The Microsoft Report Designer and ReportViewer web server control allow formatted reports to be viewed through an asp.net web page. Report definitions are stored in an XML formatted file with the extension .rdlc. .rdlc files can be used with locally on the web site or on the SQL server using reporting services.

There are a few significant differences between .rdlc reports and reports generated through a GridView.

Columns will wrap on blanks in data. There is currently no way to prevent word-wrap. Columns have to be made wide explicitly to prevent word-wrap. Columns can be set to expand automatically for data which does not contain blanks.

HTML cannot be rendered in a column. By default, any column containing HTML markup will display the markup text. There is a setting which will allow some “style” markup like <b>, <i>, and <u> but any other markup will be lost.

Links have to be set as properties on either the displayed text or on the table cell. The text color and underlining have to be set explicitly in thereport definition.

CSS styles cannot be applied to the report contents. Styles have to be set in the report definition.

# Creating a report

1. Create a TableAdapter to connec t to the desired table or view.
   1. Open DWS No Rep DataDataSet.xsd (App\_Code folder)
   2. Right-click and select Add – TableAdapter
   3. Select Use SQL Statement and enter the desired SQL Statement or use Query Builder. Include any necessary parameters as you would in a SqlDataSource. IMPORTANT: if any of the selected column names include spaces you must use an alias name which does not.
   4. Example SQL:

SELECT Terminal, Date, Profile, Type, Carrier, Description, [Product Group 1] AS Product\_Group\_1, [Product Group 2] AS Product\_Group\_2, [Product Group 3] AS Product\_Group\_3, [Item Ct] AS Item\_Ct, Pounds, [Linear Ft] AS Linear\_Ft, Receiver, BOL, [PDF of Load] AS PDF\_of\_Load, release, Notes, POD, [PDF File Name] AS PDF\_File\_Name

FROM [COPS Transactions Log by carrier for web]

WHERE ([Client ref] = @Client\_ref) AND ([Product Group 1] = @PG1) AND (Profile = @Profile)

ORDER BY Terminal, Date

1. Create the report definition file (Add Item – Report or Report Wizard)  
   During report definition select the table adapter from step 1 above. If starting with a blank report page us the New - DataSet function on the Report Data toolbar. You may have to launch in debug mode for the Report Data toolbar to show.

You may get an error message "Could not load file or assembly BopsDataAccess". If so you will need to rename the BIN folder, then complete the operation and rename it back to BIN. Do not do a build until you have named it back.

Add colums to the report as desired.

1. Add a ReportViewer to the desired asp.net page

Connect the viewer to the report definition (ReportViewer Tasks – Select Report)

1. Configure an ObjectDataSource for the report viewer (ReportViewer Tasks – Choose Data Sources)

Select “New Data Source”, pick “Object” as the source.

Select the TableAdapter from #1 as the Business Object.

Select the GetData method.

Map any parameters as you would for a SqlDataSource.

Report should now display data when run.

# Topics

## Cell and column formatting

Try to set colors and fonts on the first column before adding additional columns. New columns will be set to the same format as existing columns.

## Viewer Scrollbars

Report viewer will have scrollbars if it cannot contain the entire report page. Set the report viewer width and height larger than the size of a report page to remove scrollbars.

## Report Page Size

Select the report in the report designer. InteractiveSize property controls the page size for the report viewer (PageSize sets it for printing). To show fewer row/page reduce the height.

## Header Rows

To get header rows to display on each page.

Reference:

http://blog.teamgrowth.net/index.php/net/how-to-repeat-header-rows-on-each-page-in-reportviewer-rdlc

Click drop-down to far right of "Column groups" at bottom of report in report designer.

Select "Advanced Mode".

Select the "static" item under "Row groups" associated with the header row.

Set properties: KeepWithGroup=After, RepeatOnNewPage=True.

## Hyperlinks

You cannot embed HTML in a field. Links must be set as properties of a PlaceHolder (where the field text goes) or TextBox (the grid cell).

Select element to be linked.

Right-click, select Placeholder properties.

Select action, check "goto url"

If a field contains fully-qualified (http://server/path/file) URL select it. Otherwise use an expression to build a URL.

If unable to build fully-qualified URL use a Javascript link to jump to the page as in:

="javascript:void(window.location.assign('tally\_detail.aspx?Load\_ref=" & Str(Fields!BOL.Value) & "'))"

Set link text color and underline explicitly for the placeholder.

On the report viewer page.

Select the report viewer and go to the properties window.

Set LocalReport.EnableHyperlinks = True.

## Custom Report Functions

There are some custom functions in the CopsWebReportUtils project.

If any of the functions are used in a report and changes are made to that project you need to copy

CopsWebReportUtils.dll to

C:\Program Files\Microsoft Visual Studio 10.0\Common7\IDE\PrivateAssemblies

or builds will fail.

Custom functions are included in a report by adding a reference to CopsWebReportUtils.dll in the report properties. Then the methods in CopsWebReportUtils.dll can be using in report expressions. For example:

="javascript:void(window.navigate('"

& Cops.Web.Reports.COPSReportUtils.HrefFromAnchorTag(Fields!PDF\_of\_Load.Value)

& "'));"

## Parameter selection on page

If the report web page has controls which can be used to change the report search parameters then the report must be explicitly refreshed. Add the following to the “value changed” event associated with each control:

ReportViewer1.LocalReport.Refresh();

i.e;

protected void DropDownList1\_SelectedIndexChanged(object sender, EventArgs e)

{

ReportViewer1.LocalReport.Refresh();

}