, and to specify how it should be sorted and grouped.

2. The next step is to specify how the data should be formatted for presentation in the report. The graphical UI and editing tools help you visualize and fine tune the report output.

3. The final step is to run the report. At this point, you supply values for query parameters, and generate the PDF file.

The following figure presents an overview of report production with DeepSee Visual Reporting. Colored blocks indicate DeepSee Visual Reporting classes. In the data specification step, you can use a Data Collector to draw data for the report directly from persistent classes, or you can create a Data Set Schema to use existing Zen reports or query classes. Then you build a Data Presenter, which specifies layout and appearance. The final step uses the Data Presenter to generate PDF output.

Figure 1–1: DeepSee Visual Reporting Overview
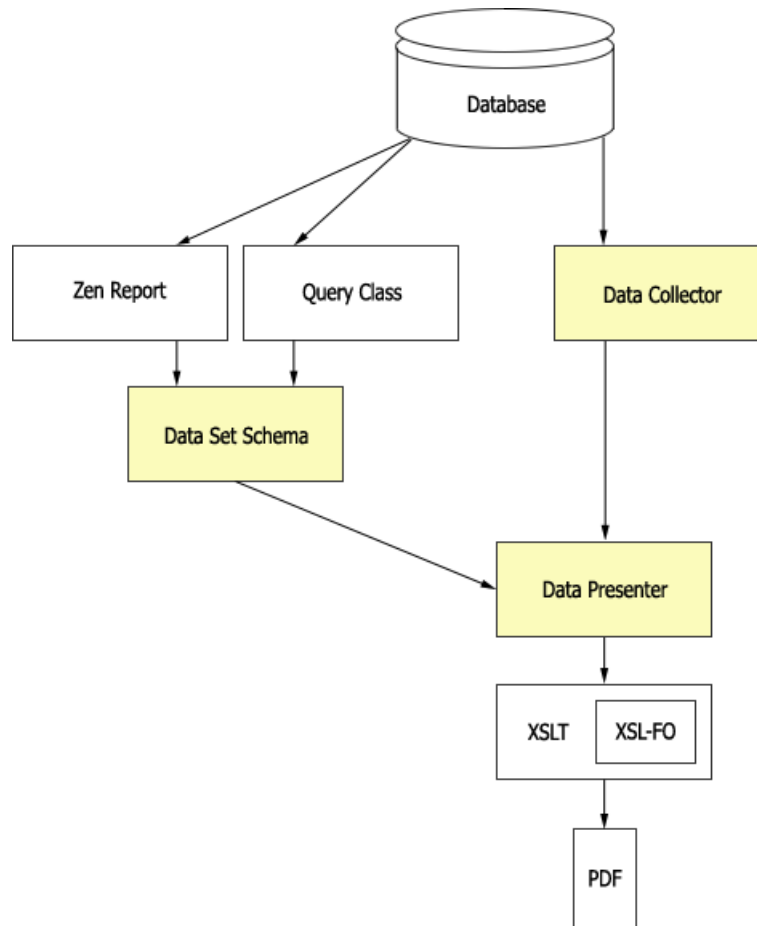
1.1 Additional Reading

For a more detailed understanding of Zen reports, see *Using Zen Reports*. For information on using the Management Portal, see the chapter "Using the Management Portal" in the book *Caché System Administration Guide*.

1.2 Tutorials

1.2.1 DeepSee Visual Reporting Overview

The following exercise uses DeepSee Visual Reporting to create and run a simple report. If you have a new Caché installation, before you begin this exercise you must first run the ZENDemo home page. Loading this page silently generates data records for the SAMPLES namespace. You only need to do this once per Caché installation.

Enter the following URI in the browser:

`http://localhost:57772/csp/samples/ZENDemo.Home.cls`

Where 57772 is the Web server port number that you have assigned to Caché.

Now begin the exercise as follows:

- Initial set-up:

  1. Click the InterSystems Launcher and then click **Management Portal**.

     Depending on your security, you may be prompted to log in with a Caché username and password.

  2. Switch to the SAMPLES namespace as follows:

     a. Click **Switch**.

     b. Click SAMPLES.

     c. Click **OK**.

- Building the Data Collector:

  1. Click **DeepSee**, **Visual Reporting**, **Report Data Definition**, and then **Data Collection Builder**.

  2. Click **New**. The system displays a dialog box where you enter the following values:

     a. **Package Name** — MyApp

     b. **Class Name** — TheaterDC

  3. Click **OK**. The page shows the default Data Fields runTime and runBy.

  4. At the top of the **Data Source** column, click **Change**. The system displays a dialog box. Select Cinema.Theater.

     The page now displays a default data structure for the persistent class Cinema.Theater.

  5. Click **Save**.

  6. This page provides tools to refine the data structure of the report. That process is discussed in section "Editing the Report Content." For now, use the default data structure.

- Building the Data Presenter:

  1. Click **DeepSee**, **Visual Reporting**, **Report Builder** and then **Line Item Report Builder**.

  2. Click **New**. The system displays a dialog box where you can enter the following values:

     a. **Package Name** — MyApp

     b. **Class Name** — TheaterReport

  3. Click **OK**.

  4. At the top of the **Fields** column, click **Set source**. The system displays a dialog box. Select the Data Collector you created previously, by clicking MyApp, then TheaterDC.

  5. The page now displays an empty report, with areas where you can define the main sections of a Zen report:

     – Define Report Header

     – Define Page Header

     – Report Body (query result)

     – Define Report Footer

     – Define Page Footer

  6. At the top of the **Fields** column, click **Auto-fill**. The system fills in the report with a default set of labels and fields according to a format defined in a default report template.

7. Click **Save**.

8. Click **Preview** to view the default report.

   This page provides a number of editing features you can use to refine the layout and appearance of the report. That process is discussed in the section "Editing the Report Layout".

- Running the Report:

  1. Click **DeepSee**, **Visual Reporting**, and then **Run Reports**.

  2. Click **Open**. The system displays a dialog box where you can select a report. Click MyApp, then TheaterReport. The page updates to show the Data Collector for this report. In this example, that is MyApp.TheaterDC.

  3. Click **Run**. After a brief pause, PDF output appears in the browser window.

1.2.2 Using Zen Reports and Query Classes

DeepSee Visual Reporting also allows you to use existing Zen reports and query classes as the basis for new reports. The following exercise shows you how to use these alternative ways of defining the data structure of a report.

- From an existing Zen report:

  1. Click **DeepSee**, **Visual Reporting**, **Report Data Definition**, and then **Extract from legacy report**.

  2. Click **New**. The system displays a dialog box where you can enter the following values:

     a. **Package Name** — MyApp

     b. **Class Name** — ExtractMyReport

  3. Click **OK**. The page updates to show **Data Source** and **Data Projection** columns.

  4. At the top of the **Data Source** column, click **Change**. The system displays a dialog box. Select ZENApp.MyReport. The page updates to show the structure of the report definition. You do not have tools to reorganize the report on this page, because the data structure is determined by the original Zen report.

  5. Click **Save**.

- From a query class

  1. Click **DeepSee**, **Visual Reporting**, **Report Data Definition**, and then **Build from query**.

  2. Click **New**. The system displays a dialog box where you can enter the following values:

     a. **Package Name** — MyApp

     b. **Class Name** — BuildQueryFilms

  3. Click **OK**. The page updates to show **Data Source** and **Data Projection** columns.

  4. At the top of the **Data Source** column, click **Change**. The system displays a dialog box. Select Cinema.Film_TopFilms. The page updates to show the default structure of the report.

  5. Click **Save**.

  6. You can use this page to refine the data structure of the report. That process is discussed in section "Editing the Report Content."

From this point, you can go on to define the data presentation and run the report as described in the steps "Building the Data Presenter" and "Running the Report" in the section "DeepSee Visual Reporting Overview."

2

Editing the Report Content

2.1 Introduction

The Data Collection Builder provides a graphical editor that lets you modify the content and structure of the data retrieved from the Caché database and included in the report. The following section provides a brief introductory tutorial. The subsequent section describes the main elements of the Data Collection Builder user interface.

2.2 Tutorial

In the following exercise, you begin to build a report similar to ZENApp.MyReport provided in SAMPLES namespace.

1. Click **DeepSee**, **Visual Reporting**, **Report Data Definition**, and then **Data Collection Builder**.

2. Click **New**. The system displays a dialog box where you can enter the following values:

    *   **Package Name** — MyApp

    *   **Class Name** — MyReportDC

3. Click **OK**.

4. At the top of the **Data Source** column, click **Change**, and in the dialog box, select ZENApp.Report.Invoice.

    Click **Save**.

    If you go on from this point to define a data presentation and run the report, as you did in the previous tutorial, the result is a report containing all the data you see in ZENApp.MyReport, but without grouping, sorting or aggregation. Instead, follow the next steps to change the organization of the report.

5. Note that by default, all the data fields in the report are in a group with the field name queryResult. Click the **Group** button at the top of the **Data Fields** column. A new line appears in the report with the field name **Group**. Note that the new group is flagged as having a problem, because it has no content.

    Drag all fields except SalesRep from the group queryResult to the new group.

6. Select the field queryResult and perform the following edits in the **Properties** column:

    a. Change **Field Name** from queryResult to SalesRep.

    b. Click the down arrow to the right of the **Break On** field, and select SalesRep from the list.

7. Select the field Group and perform the following edits in the **Properties** column:

    a. Change **Field Name** from Group to Record.

    b. Click the down arrow to the right of the **Secondary Sort By** field, and select SaleDate from the list.

    c. Click **Save**.

From this point, you can go on to define the data presentation and preview or run the report as described in the steps "Building the Data Presenter" and " Running the Report" in the section "DeepSee Visual Reporting Overview."

2.3 Data Collection Builder UI

The Data Collection Builder offers several views of the data definition. The available views are as follows:

- lets you edit the report data definition

- lets you edit report properties

- lets you edit data group filters

- lets you edit supplemental methods

2.3.1 Report Data Definition

The report data definition view organizes the page into three columns:

- Data Source – Shows the class that is providing the data and lists the data fields available from that source.

- Data Fields – Shows the data fields currently in the report, and any grouping, sorting, and aggregation you have added.

- Properties – Lets you edit properties of the report component selected in the **Data Fields** column.

2.3.1.1 Data Source Column

The Data Source column is empty until you have used the **Change** button to select the class that provides data for the report. Once you have selected a data source, the column lists the class you selected, and the data fields available in that class.

2.3.1.2 Data Fields Column

The Data Fields column lists the items and groups currently in use in the report. By default, the report includes items for the time when the report was generated, and the person who generated it. You can delete entries by clicking on the red **X** at the far right of the column. You can add entries by dragging and dropping data fields from the **Data Source** column, or by clicking the **Item** or **Group** buttons at the top of the column. Item and group names must be unique.

Items

Items provide the data that appears in the report. Each entry shows the field name, and the derivation, or source expression, that provides that data. You can add database-derived items by dragging data fields from the Data Source column. You may want to drag the same data field into the report more than once, for example to provide data for aggregation. You can add items that are not derived from the database by clicking the **Item** button at the top of the column.

Groups

Groups let you organize the items in your report. Click the **Group** button to add a group. When you first add a group, it is flagged with an error indicator. You can click on the red exclamation point icon to see the error message, which tells you that the group has no defined contents. Adding items to the group resolves the error condition.

2.3.1.3 Properties Column

The Properties column lets you edit properties of the items and groups listed in the Data Fields column. When you select an entry in the Data Fields column, appropriate fields and values appear in the Properties column.

For items supplied by fields from the database, you can edit the **Field Name**, but not the **Source Expression**. For items not derived from the database, you can edit the **Source Expression** as well. Database-derived fields also provide a field to select an aggregation.

Groups allow you to edit the **Field Name**, which is the name of the group. You can also specify a field to use for sorting and grouping records in the report. When you generate the report, records are sorted by the "Break On" field, and output in groups where the value of the field is the same.

2.3.2 Report Properties

The Report Properties view lets you add properties to the report. Click the **Add** button to add a new property. You can then edit the following information about the property:

- Property – The name of the property. The default value provided is RptProp.

- Type – The datatype of the property. The default value provided is %String.

- URL Token – The name used to provide a value for this property in the URI string supplied to the browser to invoke the report.

- Default – The default value for this property.

- Constraint – Constraints on the value of this property, such as maximum and minimum values.

- Description – A brief description of the property.

2.3.3 Data Group Filters

When you use the **Change** button in the Data Source column of the Report Data Definition view DeepSee Visual Reporting creates an SQL statement that selects all available data records for the report. The Data Group Filters view lets you supply an SQL WHERE clause that limits the data returned to the report.

2.3.4 Supplemental Methods

The Supplemental Methods view lets you add methods to the report. Click the **Add** button to add a new method. The screen provides fields and controls to edit the following information about the method:

- **Name:** – The name of the method.

- **Returns:** – Data type of the value returned by the method. The default is %String.

- **Language:** – The language used to write the method. The default is ObjectScript. You can write other values in the field.

- **Arguments:** – Define arguments for the method. A **+** button at the right end of the form lets you add a new argument. Click on the red **X** at the far right of the form to delete an argument.

- **Implementation:** – Use this field to edit the body of the method.

3

Editing the Report Layout

3.1 Introduction

The Line Item Report Builder provides a graphical editor that lets you modify the default report format to improve the report output. The following section provides a brief introductory tutorial. The subsequent section describes the main elements of the Report Builder user interface.

3.2 Tutorial

In the following exercise, you create and modify the report layout for the data structure you defined in MyApp.MyReportDC.

1. Click **DeepSee**, **Visual Reporting**, **Report Builder** and then **Line Item Report Builder**.

2. Click **New**. The system displays a dialog box where you can enter the following values:

   a. **Package Name** — MyApp

   b. **Class Name** — MyReport

3. Click **OK**.

4. At the top of the **Fields** column, click **Set source**. The system displays a dialog box. Select the Data Collector you created previously, by clicking MyApp, then MyReportDC. At the top of the **Fields** column, click **Auto-fill** to create a default report.

5. Click **Preview** to view the default report.

6. In this step, you modify the report by removing the sysID data field.

   a. In the Report Body section, select either the sysID data field, or field label.

   b. Click **Edit** to open the editing toolbar, and click **Cut**.

   c. The sysID data field disappears from the report. If you want to add a data field to the report, drag it from the Fields column.

7. Edit table properties.

   Select a component in the Report Body section. A collection of tools for formatting tables in reports appears in the Properties column.

---

a. In the section labeled Borders and Dividers, select Toggle table border. It is the first selection in this section, and draws a border around the table.

b. In the section labeled Borders and Dividers, select Toggle zebra striping in body. It is the last selection in this section, and applies zebra stripe coloration to rows in the table.

8. Add an element to the report layout.

   Drag $pageNum from the Fields column and drop it in the Page Footer section. You can position this element by dragging it, or by editing values in the Size and Position section of the Properties column.

9. Click **Save**.

10. Click **Preview** to see the effect of your modifications.

3.3 Line Item Report Builder UI

The Line Item Report Builder provides a single view, which shows the report layout. It provides the following buttons:

- **Open** – Open an existing report

- **New** – Create a new report

- **Save** – Save the current report

- **Setup** – Opens the **Page Set-up Dialog**, which you can use to set characteristics for the printed page.

- **Preview** – Creates a preview of PDF output of the current report

The **Page Set-up Dialog** lets you set the following parameters of the physical page the report is printed on:

- The **Page size** – Select a page size from the dropdown list. The list of available paper sizes can be configured for your system. See Customizing Paper Options.

- The **Orientation** – Select "Portrait" or "Landscape"

- The **Margins** – You can set all four margins with the spinner control or by typing in the text field. Units used for margins reflect the units used to specify the paper size.

- The **Running Headers** – You can choose to omit running headers on the first page, last page, or both.

When a report is open in the editor, the page is organized into the following columns:

- **Fields** – lets you select a data source, and displays the available data fields. The **Auto-fill** button creates a default report layout.

- **Layout** – the area where you edit the report layout. The column is divided into sections that correspond to the layout sections of the report. Buttons at the top of the column provide access to graphical editing tools.

- **Properties** – lets you edit graphical properties of the report component selected in the **Layout** column.

The next sections discuss the content and function of these columns.

3.3.1 Fields Column

The fields column is empty until you select a data collector for the report with the **Set source** button. Once you have selected a data source, the **Fields** column lists the data fields you defined when you created the data collector, along with the additional

fields runTime, RunBy, and $pageNum. You can drag and drop fields from the **Fields** column into the report layout column. When you begin to drag a field, report sections where you cannot drop it turn gray.

The **Auto-fill** button populates the layout section with a default report layout based on the selected data source.

3.3.2 Layout Column

The layout column is where much of the report layout editing takes place. The **Auto-fill** button in the **Fields** column, creates a default report layout in the layout column, which you can use as a starting point. The layout column is divided into a number of sections, and provides a variety of tools and controls.

3.3.2.1 Toolbar Buttons

Buttons at the top of the column provide access to graphical editing tools. Click on the button to open a toolbar, click again to close it. Once the toolbar is open, you can drag it to another part of the page. The following image shows the edit toolbar open and docked with the **Edit** button:
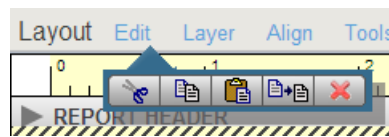
This image shows the edit toolbar after it has been dragged away from the **Edit** button:
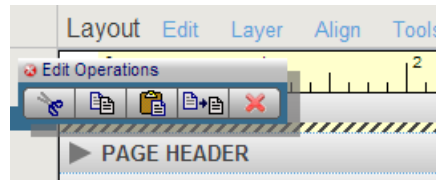
The toolbars invoked by the editing buttons, and the tools they contain, are as follows:

- **Edit** – tools for basic editing functions such as cut and paste:

    - cut selection.

    - copy selection.

    - paste selection previously copied or cut.

    - duplicate selection.

    - delete selection.

- **Layer** – tools to control z-order of objects in the report; for example, to put text in front of a filled rectangle.

    - move to the back

    - move backward one level

    - move forward one level

    - move to the front

- **Align** – tools to align, distribute, and size objects with respect to other objects. The object selected first in the selection group is the standard for alignment, centering, or adjusting height or width. Objects are spaced evenly between the right- and left-most (or top- and bottom-most) objects in the selection group.

- – align left edges.

- – align top edges.

- – align right edges.

- – align bottom edges.

- – center horizontally.

- – center vertically.

- – apply equal horizontal spacing.

- – apply equal vertical spacing.

- – apply uniform height.

- – apply uniform width.

- **Tools** – tools to perform graphical editing functions like moving report elements and adding text and graphics.

  - – select an object. **shift**-click to select multiple objects.

  - – **T** add text.

  - – add an orthogonal line.

  - – add a rectangle.

  - – add a chart.

  - – add a barcode.

  - – add an image.

3.3.2.2 Configuring Charts, Barcodes and Images

When you add a chart, barcode, or image, the editor puts a placeholder object into the report. Right-click on the placeholder to open a dialog box that enables you to supply additional information about the object. The following sections describe how to configure these objects:

Chart

This image shows the chart configuration dialog.
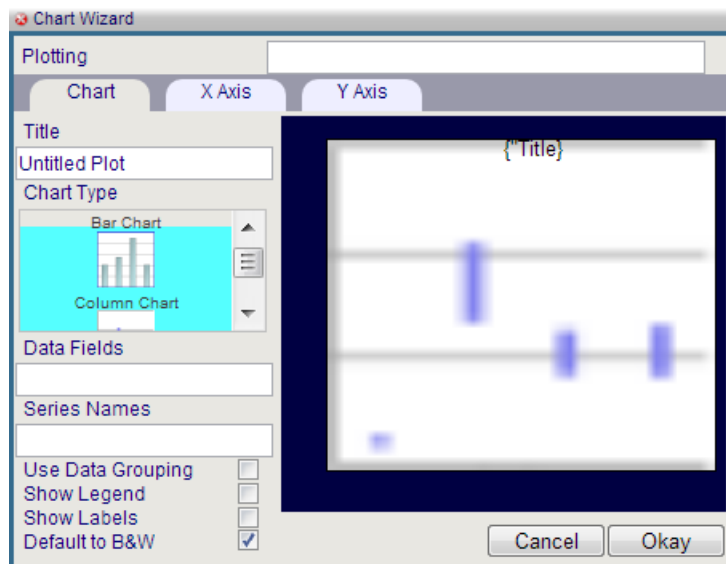
This image shows the dialog with the X-Axis tab active.
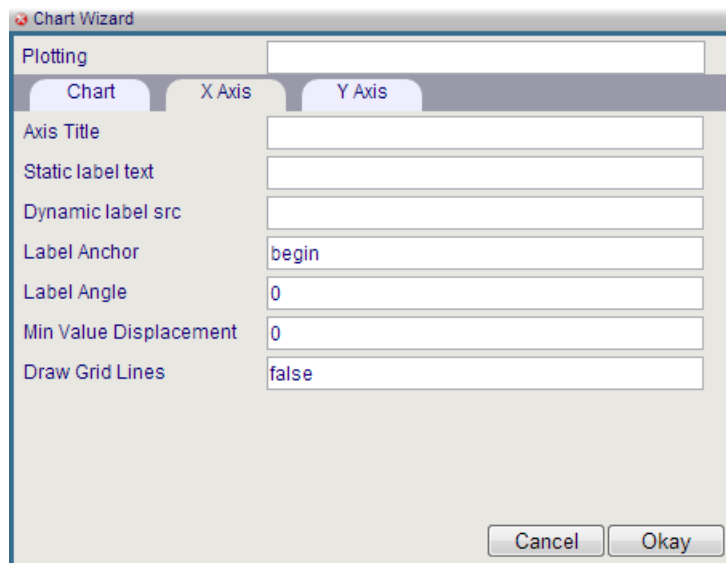
Barcode

This image shows the barcode configuration dialog.

It is divided into the following sections:

*   Data source – This section lets you specify whether the barcode displays static information, or a value from the database. To display a static value, select the **Static** radio button, and type the value into the text field. The barcode placeholder displays the value:

    To display a value from the database, drag and drop a field name from the fields column. The upper-left corner of the field bounding box must be inside the bounding box of the placeholder. The placeholder then displays the XPATH expression the report uses to get data from that field:

    You can also type the XPATH expression directly into the text field next to the **Dynamic** radio button.

*   Format – Select either Code-39 or Code-128.

*   Options – Specify options for the barcode, including whether it renders the value in human-readable text.

Image

This image shows the image configuration dialog. You can specify an image by typing the URL into the text field, or by browsing for a local file.

3.3.2.3 Report Sections

The main area of the Layout column is divided into areas that correspond to the sections that make up a report. These sections are as follows:

• Report Header

• Page Header

• Report Body

• Report Footer

• Page Footer

Each report section has a gray bar across the top of the section that contains a variety of controls. Not all sections contain all available controls. The following list describes these controls:
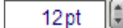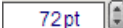
• ➕ add a section.

▶ / ▼ expand/hide the section.

🔲 Graphic banner layout. This is the only available layout for some report sections. It has the following characteristics:

  – Projects as an SVG graphic

  – Height is Fixed

  – Supports absolute positioning of elements

  – Supports layering of elements

  – Supports graphical elements such as bar codes, images, charts, and lines.

  – Rendering styles are static.

  – Text in long fields does not wrap.

  – Only available when processing XSL nodes, not node sets. When used to replace the rendering of a group body, the tool automatically renders one banner layout section per record in the data stream. In essence, it renders a banner layout for what would have been a row in a table.

• 🗒 paragraph style layout. This layout has the following characteristics:

  – Projects as in ZSL-FO Block paragraph

  – Height is variable

  – Margins can be adjusted separately for each item

  – Text wraps automatically

  – Supports limited embedded style markers in data-driven elements

  – Does not support graphical elements

  – Inherently single column flow. No side by side, layering, or multi-column absolute positioning

  – Only available when processing XSL nodes, not node sets. When used to replace the rendering of a group body the tool automatically renders one paragraph layout section per record in the data stream. In essence, it renders a paragraph layout for what would have been a row in a table.

- name/value pair layout. This layout has the following characteristics:

    – Projects as an XSL-FO table

    – Height is variable

    – Text wraps automatically in table cells

    – Columns are Automatically aligned

    – Supports header repeat on multi-page tables

    – Rendering styles are static

    – No Graphical elements other than the table's own grid lines/backgrounds

    – Only available when processing XSL nodes, not node sets. When used to replace the rendering of a group body, the tool automatically renders one name-value pair layout per record. In essence, it renders a name-value pair layout for what would have been a row in a table.

- table layout. This layout has the following characteristics:

    – Projects as an XSL-FO table

    – Height is variable

    – Text wraps automatically in table cells

    – Columns are Automatically aligned

    – Supports zebra striping

    – Supports header repeat on multi-page tables

    – Rendering styles are static

    – No Graphical elements are allowed other than the table's own grid lines and backgrounds

    – Only available when processing XSL node sets (group bodies)

- Add or remove a page break at the end of the section. A line that looks like this: /////// indicates a page break.

- Add a conditional sub-section or running header.

- Delete the section and its contents. You can restore the section by clicking on the gray bar, but the restored section is empty.

3.3.3 Properties Column

The properties column lets you specify the style attributes of objects in the layout column. The possible options are grouped according to the relevant style properties for various types of objects. The selection you make in the layout column determines what options are available in the properties column. The available options are as follows:

- **Fill Style**: Lets you specify fill color and opacity. These options are also available for **Line Style** and **Text Style**.

    – Opens a color picker from which you can select the fill color. The same icon is used to select color for lines and text.

    – Move the slider to select the opacity, from 100% (far right) to 0% (far left). The same icon is used to select opacity for lines and text.

- **Line Style**: In addition to color and opacity, the **Line Style** group also lets you set the following properties.

  - ▆▆▬ Lets you select the line width.

  - ▭┈┉▬ Lets you select the line style.

- **Text Style**: In addition to color and opacity, the **Text Style** group also lets you set set the following properties.

  - B *I* U Lets you select bold, italic, or underlined text. Each button in this control toggles, and you can select more than one.

  - 12pt ⬍ Lets you select the font size by either typing into the field, or using the spinner control.

  - ☰☰☰ Lets you select the paragraph justification.

  - sans-serif ⬍ Lets you select the font family by using the spinner control.

- **Borders and Dividers**

  - Toggles the border surrounding the table.

  - Toggles the border surrounding the table header.

  - Toggles the divider between table header and the table body.

  - Toggles the dividers between header columns.

  - Toggles the dividers between body columns.

  - Toggles the dividers between body rows.

  - Toggles zebra striping in the table body.

  - Opens a color picker that lets you select the alternate color for zebra striping.

- **Size and Position**

  - 72pt ⬍ Lets you select the font size by either typing into the field, or using the spinner control.

4

Security

DeepSee Visual Reporting is secured by **`%DeepSee_ReportBuilder`** and **`%Development`** resources. Specifically, users must have the following privileges to perform the indicated tasks:

- In order to define a report from the **Report Builder** menu, the user must have USE permission for the **`%DeepSee_ReportBuilder`** resource.

- In order to define report data from the **Report Data Definition** menu, the user must also have USE permission for the **`%Development`** resource.

A

Customizing Paper Options

The **Page Set-up Dialog** of the **Line Item Report Builder** enables users to select the paper size for a printed report. Site administrators can customize the list of paper sizes that appears in this dialog in the same manner that they can set up a custom font list.

To add a custom page description, use Studio or another appropriate editor to edit the class %DeepSee.Report.Config.configData. In this class, is an XData section named `CustomPageSizes` whose contents looks like this:

```
<pageSizes>
 <![CDATA[
  [
   {displayName:"Sticker",units:"mm",width:60.9,height:101}
  ]
 ]] >
</pageSizes>
```

This is a JSON encoded array. If the array contains more than one item, the items are separated by commas. Each element of the array is a JSON object with 4 attributes.

- *displayName* is the human readable tag to be associated with this entry

- *units* is the base measure used to specify the width and height scalars and should be either "in" or "mm"

- *width* is the minor dimension of the page in the specified units

- *height* is the major dimension of the page in the specified units

For example, to add a business card blank to the list, edit the file to look like this:

```
<pageSizes>
 <![CDATA[
  [
   {displayName:"Sticker",units:"mm",width:60.9,height:101},
   {displayName:"Card",units:"in",width:2,height:3.5}
  ]
 ]]>
</pageSizes>
```

Note both the comma added as a separator at the end of the first line and that, although most business cards are printed in landscape orientation, the stock itself is defined with the height as the longer dimension.

Once this class is saved and compiled, the **Page Set-up Dialog** of the **Line Item Report Builder** finds it and adds whatever definitions it holds to the drop down of page size options. New options are appended after the "standard" options in the order they appear in the file.

You can export this class to XML to make it portable to other systems. Simply import and compile the file into the SYS namespace.