

Crystal Reports for Visual Studio .NET

Printing Web based reports with Crystal Reports for Visual Studio
.NET

Overview

This white paper describes three scenarios available for printing reports that are displayed over the Web using Crystal Reports for Visual Studio .NET.

Contents

INTRODUCTION	2
PRINTING CRYSTAL REPORTS IN ASP.NET APPLICATIONS	2
<i>Exporting the report to a printer-friendly format</i>	<i>2</i>
When would I want to use this method for Printing?.....	3
<i>Printing the report on the server side (PrintToPrinter method)</i>	<i>4</i>
When would I want to use this method for Printing?.....	5
<i>Printing the report on the client side via Internet Explorer</i>	<i>5</i>
When would I want to use this method for Printing?.....	6
FINDING MORE INFORMATION	6

Introduction

When viewing reports over the Web in the Web Forms viewer, users are not given the option to print an entire report in the viewer's toolbar. The reason for this is that the Web Forms Viewer control emits static HTML3.2 or HTML4.0 script in the client's browser providing a true Zero Client solution for viewing reports. A limitation of static HTML is that printing is limited to printing capabilities of the browser resulting in the ability to print the currently displayed page only.

It is not uncommon for users to wish to print entire reports to a printer and this paper offers the following solutions to printing an entire multi-page report accessed over the Web:

- Exporting the report to a format that can be printed by a client's existing application.
- Printing the report on the server side using the "PrintToPrinter" method.
- Printing the report on the client side via the Internet Explorer browser.

Printing Crystal Reports in ASP.NET Applications

In order to be able to print an entire Crystal report at once to a client's printer, client-side code must be executed. For example, Adobe Acrobat and Microsoft Word print entire documents by supplying a client-side control that can access the entire document and print it. Crystal Reports offers the ability to export reports to several different formats that can be read by other applications.

Exporting the report to a printer-friendly format

The most popular method of client-side report printing is method of first exporting the report to a printer-friendly format (usually Portable Document Format (PDF) or Microsoft Word/ Rich Text Formats).

Essentially, the report is exported to a format that can be recognized and used by a control installed on the client. The report is exported on the Web server and the exported file is then streamed down to the client's browser. Most clients will have an application installed on the client computer that can read these formats and print the report.

The following VB (Visual Basic) code demonstrates exporting a report to PDF format and streaming it to the client's browser to be displayed as a PDF document.

```
' Define Crystal Reports variables
```

```
Dim crReportDocument As ReportDocument
Dim crExportOptions As ExportOptions
Dim crDiskFileDestinationOptions As DiskFileDestinationOptions
Dim FName as String
```

```
' The following code can be placed directly after the call to
' InitializeComponent() in the form's constructor, or inside of
' a Button_Click event where the button is used by the client to
' get a printable copy of the report.
```

```
CrReportDocument = New ReportDocument()
' The following line of code loads the sample report "Chart.rpt" that installs
' with Visual Studio .NET
CrReportDocument.Load ("C:\Program Files\Microsoft Visual Studio .NET\Crystal
Reports\Samples\Reports\Feature Examples\Chart.rpt")
```

```
Fname = "c:\exports\" & Session.SessionID.ToString & ".pdf"
CrDiskFileDestinationOptions = New DiskFileDestinationOptions()
CrDiskFileDestinationOptions.DiskFileName = FName
CrExportOptions = crReportDocument.ExportOptions
With crExportOptions
    .DestinationOptions = crDiskFileDestinationOptions
    .ExportDestinationType = ExportDestinationType.DiskFile
    .ExportFormatType = ExportFormatType.PortableDocFormat
```

```
End With
CrReportDocument.Export()
' The following code writes the pdf file to the Client's browser.
Response.ClearContent()
Response.ClearHeaders()
Response.ContentType = "application/pdf"
Response.WriteFile(Fname)
Response.Flush()
Response.Close()
```

```
'delete the exported file from disk
System.IO.File.Delete(Fname)
```

When would I want to use this method for Printing?

Exporting a report to a printer-friendly format is useful to Web users who need the ability to print multiple page reports.

You may wish to use the Web Forms viewer control to display the report on the client and provide a button on the form that will perform the

export. This will give the user the ability to utilize the interactivity (Drill-Down, etc.) of the Crystal report format within the viewer control and export the report to the more printer-friendly format at will.

Printing the report on the server side (PrintToPrinter method)

Server side printing results in the report being generated on the Web Server and being printed to a printer that is installed to the Web server.

Printing to a printer installed to the Web server requires that the process under which ASP.NET is running have sufficient permissions to access Network printers. In order to provide these permissions, changes to the account that the ASPNET worker process runs under are necessary as well as changes to the registry on the Web server.

By default, ASP.NET runs with the permissions of the local "machine" (ASPNET account for the ASPNET worker process) account. In order to achieve network printing through IIS (Internet Information Services), the Framework should be run under the local SYSTEM account. To make this change, it is necessary to edit the default configuration settings in the Machine.config file. This file is located by default at:

```
C:\WINNT\Microsoft.NET\Framework\<>VersionNumber>\CONFIG\
```

Where <VersionNumber> is the version of the .NET Framework installed to the computer. The account setting is contained within the <ProcessModel> tag of this file. The default setting for the account is:

```
userName="machine" password="AutoGenerate"
```

This can be changed to:

```
userName="SYSTEM" password="AutoGenerate"
```

Then save the file.

It is now possible to give the local SYSTEM account on the server permissions to print. If the printer is a network printer, we must expose the network printer to the local SYSTEM account. The reason behind this is that when a printer is installed to a computer, the settings for this printer are stored in the registry under the HKEY_CURRENT_USER registry hive (Registry file). The IIS process will never use this registry hive since it always runs under the context of the local SYSTEM account and cannot be logged onto a server as a "Current User". By default, the SYSTEM account does not have any printers set up in the registry. There are two Microsoft knowledge base articles, 152451 and 184291 that explain how to expose the printer settings to the System account. Both of these articles require editing the registry on the Web server and copying the printer settings from the HKEY_CURRENT_USER hive to the HKEY_USERS/.DEFAULT hive.

NOTE	For information about how to edit the registry, view the "Changing Keys and Values" Help topic in Registry Editor (Regedit.exe) or the "Add and Delete Information in the Registry" and "Edit Registry Data" Help topics in Regedt32.exe. Note that you should back up the registry before you edit it. If you are running Windows NT or Windows 2000, you should also update your Emergency Repair Disk (ERD).
-------------	---

Once these changes have been made, it will be necessary to stop and start IIS to apply the permissions changes made to the ASPNET worker process.

When would I want to use this method for Printing?

Server side printing is most useful in an Intranet environment where the printers installed to the server are connected to and accessible over the network. Clients can be given the option to select a printer that is near their location and allow the printing to be handled by the server. The requirements here are that the printers be installed to the Web server, the local SYSTEM account is given access to the printers and the Machine.config file holding configuration settings for the .NET framework is edited to give the ASPNET worker process access to the system settings on the server. Keep in mind that using this method involves changing the default permissions on the server and is not recommended for applications that are exposed to the Internet.

Printing the report on the client side via Internet Explorer

This method of printing a report is useful in environments where client computers do not have the ability to print using client side controls such as the Adobe Acrobat reader or Microsoft Word.

This method displays an entire report in the Web Forms viewer control as one continuous page such that using the printing capabilities of the browser can be used. A button can be provided on the main page displaying the report in the Web forms viewer that will redirect the client to a new Web form that displays the entire report as one continuous page in the browser. The Viewer on the second form can be formatted such that the group tree and toolbar are disabled and the SeparatePages property of the viewer is set to False. A sample of how to set these properties is below:

```
crReportDocument = New Income_Statement()  
CrystalReportViewer1.ReportSource = crReportDocument  
CrystalReportViewer1.SeparatePages = False  
CrystalReportViewer1.DisplayGroupTree = False  
CrystalReportViewer1.DisplayToolbar = False
```

Limitations using this method include the possibility of page breaks not occurring in the desired locations

When would I want to use this method for Printing?

This method of printing a Web based report would be useful in an environment where the client does not have access to the applications required to one of the exported document formats (i.e. Adobe Acrobat Reader, Microsoft Word, and Microsoft Excel). The main limitation of this method is that the positioning of page headers, footers and page breaks cannot be controlled.

Finding more information

For more information and resources, refer to the product documentation and visit the support area of the web site at:

<http://www.businessobjects.com/>

NOTE	<p>To highlight text in a PDF document for copying and pasting code, click the Text Select Tool toolbar button in Adobe Acrobat.</p>  <p>This procedure applies to Adobe Acrobat 4.0 and 5.0.</p>
-------------	--

► www.businessobjects.com

No part of the computer software or this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from Business Objects.

The information in this document is subject to change without notice. Business Objects does not warrant that this document is error free.

This software and documentation is commercial computer software under Federal Acquisition regulations, and is provided only under the Restricted Rights of the Federal Acquisition Regulations applicable to commercial computer software provided at private expense. The use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at 252.227-7013.

The Business Objects product and technology are protected by US patent numbers 5,555,403; 6,247,008; 6,578,027; 6,490,593; and 6,289,352. The Business Objects logo, the Business Objects tagline, BusinessObjects, BusinessObjects Broadcast Agent, BusinessQuery, Crystal Analysis, Crystal Analysis Holos, Crystal Applications, Crystal Enterprise, Crystal Info, Crystal Reports, Rapid Mart, and WebIntelligence are trademarks or registered trademarks of Business Objects SA in the United States and/or other countries. Various product and service names referenced herein may be trademarks of Business Objects SA. All other company, product, or brand names mentioned herein, may be the trademarks of their respective owners. Specifications subject to change without notice. Not responsible for errors or omissions.

Copyright © 2005 Business Objects SA. All rights reserved.