

# NetAdvantage<sup>®</sup> Reporting 2012.2 - Release Notes

---



Use NetAdvantage Reporting, the industry's first WPF and Silverlight-based design-time and rendering reporting tool, to create elegant and easy-to-design reports engineered to help you deliver information to your users in the shortest time possible —without the need for legacy code.

## Index

1. **Installation**
2. **What's New – Summary**
3. **Changes/Additions for 2012 Volume 2**
4. **Bug Fixes**

## Installation

<p><b>Downloading</b></p>	<p><a href="#">Download NetAdvantage Reporting here.</a></p>						
<p><b>Notes</b></p>	<p>NetAdvantage Reporting does not support side-by-side installations. Depending on the installation form of your choice, an additional step may be required:</p> <table border="1" data-bbox="453 751 1414 1371"> <thead> <tr> <th data-bbox="453 751 761 835">Installing through...</th> <th data-bbox="764 751 1414 835">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="453 837 761 1161">Platform Installer</td> <td data-bbox="764 837 1414 1161">When installing NetAdvantage Reporting 2012 Volume 2 through the Platform Installer (self-extracting wrapper) and having a previous NetAdvantage Reporting release already installed, it will upgrade your previous installation to 12.2.</td> </tr> <tr> <td data-bbox="453 1163 761 1371">Stand-alone installer</td> <td data-bbox="764 1163 1414 1371">When installing NetAdvantage Reporting using the stand-alone installer you will have to manually uninstall a previous version before installing 12.2.</td> </tr> </tbody> </table>	Installing through...	Description	Platform Installer	When installing NetAdvantage Reporting 2012 Volume 2 through the Platform Installer (self-extracting wrapper) and having a previous NetAdvantage Reporting release already installed, it will upgrade your previous installation to 12.2.	Stand-alone installer	When installing NetAdvantage Reporting using the stand-alone installer you will have to manually uninstall a previous version before installing 12.2.
Installing through...	Description						
Platform Installer	When installing NetAdvantage Reporting 2012 Volume 2 through the Platform Installer (self-extracting wrapper) and having a previous NetAdvantage Reporting release already installed, it will upgrade your previous installation to 12.2.						
Stand-alone installer	When installing NetAdvantage Reporting using the stand-alone installer you will have to manually uninstall a previous version before installing 12.2.						
<p><b>Sample Browser</b></p>	<p><b>Requirements</b></p> <ul style="list-style-type: none"> <li>• <b>RIA Services must be installed.</b></li> <li>• <b>SQL Express is required by default</b>, if not available you may configure or use an alternate connection string defined in the Web.Config file.</li> </ul>						

## What's New – Summary

### What's New – 2012 Volume 2

NetAdvantage Reporting 12.2 release introduces the possibility to display data from hierarchical data sources using sections, tables or a combination of both. Improved performance is achieved, leading to faster rendering, as well as less memory and disk space use. A new control and new layout properties are introduced; *RepeatOnEveryPage* and *CanGrow* properties, and the Page Break control which can be used to add a page break within a section. The lasso tool is now available, allowing you to select multiple elements just by using your mouse. Finally, you can design your reports using Visual Studio 2012 and experience the new theme-aware designer.

### Hierarchical Data Support – NEW!

NetAdvantage Reporting now supports creating reports from hierarchical data sources. Creating a master-detail report (e.g., Orders and Order lines) can be achieved easily by dragging objects to the design surface. There is no need for sub-reports or complex configurations. You can display child data in either report sections or by using our flexible table control.

The image below shows a report using sub-sections and a table to display hierarchical data.

Printing Paper Size: Letter

PAGE HEADER

ADD REPORT HEADER

## List of Customers Orders by Country

CLICK / DROP HERE

**Country** =Fields.Country

Customers in Country

CLICK / DROP HERE

**Company Name**      **City**

=CompanyName      =City

CLICK / DROP HERE

Order ID	Order Date	Freight	Product Name	Product ID	Quantity	Unit Price	Discount	Line Total
=OrderID	=OrderDate	=Freight	=Product.Product Name	=ProductID	=Quantity	=UnitPrice	=Discount	UnitPrice * Quantity
Order total								=Sum(Quantity*UnitPrice)
=Parent.Parent.CompanyName + "'s totals"								=Sum(Sum(Fields.UnitPrice * Fields.Quantity))

ADD FOOTER

ADD FOOTER

= "page " + Global.CurrentPage + " of " + Global.TotalPages

ADD REPORT FOOTER

AGE FOOTER

For each Customer Group By Country

Using Orders

The image below shows a preview of the report introduced above.

### List of Customers Orders by Country

Austria					
Customers in	2	city/cities			
Customers in Country	2				
Company Name	City				
Ernst Handel	Graz				
Order ID	Order Date	Freight			
10258	7/17/1996	\$140.51			
Product Name	Product ID	Quantity	Unit Price	Discount	Line Total
Chang	2	50	\$15.20	\$0.20	\$608.00
Chef Anton's Gumbo Mix	5	65	\$17.00	\$0.20	\$884.00
Mascarpone Fabioli	32	6	\$25.60	\$0.20	\$122.88
Order total					\$1,755.39
Order ID	Order Date	Freight			
10263	7/23/1996	\$146.06			
Product Name	Product ID	Quantity	Unit Price	Discount	Line Total
Pavlova	16	60	\$13.90	\$0.25	\$625.50
Guaraná Fantástica	24	28	\$3.60	\$0.00	\$100.80
Nord-Ost Matjeshering	30	60	\$20.70	\$0.25	\$931.50
<b>Ernst Handel's totals</b>					<b>\$104,874.98</b>

page 10 of 489

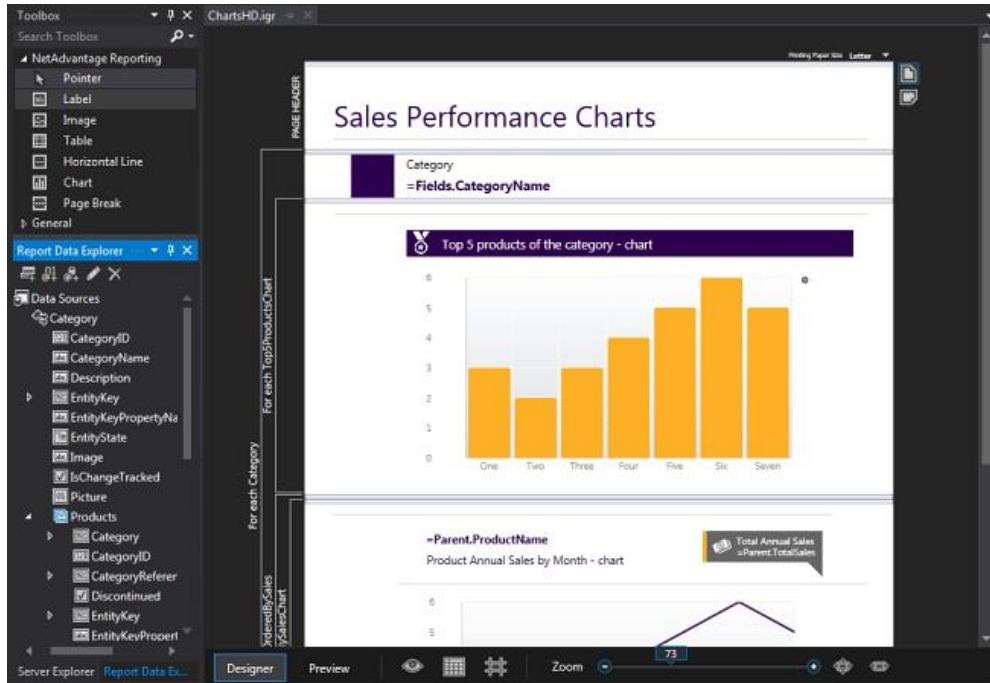
## **Improved Report Engine Performance**

The reporting engine was refactored and optimized to increase performance. As a result, the performance gain is remarkable, especially in scenarios that include reports with many pages (>500). Exceptional performance can be achieved on big reports by enabling parallelism when generating client or server side reports. The report engine also consumes less memory and less disk space when rendering server side reports.

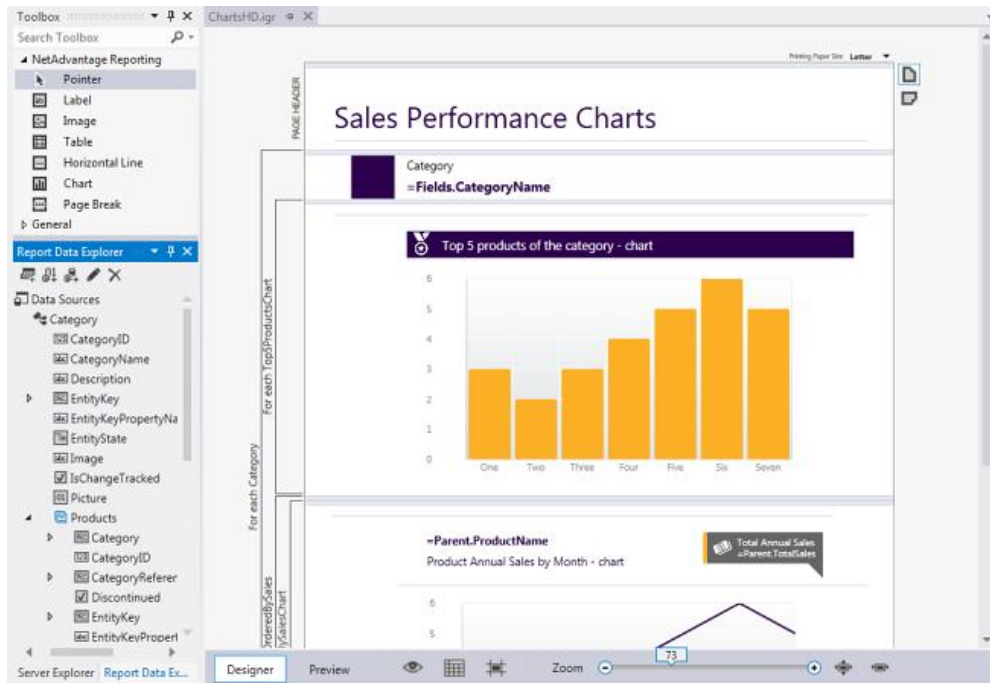
## **Visual Studio 2012 Support – NEW!**

With the release of Visual Studio® 2012, Infragistics continues its dedicated, first class support for the latest version of the Microsoft developer IDE. The report designer has been completely restyled to give you seamless integration and the experience you would expect when working with the newest Visual Studio 2012 IDE. The restyle also implies a complete refresh of the Visual Studio 2010 too. In addition to supporting the new Visual Studio 2012 IDE, we made our reporting designer fully theme aware. When you change your Visual Studio theme, our report designer will change its style accordingly.

The image below shows the report designer with the Visual Studio 2012 dark theme.

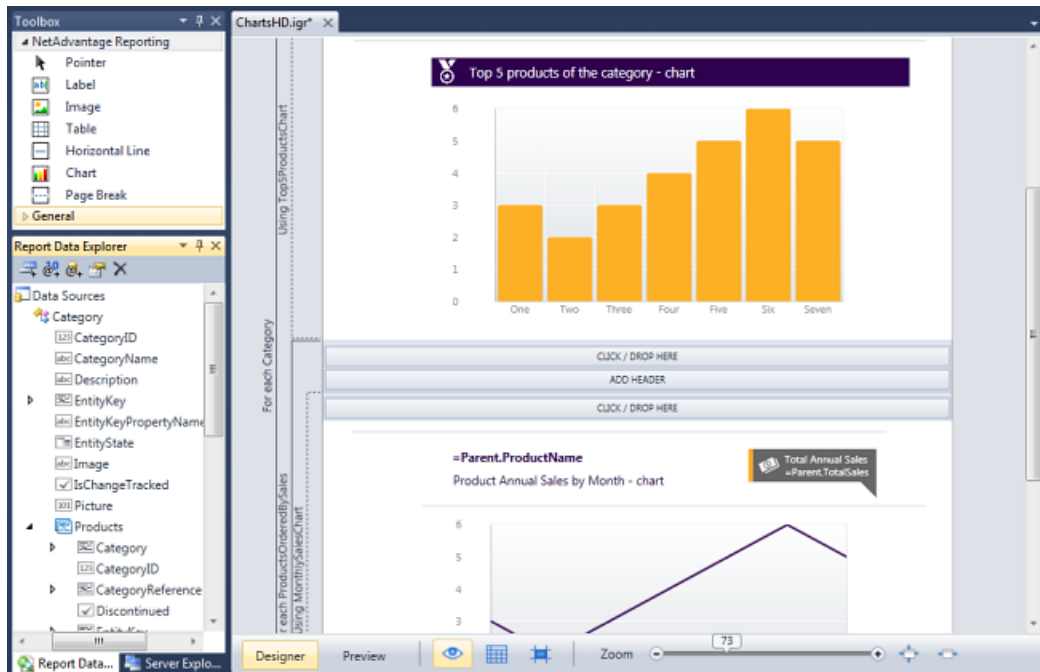


The image below shows the report designer with the Visual Studio 2012 light theme.





The image below shows the report designer in Visual Studio 2010.

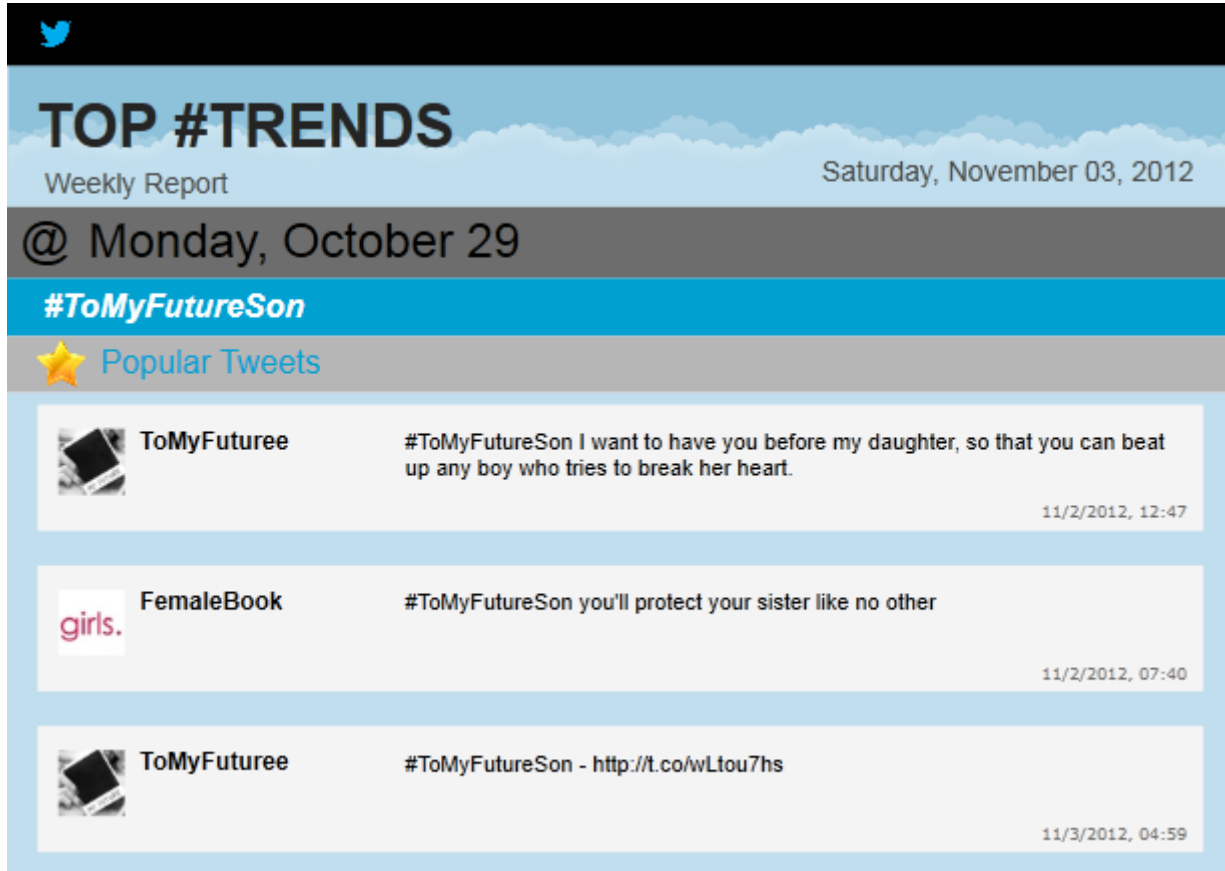


## New Layout Properties




The `RepeatOnEveryPage` property is typically available for header/footer sections of a content region. Since the content region can extend several pages, this property establishes whether the header/footer should appear on each page the content appears. This property is also available for table header/footer rows.

The `CanGrow` property specifies if a control can increase its height if the content requires it. It was previously available in the label control, now its support has been extended to the report sections.

The image below shows a report retrieving information from Twitter's API, which uses the `RepeatOnEveryPage` property. The property is set to `True` for the three content regions included at the top of the report, i.e., the date, trend name, and "Popular Tweets" title.

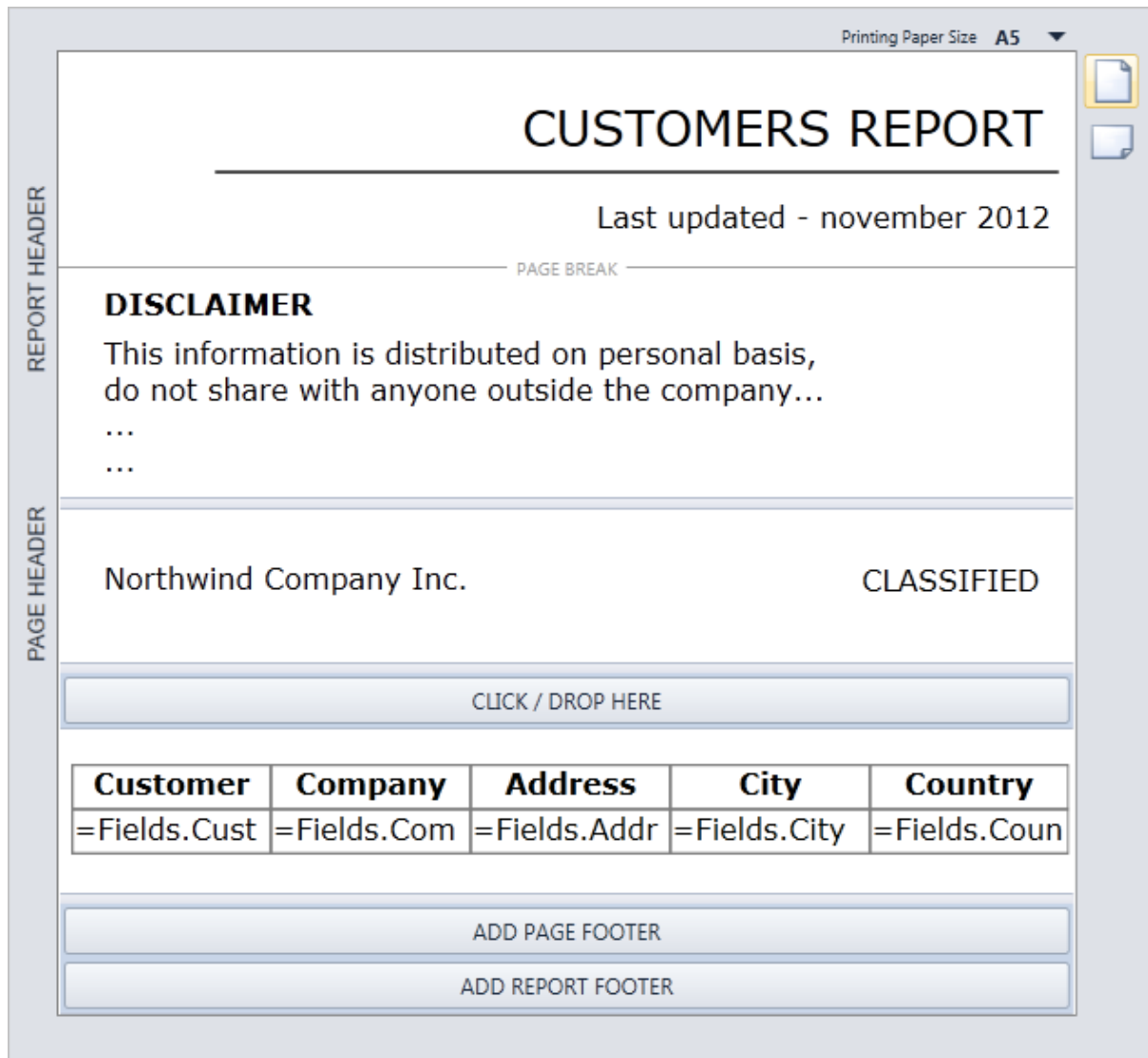


The screenshot displays a Twitter report interface. At the top left is the Twitter logo. The main header reads "TOP #TRENDS" in large bold letters, with "Weekly Report" underneath. On the right side of the header, the date "Saturday, November 03, 2012" is shown. Below the header is a dark grey bar with the text "@ Monday, October 29". A blue bar highlights the trend name "#ToMyFutureSon". Underneath is a grey bar with a yellow star icon and the text "Popular Tweets". The main content area lists three tweets:

Profile Picture	Username	Tweet Text	Timestamp
	ToMyFuturee	#ToMyFutureSon I want to have you before my daughter, so that you can beat up any boy who tries to break her heart.	11/2/2012, 12:47
	FemaleBook	#ToMyFutureSon you'll protect your sister like no other	11/2/2012, 07:40
	ToMyFuturee	#ToMyFutureSon - <a href="http://t.co/wLtou7hs">http://t.co/wLtou7hs</a>	11/3/2012, 04:59

**Page Break Control – NEW!**

You can use a page break control to specify where you want to start a new page within a section. For example, if you want a title page for the report and introductory message printed on separate pages, place a page break in the report header after the controls that you want to appear on the title page and before the controls for the second page.



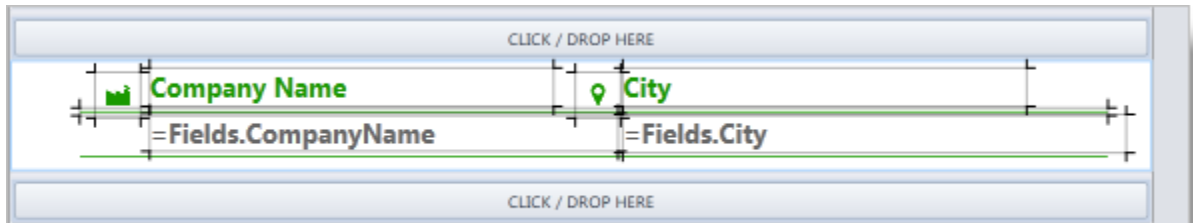
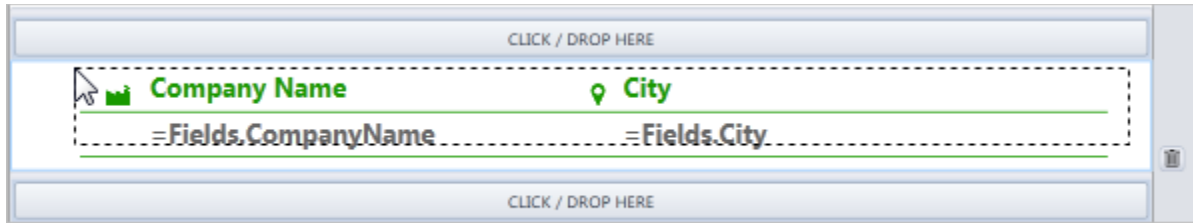
The screenshot shows a report design tool interface. At the top right, it says 'Printing Paper Size A5'. The main content area is titled 'CUSTOMERS REPORT' and includes a horizontal line, followed by 'Last updated - november 2012'. A 'PAGE BREAK' control is positioned below this. The report content is divided into sections: 'REPORT HEADER' (containing the title and update date), 'PAGE HEADER' (containing 'Northwind Company Inc.' and 'CLASSIFIED'), and a table with columns for Customer, Company, Address, City, and Country. Below the table are buttons for 'CLICK / DROP HERE', 'ADD PAGE FOOTER', and 'ADD REPORT FOOTER'. On the left side, there are vertical labels 'REPORT HEADER' and 'PAGE HEADER' indicating the current section being edited.

Customer	Company	Address	City	Country
=Fields.Cust	=Fields.Com	=Fields.Addr	=Fields.City	=Fields.Coun

### Lasso Tool – NEW!

The Lasso Tool simplifies the selection process on the report design surface. It allows you to select multiple objects by drawing a freehand outline around objects similar to using a pencil. Any objects within the lasso outline will become selected.

The images below show an example on how to select labels and images with the Lasso Tool.



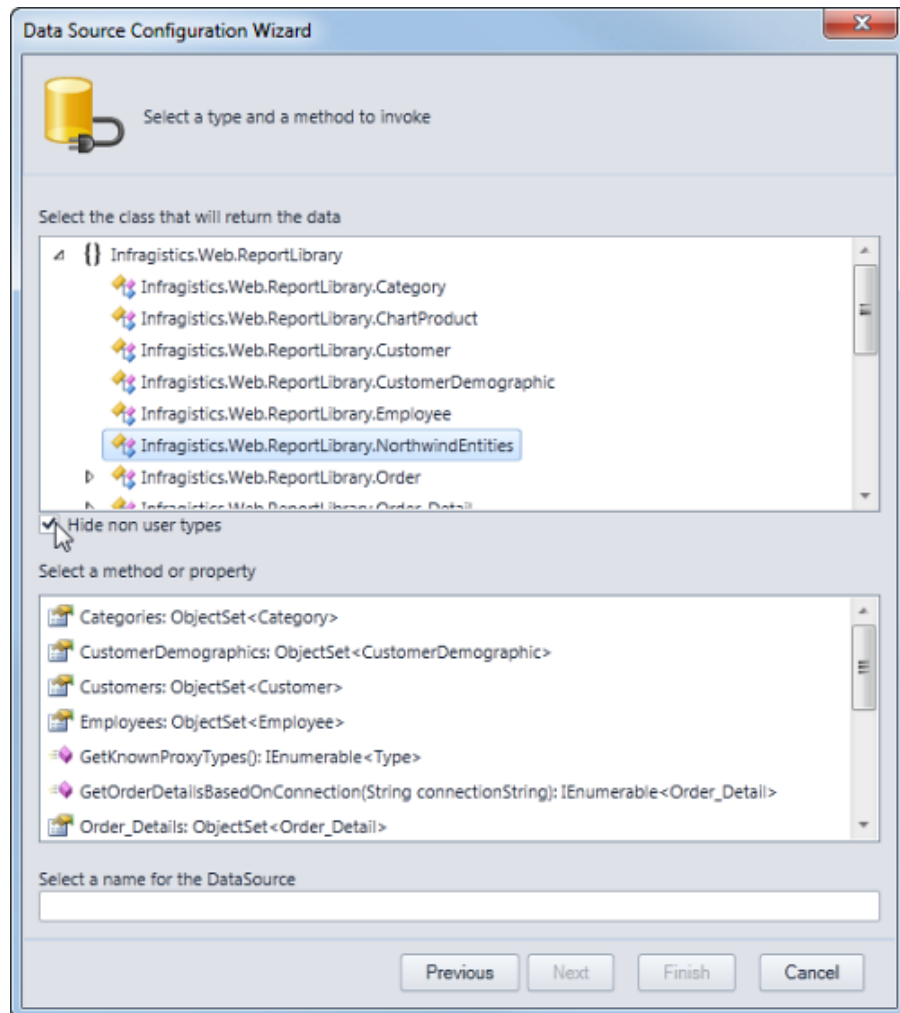
## Changes/Additions for 2012 Volume 2

### Overview

The following table summarizes all changes for 2012 Vol 2. Additional details are available below.

Change	Description
Report Migration	Automatic and programmatic conversion of reports.
Object Data Source Wizard default types	Better control of the types shown by the object data source wizard.
Units Customization	Default units can be customized.
Expression Assistant update	Several usability enhancements have been done in the Expression Assistant.
Expression Language	The expression language has been extended.
Reporting Configurations	The usage of parallelism in the report engine can be enabled/disabled through configuration files.
Object Data Source	New types supported for creating object data sources.
ContentBackground Property	Set the background color of the report content area.
Query timeout for SQL data sources	Set the timeout for SQL data source queries.

<p><b>Report Migration</b></p>	<p><b>Runtime Report Migration</b></p> <p>When loading a report from a previous version using the Report Viewer, the report will be automatically converted to the current version installed before rendering it. Since the migration process is done on the fly, it will slightly affect the performance every time you render an old report. In order to avoid this overhead you can migrate the report in advance, using Visual Studio Reporting plugin or the Migration API</p> <p><b>Migration API</b></p> <p>Reports can be migrated programmatically, using our new migration API. To do this, you need to use the <i>Infragistics.Reports.ReportMigrator</i> class, available both in Silverlight and WPF through the following assemblies:</p> <ul style="list-style-type: none"> <li>• <i>InfragisticsSL5.Reports.v12.2</i></li> <li>• <i>InfragisticsWPF4.Reports.v12.2</i></li> </ul> <p>Here there is a practical example:</p> <pre>using (var inputStream = File.OpenRead(@"C:\&lt;PathToTheReport&gt;\ReportOld.igr")) {     using (var outputStream = File.OpenWrite(@"C:\&lt;PathToTheReport&gt;\ReportNew.igr"))     {         ReportMigrator.Migrate(inputStream, outputStream);     } }</pre>
<p><b>Object Data Source Wizard default types</b></p>	<p>When using the Object Data Source Wizard, classes belonging to assemblies from .NET framework (i.e., all assemblies under <i>System</i> namespace) are now hidden by default. If you need to use one of these classes, uncheck the option <i>Hide non user types</i>.</p>



## Units Customization

NetAdvantage Reporting supports different measure units common to all elements within the Report Designer. Now you can configure the default values for each available measure type through the *Visual Studio Options* panel (under **Infragistics NetAdvantage Reporting > General**).

Available types are:

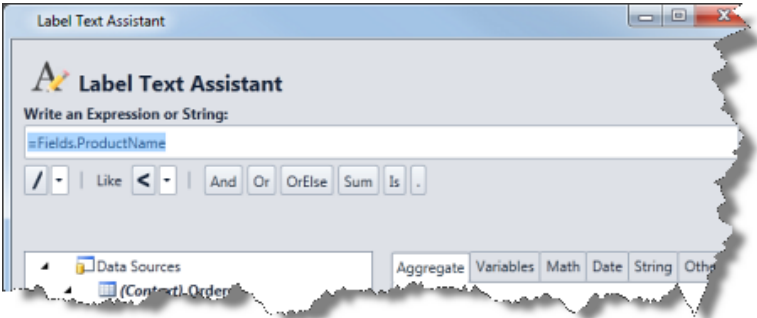
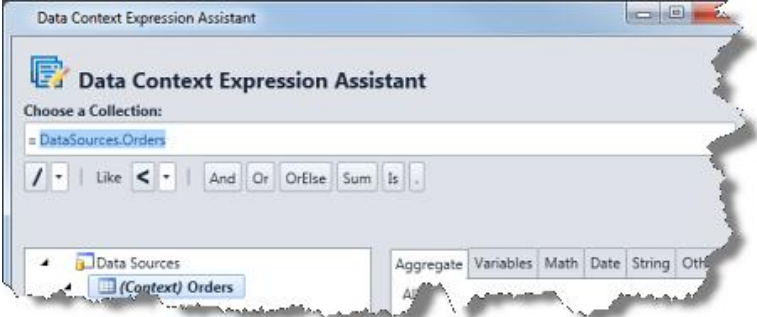
Measure type	Where it is used	Default unit of measure
<b>Paper size and margins</b>	The paper size and report margins.	Inches (in) or centimeters (cm), depending on the region.
<b>Item bounds</b>	The position and size of each control in the report.	Inches (in) or centimeters (cm), depending on the region.
<b>Thickness</b>	Line thicknesses, borders and padding.	Points (pt).
<b>Font sizes</b>	The font size used in labels and other items containing text.	Points (pt).



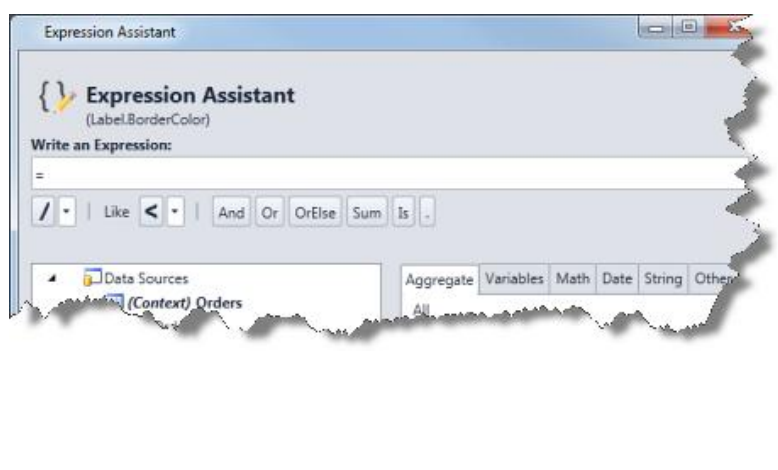
**Expression Assistant update**

**Assistant awareness**

The Expression Assistant now reflects the context where the assistant is invoked, so you can always have a clear idea of what you are editing.

Opening from...	Looks like...
Label	 <p>The screenshot shows the 'Label Text Assistant' dialog box. The title bar reads 'Label Text Assistant'. The main area contains the text 'Label Text Assistant' and 'Write an Expression or String:'. Below this, the expression '=Fields.ProductName' is entered in a text field. A toolbar below the text field includes a slash icon, a 'Like' button, a left arrow, and buttons for 'And', 'Or', 'OrElse', 'Sum', and 'Is'. At the bottom, there is a tree view showing 'Data Sources' and '(Context) Orders', and a row of buttons for 'Aggregate', 'Variables', 'Math', 'Date', 'String', and 'Other'.</p>
Data Context	 <p>The screenshot shows the 'Data Context Expression Assistant' dialog box. The title bar reads 'Data Context Expression Assistant'. The main area contains the text 'Data Context Expression Assistant' and 'Choose a Collection:'. Below this, the expression '=DataSources.Orders' is entered in a text field. A toolbar below the text field includes a slash icon, a 'Like' button, a left arrow, and buttons for 'And', 'Or', 'OrElse', 'Sum', and 'Is'. At the bottom, there is a tree view showing 'Data Sources' and '(Context) Orders', and a row of buttons for 'Aggregate', 'Variables', 'Math', 'Date', 'String', and 'Other'.</p>

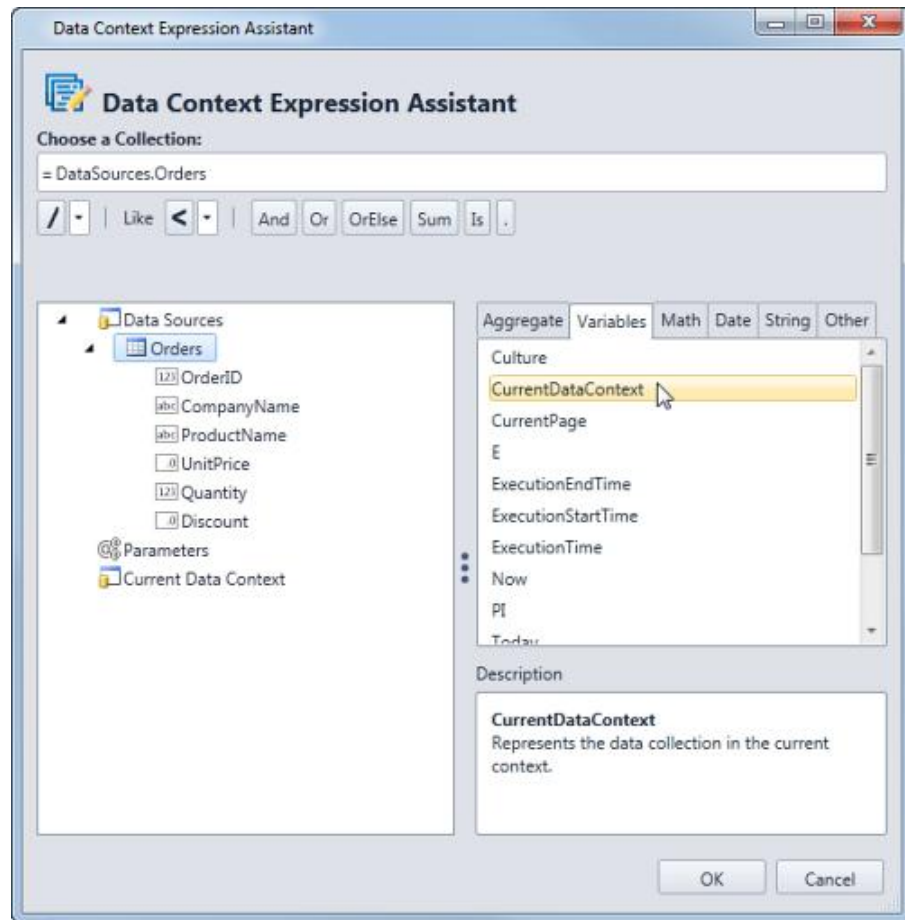
Other bindings



### Assistant Revamp

The expression assistant went through several changes in order to improve its usability, in particular:

- The Data Sources tree has now more space, in order to facilitate the navigation through the data source structure. Parameters are also available in the tree now as they are in the Report Data Explorer.
- A splitter has been added so you can control how much space is available for the tree and the functions/variable Tab control.
- When selecting elements from the tree or the function/variable list, a preview of the resultant expression is shown. Double-clicking an expression makes it permanent.



## Expression language

### Data Sources keyword added to the language.

To reference a data source in the expression language, you must now specify it using: `DataSources.name` where *name* is the data source's name. When migrating old reports to 12.2, expressions including references to data sources will be automatically updated.

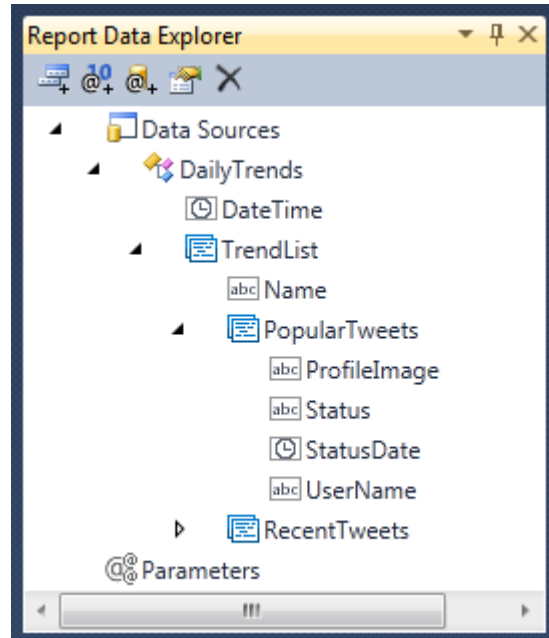
### CurrentDataContext

The Current Data Context node was added to the Data Sources and Parameters area of the Expression Assistant. `CurrentDataContext` represents the current context's data collection.

### Referencing hierarchical data

When working with hierarchical data, you can access the data fields located in the upper level of the hierarchy by using the keyword `Parent`. In addition, you can access multiple levels upwards as shown in the example below.

For this example we will work with a hierarchical data source with the following hierarchy: *DailyTrends > TrendList > Popular Tweets*. As shown in the image below.



To access the DateTime field located in the top level (DailyTrends) from the PopularTweets child, you have to use the following expression:  
`Parent.Parent.DateTime`

**New functions**

- *Any( boolExpression [, scopeExpression] )*  
 Indicates if any element in the collection satisfies boolExpression in the scopeExpression, or in the current data context if scopeExpression is omitted.
- *All( boolExpression [, scopeExpression] )*  
 Indicates if all elements in the collection satisfy the boolExpression in the scopeExpression, or in the current data context if scopeExpression is omitted.
- *CountIf( boolExpression [, scopeExpression] )*  
 Returns the total amount of elements in the collection that satisfy the boolExpression in the scopeExpression, or in the current data context if scopeExpression is omitted.

**Extended Scope – Aggregate functions.**

All aggregate functions can now be applied not only to data sources but also to any collection. The scopeExpression can be used to pass the data source or collection. If no collection is passed the current data context is used.

**Count syntax has changed**

First parameter deprecated.

Before	Now
<i>Count( expression [, scopeExpression] )</i>	<i>Count( [scopeExpression] )</i>

## Reporting Configurations

### Report Rendering Process

The report rendering process was redesigned to take advantage of multithreading and multicore environments allowing to reduce processing time. Parallelism can be enabled/disabled through the application configuration file (i.e. web.config/app.config) or by passing a parameter to the application (in the case of Silverlight).

#### On Client-Side Rendering

WPF and Winforms Applications	Silverlight
<ul style="list-style-type: none"> <li>Configured in the app.config file.</li> <li><b>Enabled</b> by default.</li> </ul>	<ul style="list-style-type: none"> <li>Configured by passing a parameter to the Silverlight object.</li> <li><b>Enabled</b> by default.</li> </ul>

#### On Server-Side Rendering

- Configured in the web.config.
- Disabled** by default.

### Text Measurement Process

This process was redesigned (excluding silverlight client-side rendering) to take advantage of multicore and multithread environments in order to reduce text measuring and layout processing. Parallelism in text measurement can be enabled/disabled through the application configurations file (i.e. web.config/ app.config).

#### On Client-Side Rendering

WPF and Winforms Applications	Silverlight
<ul style="list-style-type: none"> <li>Configured in the app.config file.</li> <li><b>Enabled</b> by default.</li> </ul>	<ul style="list-style-type: none"> <li><b>Not Available</b></li> </ul>

#### On Server-Side Rendering

- Configured in the web.config
- Disabled** by default.

### Code Sample (Web.Config/App.Config)

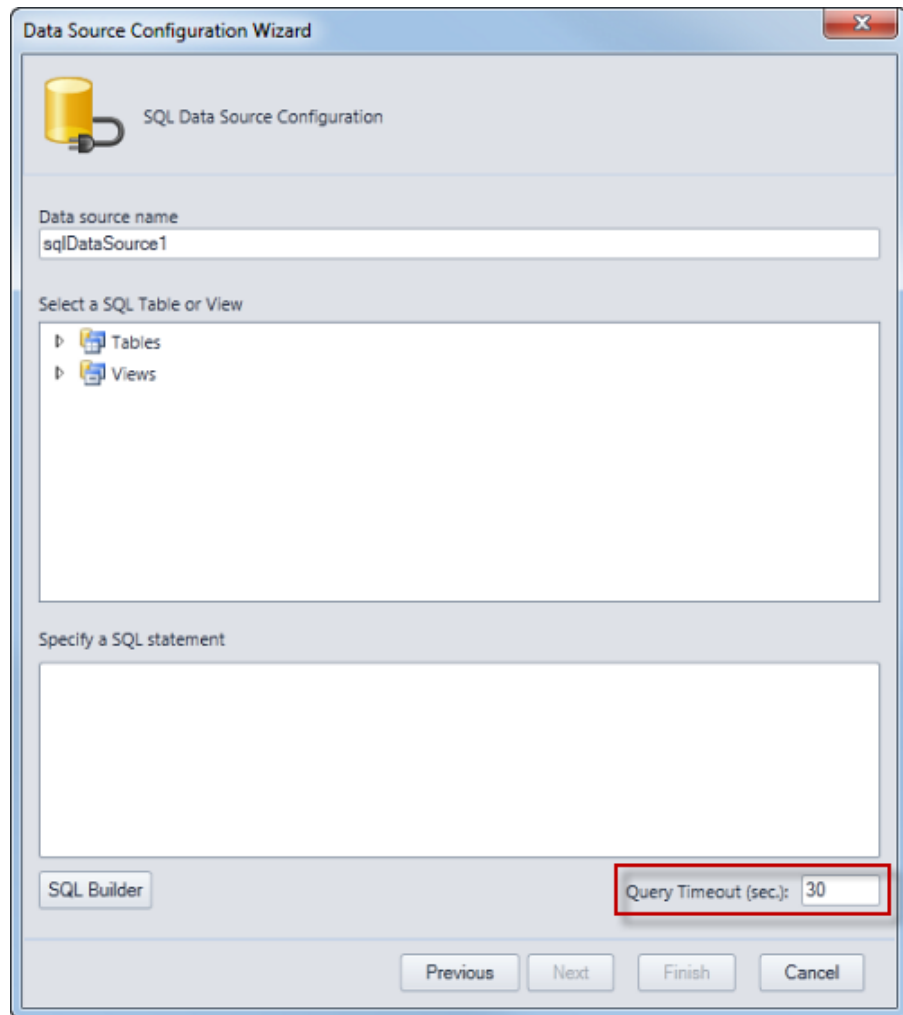
```
<configuration>
  <configSections>
    <section name="infragistics.reports"
      type="Infragistics.Reports.ReportsConfigurationSection, InfragisticsWPF4.Reports.Client.v12.2 ..." />
  </configSections>
  <infragistics.reports>
    <engineSettings useParallelProcessing="false"
      useParallelMeasuring="false" />
    ...
  </infragistics.reports>
  ...
</configuration>
```

### Code Sample (Silverlight Client-Side Rendering)

```
<html>
<head>
</head>
<body>
  <form id="form1" runat="server" style="height:100%">
    <div id="silverlightControlHost">
      <object data="data:application/x-silverlight-2,"
        type="application/x-silverlight-2" width="100%" height="100%">
        <param name="initparams"
          value="infragisticsReportingUseParallelProcessing=False"/>
        <param name="source"
          value="ClientBin/SLTest.xap"/>
        <param name="onError" value="onSilverlightError" />
      />
      <param name="background" value="white" />
      <param name="minRuntimeVersion"
        value="5.0.61118.0" />
      <param name="autoUpgrade" value="true" />
      <a
        href="http://go.microsoft.com/fwlink/?LinkId=149156&v=5.0.61118.0"
        style="text-decoration:none">
        
        </a>
      </object><iframe id="_sl_historyFrame"
        style="visibility:hidden;height:0px;width:0px;border:0px"></iframe>
    </div>
  </form>
</body>
</html>
```

<p><b>Object Data Source</b></p>	<p><b>Extended Supported Types</b></p> <p>Now Object Data Sources can be defined from:</p> <ul style="list-style-type: none"> <li>• Arrays (e.g. Customer[])</li> <li>• IEnumerable of a generic type (e.g. IEnumerable&lt;Tuple&lt;int,string&gt;&gt;)</li> </ul>
<p><b>ContentBackground Property</b></p>	<p><b>Content Area Background</b></p> <p>This property allows you to set the background color of the content area for the report. The content area is the page area within the defined margins. This property is available in the PageSettings of the Report.</p>
<p><b>Query timeout for SQL data sources</b></p>	<p>You can configure the timeout for SQL query execution in SQL data sources using the SQL Data Source Configuration wizard. Timeout is specified in seconds and its default value is 30 seconds. A timeout value of 0 indicates no limit, i.e. attempts to execute the query will wait for a response indefinitely.</p>





When trying to render a report and the waiting time set for query timeout is exceeded, an error message is displayed.

## Bug Fixes

Component	Product Impact	Description
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	Cascading parameters issue. When ordering parameters in the Viewer's parameter pane, cascade parameters precedence was not correctly taken into account.
<b>NetAdvantage Reporting (Designer)</b>	Bug Fix	Dynamic parameter over internal property issue. When rendering a report with a dynamic parameter based on an internal property of an Object Data Source, the value in the Viewer's parameter pane was not displayed as expected.
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	Silverlight printing incorrect error message issue. When invoking printing in Silverlight and working server-side, an incorrect error message was shown when the session has expired.
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	Chart axis background color HTML5 issue. When the background color of the Chart Axis is expressed using #XXXXXXXX format, it wasn't correctly rendered in the HTML5 viewer.
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	Some theme roller themes cause layout issues. When some themes in the theme roller (e.g. Pepper Grinder) were applied to the HTML5 Viewer, the toolbar buttons were not correctly aligned.
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	WPF/WinForms viewers SSL configuration issues. WPF and WinForms viewers fail to render server-side reports over SSL, giving the error: "The provided URI scheme 'https' is invalid; expected 'http'".
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	jQueryUI specific version issue. The HTML5 Viewer was not working properly with jQueryUI v1.9.0 version

<b>NetAdvantage Reporting (Table)</b>	Bug Fix	Table cell delete issue. When a table cell was deleted using del/supr, the cell format was lost.
<b>NetAdvantage Reporting (Viewer)</b>	Bug Fix	HTML5 Viewer is not working properly on iOS 6.0. iOS 6.0 Safari caches all POST requests by default, which caused several issues in the HTML5 viewer.