

# **SQL Server 2005 Reporting Services Developer 1 Workshop**

Vision Harvest, Inc.

888.236.8087  
sales@visionharvest.com

Revised: June 30, 2006  
Part: VISI100501E  
Version 1.1

©2006 Vision Harvest, Inc.

ALL RIGHTS RESERVED

This course covers Microsoft SQL Server 2005 Reporting Services

No part of this manual may be copied, photocopied, or reproduced in any form or by any means without permission in writing from the Author - Vision Harvest, Inc. All other trademarks, service marks, products or services are trademarks or registered trademarks of their respective holders.

This course and all materials supplied to the student are designed to familiarize the student with the operation of the software programs. We urge each student to review the manuals provided by the software publisher regarding specific questions as to the operation of the programs. THERE ARE NO WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, MADE WITH RESPECT TO THESE MATERIALS OR ANY OTHER INFORMATION PROVIDED TO THE STUDENT. ANY SIMILARITIES BETWEEN FICTITIOUS COMPANIES, THEIR DOMAIN NAMES, OR PERSONS WITH REAL COMPANIES OR PERSONS IS PURELY COINCIDENTAL AND IS NOT INTENDED TO PROMOTE, ENDORSE OR REFER TO SUCH EXISTING COMPANIES OR PERSONS.

VISION HARVEST, INC. NOR ITS RESPECTIVE PRODUCTS OR SERVICES ARE AFFILIATED WITH, OR ENDORSED, LICENSED, OR SPONSORED BY MICROSOFT, THE OWNER OF SQL SERVER 2005 REPORTING SERVICES.



This courseware has been developed by the professional team at Vision Harvest, Inc., [www.visionharvest.com](http://www.visionharvest.com). Questions, comments or concerns about courseware content should be sent to [sales@visionharvest.com](mailto:sales@visionharvest.com) or 888.236.8087.

## Table of Contents

<b>Introduction</b> .....	<b>1</b>
Introduction Objectives.....	2
About SQL Server Reporting Services.....	2
Training Philosophy.....	3
Class Objectives .....	3
About This Manual .....	4
Tips, Notes, and Warnings.....	5
<b>Lesson 1: The Big Picture</b> .....	<b>7</b>
Lesson Objectives.....	8
SQL Server 2005 Reporting Services Features.....	9
The Reporting Services Key Components .....	10
Client Components .....	11
Web Components.....	12
Report Server Components.....	13
Database Components.....	14
Report Design Options.....	15
Visual Studio 2005 and Development Studio .....	15
Data Processing Extensions .....	17
Included Extensions .....	17
Custom Data Processing Extensions .....	17
Deployment Options.....	18
Key Considerations .....	18
Distribution Options.....	19
Rendering Extensions .....	19
<b>Lesson 2: The Report Wizard</b> .....	<b>23</b>
Lesson Objectives.....	24
Starting SQL Server Business Intelligence Development Studio.....	25
Creating a New Report Project .....	26
Welcome to the Report Wizard Dialog .....	29
Select the Data Source Dialog .....	30
Design the Query Dialog .....	33
Select the Report Type Dialog.....	35
Design the Table or Design the Matrix Dialog .....	36
Choose the Table Layout .....	39
Choose Table Style or Matrix Style Dialog .....	40
Completing the Wizard Dialog .....	41
Understanding What Just Happened .....	42
Previewing Reports.....	44
Publishing Reports .....	45
Accessing Reports via the Report Manager.....	48

<b>Lesson 3: The Report Designer</b> .....	<b>53</b>
Lesson Objectives.....	54
The Main Components of the Design Environment .....	55
Menus.....	55
Toolbars.....	55
Windows .....	56
Usability Features.....	57
Creating a Data Source and Dataset .....	58
Create a Data Source.....	59
Create a Dataset .....	61
Query tab.....	62
Fields tab.....	63
Data Options.....	63
Parameters .....	64
Filters.....	64
Using the Graphical Query Designer .....	66
Designing the Layout .....	68
The Layout View .....	69
The Toolbox .....	70
Adding a Table Data Region .....	72
Adding Data Fields to a Table Data Region.....	73
A Closer Look at Preview.....	76
Adding Textboxes .....	78
Changing Columns and Rows.....	78
Adding or Deleting.....	79
Resizing.....	80
Moving.....	80
Table Data Region Record Sorting .....	82
Using Expressions .....	83
Using the Expression Editor Dialog.....	84
Grouping Data.....	90
General.....	91
Filters.....	92
Sorting .....	92
Visibility.....	92
Data Output.....	92
Summary Calculations .....	94
<b>Lesson 4: Query and Report Parameters</b> .....	<b>97</b>
Lesson Objectives.....	98
Query vs. Report Parameters .....	99
Create a Report Query.....	99
Add Report Parameter to a Dataset.....	101
Previewing a Report with Parameters.....	103
Displaying Parameters in a Report .....	104
Dynamic Parameter Values (From Query).....	105

<b>Lesson 5: Formatting Reports</b> .....	<b>109</b>
Lesson Objectives.....	110
Report Item Style Properties.....	111
Quick Formatting using Toolbar and Menu Options.....	114
Report Formatting Toolbar.....	114
Report Borders Toolbar.....	114
Formatting Textboxes with the Textbox Properties Dialog.....	116
General.....	117
Visibility.....	118
Navigation.....	118
Font.....	119
Format.....	119
Interactive Sort.....	120
Data Output.....	120
Managing Tables with the Table Properties Dialog.....	123
General.....	124
Visibility.....	125
Navigation.....	125
Sorting.....	125
Groups.....	126
Filters.....	126
Data Output.....	127
Setting the Report Properties.....	128
General.....	129
Layout.....	129
Code.....	129
Data Output.....	130
Drawing Lines and Rectangles.....	131
<b>Lesson 6: Administration</b> .....	<b>135</b>
Lesson Objectives.....	136
Managing Reporting Services.....	137
Using the Report Manager.....	138
Settings.....	139
Security.....	140
Other.....	141
SQL Server Management Studio Overview.....	142
The Management Studio Object Explorer.....	144
Managing Objects in Object Explorer.....	145
Server Properties.....	145
Web Folder Properties.....	146
Data Source Properties.....	150
Shared Schedules.....	151
Using the Reporting Services Configuration Manager.....	153
Tools for Managing Reports.....	157
Deploying Reports with Development Studio.....	158
Linked Reports.....	161
Managing Parameters.....	163

---

<b>Lesson 7: Subscriptions, Snapshots and Caching .....</b>	<b>167</b>
Lesson Objectives .....	168
Understanding Subscriptions .....	169
Creating a Subscription .....	170
Static Subscription .....	170
Data-driven Subscription .....	174
Understanding Snapshots .....	179
History Properties .....	179
Understanding Caching .....	182
Execution Properties .....	182
<b>Lesson 8: Security .....</b>	<b>185</b>
Lesson Objectives .....	186
Security Model Concepts .....	187
Understanding System-level Roles .....	187
Adding System-level Roles .....	189
Understanding Item-level Roles .....	190
Adding an Item-Level Role .....	193
Understanding Users and Groups .....	195
Understanding Security Settings .....	196
Default Folder Structure .....	197
Current Classroom Folder Structure .....	198
Adding Groups and Users .....	198
Adding Groups to Reporting Services .....	202
Adding Users .....	207
Auditing and Management Reports .....	208
Execution Log Reports .....	208
Server Management Reports .....	209
Installing Auditing and Management Reports .....	210
Deploying the Auditing and Management Reports .....	211
Deploy Sample Reports to Report Server .....	211
Create a Database for Report Execution Data .....	212
Configure and Run the Job Package .....	213
Run the Sample Reports .....	216
<b>Lesson 9: Building Report Models .....</b>	<b>219</b>
Lesson Objectives .....	220
Understand the Report Model Architecture .....	221
Creating a Report Model Project .....	222
Creating a Report Model Data Source .....	223
Creating a Data Source View .....	224
Modifying the Data Source View .....	226
Toolbar .....	226
Diagram Pane .....	227
Diagram Organizer Pane .....	228
Tables Pane .....	228
Creating a Report Model File .....	229
Report Model Generation Rules .....	229
Model Statistics .....	230
Understanding the Model Designer .....	231
Model Designer Layout .....	231

Entities .....	231
Attributes .....	232
Identifying Field .....	232
Roles .....	232
Entity and Field Folders .....	232
Perspectives .....	233
Deploying a Report Model.....	237
<b>Lesson 10: Using Report Builder .....</b>	<b>241</b>
Lesson Objectives.....	242
Understanding Report Builder.....	243
Launching Report Builder.....	243
Report Builder Design Layout .....	244
Getting Started Pane .....	244
Report Toolbar.....	246
Explorer Pane.....	247
Design Area.....	248
Building a Table Report.....	248
Adding Fields to the Table Data Region.....	248
Report Groupings .....	250
Previewing a Report .....	251
Formatting .....	252
Report Filters.....	254
Clickthrough Reports.....	255
Saving a Report .....	256
Building a Matrix (Cross-tab) Report.....	257
Exercise 10.9 – Adding Fields to the Matrix Data Region .....	257
Creating Formulas.....	259
Creating a Chart Report.....	261
Parts of the Chart .....	262
Chart Types .....	263
Chart Options .....	264
<b>Appendix A: System Requirements .....</b>	<b>269</b>
SQL Server 2005 System Requirements .....	270
Internet Requirements .....	270
Hardware Requirements.....	270
Operating System Requirements .....	271
<b>Appendix B: Installation Instructions .....</b>	<b>273</b>
Setup Instructions .....	274
Configuring Reporting Services .....	278
<b>Appendix C: Expression Functions for Development Studio.....</b>	<b>281</b>
Operators .....	282
Common Functions.....	283
Globals Functions .....	289
<b>Appendix D: Formula Functions for Report Builder .....</b>	<b>291</b>
Functions.....	292



# Introduction

## ***Introduction Objectives***

This manual is written to give you a step-by-step guide for your classroom training and a handy reference for your daily work. In this Introduction, you will learn how to use this training guide effectively. This section covers the following topics:

- An introduction to the Reporting Services application
- Class objectives
- Help with using this training guide
- Information on how to start the program

## ***About SQL Server Reporting Services***

In today's information intensive environment, every business has a database of some sort. After all, business today is all about information and databases give you a handle on the massive amounts of information you must deal with. Therefore, your business has a database and from that database, you need reports. The problem is, most reporting capabilities bundled with database applications are limited or missing entirely. Many users need to report on data from multiple sources, which may include different database vendors such as Oracle, Microsoft SQL Server, DB2 or Sybase. Another difficulty is having the ability to distribute reports within or outside the organization in multiple formats.

SQL Server 2005 Reporting Services is one of the most powerful reporting programs available with the ability to pull data from all types of data sources. You can use Reporting Services to generate reports from any of the standard PC database programs, Access, Paradox, or FoxPro, as well as from a mainframe or server database. Reporting Services also has a powerful web-reporting server allowing you to distribute your reports over the web.

Another attractive feature is Reporting Services ability to integrate with any custom built applications created using Visual Studio, one of the most widely used development platforms. Reporting Services makes report generation easy without requiring you to be a programmer or a database expert by using the Report Builder application. For advanced developers, the SQL Server Business Intelligence Development Studio and Visual Studio offer a complete authoring system to build any kind of report required.

## ***Training Philosophy***

Studies show people retain 10% of information they see, 20% of information they hear, 50% of what they see and hear, and 80% of what they see, hear and do. In line with this, this class utilizes a hands-on method of training. You will see the effects of new procedures on the screen, hear the instructor explain how and why to use features, and perform the actions yourself as you learn.

In addition, this class focuses on your ability to perform tasks using the most productive techniques. The manual may contain several methods of accomplishing a certain task. However, class time does not allow for practice of all methods for each task. Your instructor will guide you in the most effective method of performing a task, but inform you of other methods that are available.

Questions are encouraged. While we give our best effort to explain new concepts in understandable terms, you may need to hear the concept again or have it explained more thoroughly. Please let the instructor know when you need more information!

## ***Class Objectives***



This class is a performance based instructional system. It is geared to provide you with the tools you need to build and distribute reports the quickest, most efficient way. After completing this course, you will be able to perform the following tasks:

- Understand the Reporting Services key features and architecture
- How to create a basic report
- Control data access
- Perform basic formatting techniques
- Apply various administration functions
- Create subscriptions
- Enforce security models and concept
- Create Report Models and deploy them
- Create reports with the Report Builder

## About This Manual

Each section of this manual contains objectives to provide you with the overall goals for the lesson. Lessons have descriptions of features and concepts followed by systematic directions for completing a specific task. Each section ends with a challenge exercise to help you practice the skills you learned in the lesson. Challenge exercises provide you with tasks to accomplish. Try to complete these exercises on your own (not available in version 1.0 of this manual)

As you work in this Training Guide, certain conventions are used to identify specific procedures. Use the following table as a guide:

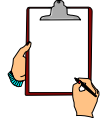
TRAINING GUIDE CONVENTIONS	
Item	Illustrated As
Menu Commands	Underlined letters for accessing menu commands are shown: Example: <u>F</u> ile/ <u>O</u> pen
Command Buttons	Command Buttons in dialog boxes are shown as buttons:  Example: 
Categories, Radio Buttons, Text Boxes, Check Boxes	All options within dialog boxes are listed in italicized text:  Example: the <i>Include Group Header</i> check box the <i>Hidden</i> radio button
Keystrokes	Keyboard keys are indicated by uppercase text:  Example: press ENTER  Keyboard combinations are shown in uppercase text with a plus sign (+) between the keys that need to be pressed simultaneously.  Example: press CTRL + S to save
Toolbar Buttons	Toolbar buttons are indicated by the button name and a graphic image of the button:  Example: click the Print  button
Typing or File Selections	Text to be typed or file names to be selected are printed in bold letters:  Example: type <b>Henry</b>
Exercises	Step-by-Step exercises in the text are indicated by bold text and the ❖ symbol.  For example: ❖ <b>Exercise - Format Objects</b>

## ***Tips, Notes, and Warnings***

Tips, notes and warnings display with the following icons. Text for these additional comments display in bold and italics as shown below:



*This icon indicates a tip or shortcut.*



*This icon points out a note of additional information.*



*This icon calls attention to a warning or very important note*

# NOTES

# Lesson 1

## The Big Picture

## ***Lesson Objectives***

After completing this lesson, you will be able to:

- Identify the Reporting Services key features
- Understand the Reporting Services key components
- Know the differences between the each report design option
- Work with data extensions
- Choose the proper deployment options
- Know the report rendering or formatting extension options

## **SQL Server 2005 Reporting Services Features**

Reporting Services is a comprehensive, server-based solution enabling the creation, management, and delivery of paper-oriented reports and interactive, web-based reports. Reporting Services combines the data management capabilities of SQL Server and Windows Server to deliver real-time information in support of daily operations and decision making. The major features are:

- **Report Authoring**  
Reports can be developed using Microsoft's feature-rich set of designer tools (covered later). Since Reporting Services uses Report Definition Language (RDL), an XML-based industry standard to define reports, any third-party development tool can be used
- **Report Management**  
Report definitions, folders, and resources are published and managed as a web service. Managed reports can be executed either on demand or on a specified schedule, and are cached for consistency and performance. New in Reporting Services, administrators can use the Management Studio to organize reports and data sources, schedule report execution and delivery, and track reporting history
- **Report Delivery**  
Reporting Services supports both on-demand (pull) and event-based (push) delivery of reports. Users can view reports in a web-based format or in e-mail
- **Report Security**  
Reporting Services implements a flexible, role-based security model to protect reports and reporting resources
- **Enterprise Reporting**  
Enterprises can use Reporting Services for their operational reporting or business intelligence applications. Using Reporting Services, corporate IT staff can design a variety of reports and deploy them to individuals throughout the enterprise
- **Ad-hoc Reporting**  
Reporting Services includes Report Builder, a new ad-hoc reporting tool that enables business users to create their own reports and explore corporate data. Report Builder incorporates a user-friendly business query model that enables users to build reports without deep technical understanding of the underlying data sources

---

**Lesson 1: The Big Picture**

- **Embedded Reporting**  
Organizations can access pre-defined or ad-hoc reports from third-party applications that use Reporting Services and use these reports as-is, customize them, or create new ones for specific business needs
- **Web-based Reporting**  
Organizations can deploy interactive web-based reports to deliver information to customers or partners over extranets of the Internet. Reporting Services isolates report consumers from the complexity of the underlying data sources, while providing personalization and interactivity

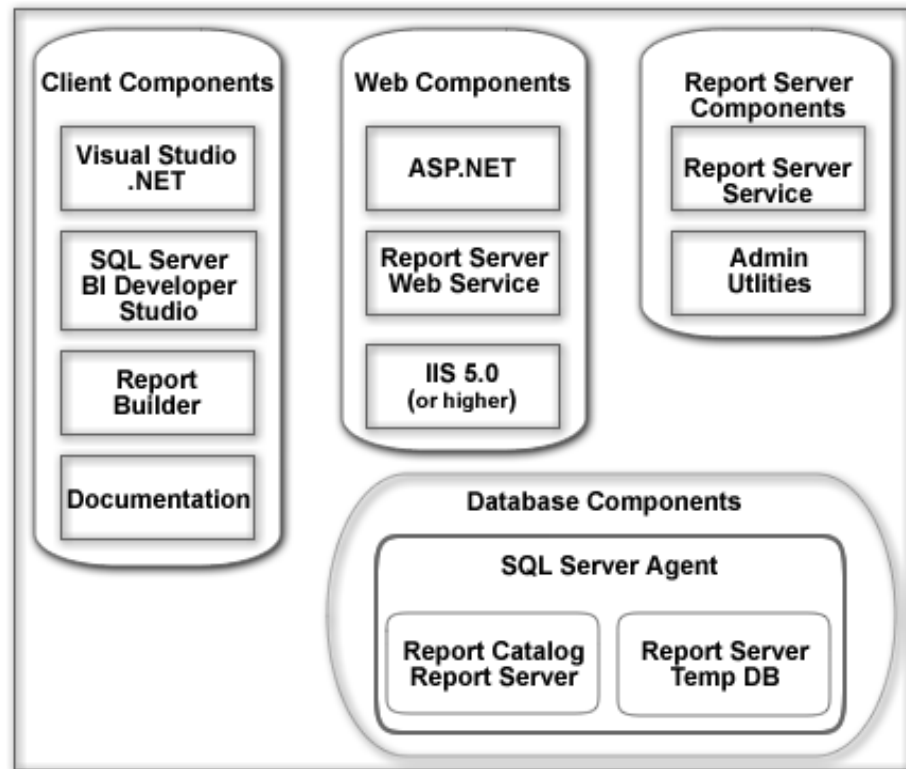
## ***The Reporting Services Key Components***

Reporting Services is a complex set of services, web applications and databases which work together to create a complete end-to-end reporting solution. The best approach to understanding what programs are required and how they work together is to separate the application into various categories. This section of the lesson will provide a high level view of these categories and provide a general understanding of the underlying program within each category.

For the purposes of this class, we will limit our categories to:

- Client Components
- Web Components
- Report Server Components
- Database Components

## Lesson 1: The Big Picture



It is important to understand not all components are used in any given situation. Much depends on the situation, such as whether a developer is designing a report or an end user is simply viewing a report. However, in a true enterprise reporting environment all components will be used at one time or another. We will breakdown each component with a brief overview.

## Client Components

The client components are tools which can be installed on a client desktop and used to develop report, report model, analysis and integration project types. Each of these project types will either be discussed in this manual or more thoroughly in the Developer 2 workshop. The developer tools of the client components are:

- **Visual Studio 2005**  
This latest release of Visual Studio includes the Report Designer module, which provides all the functionality for building any kind of Reporting Services project type. The advantage of using Visual Studio over other client components is it is a complete developer tool for building custom applications in several programming languages, such as Visual Basic, Visual C#, Visual J# and Visual C++. Visual Studio also provides an excellent Integrated Development Environment (IDE) for creating ASP.NET applications integrating Reporting Services for deploying projects to the web

---

Lesson 1: The Big Picture

- **SQL Server Business Intelligence Development Studio (Development Studio)**

The Development Studio offers the same style and functionality of Visual Studio, but is limited to only creating Reporting Services project types. It is full featured and best of all free with SQL Server 2005. Unless you have a very good reason to purchase Visual Studio 2005, this IDE will meet all of your reporting development needs. Best of all it can be deployed to anyone in the organization without additional software costs
- **Report Builder**

The Report Builder is a very good solution to a long running problem. It provides a very basic report designer to be used by non-technical report developers. As with most commercial report developer tools such as Visual Studio or SQL Server Business Intelligence Development Studio, the learning curve can be steep and usually only mastered by technically proficient developers. The Report Builder offers a very simple interface with a drag and drop approach easy to master. It does limit reports to work only with published Report Models (covered later). Additionally, even though Report Builder loads on the client as a classic Windows interface, it is deployed from a web interface and is not installed on the client desktop. This approach simplifies deployment and future upgrades
- **Documentation**

Reporting Services is an extensive feature-rich reporting solution able to compete directly with any enterprise reporting solution. This does mean the application is difficult to understand and master quickly. Fortunately, included with the installation disk are exhaustive volumes of developer information, examples and tutorials. The documentation also is dynamically connected to the web for up-to-date information and resources

## Web Components

Reporting Services is dependent upon a web-architecture cannot install or be configured without Internet Information Services (IIS) and ASP.NET 2.0 preinstalled. The following is a breakdown of each Web Component:

- **ASP.NET**

The Active Server Pages (ASP) set of server controls provide an HTML-style of declarative programming and enable you to build complex web pages with a minimal amount of programming. It competes with other HTML-style scripting languages such as Java Server Pages (JSP) and Hypertext Preprocessor (PHP). Reporting Services is dependant on the server-side controls in ASP.NET. It is very important ASP.NET is installed, configured and working properly before installing Reporting Services

---

Lesson 1: The Big Picture

- **Internet Information Services (IIS)**  
IIS is the web server used to run the web interface between a browser and Reporting Services. It essentially serves requests and responses between the two sides. IIS 5.0 or higher is required to be installed, configured and working properly before installing Reporting Services
- **Report Server Web Services**  
This is, as the name implies, a web service. A web service is any software using Extensible Markup Language (XML) to send a request via the internet to another application and then receive a response from that application. In this case the web service allows an application or web page to communicate directly with the Report Server to do many types of operations, such as publish reports, view reports and even administer the Report Server. This component is installed automatically when Reporting Services is installed. It makes sense then that ASP.NET and IIS is working properly before the web service can be installed properly

## Report Server Components

These components are at the core of Reporting Services. If there is failure in the system it is best to start looking here for fixes. Fortunately, there are only two items to understand:

- **Report Server Service**  
This service is installed during the installation of Reporting Services. Like any normal Windows service, it only works if it is running. The Report Server Service launches Reporting Services and is the most likely point of failure in the system. Like most services, it requires a valid user login with enough authorizations to access other services, programs and directories
- **Administration Utilities**  
Most of the administration utilities are command-line driven. Performing such tasks as stopping and starting the Report Server Service and changing its login credentials can be performed through the command-line. The Report Server Configuration Manager is the only GUI interface available for performing some administration operations outside of the command-line

## Database Components

Reporting Services uses the excellent data processing capabilities found in SQL Server 2005 to process reports. For SQL Server to do this properly, communications and databases need to be correctly configured and working. The three major layers to the database components are:

- **SQL Server Agent**  
The SQL Server Agent is installed as part of the SQL Server installation and is used to run scheduled jobs. Its relationship with Reporting Services is used the same way. Reporting Services uses the agent to run scheduled jobs. A scheduled job can be on demand (now) or recurring at certain times. Either way, SQL Server Agent is used to submit the request
- **Report Server Catalog Database**  
This database is created as part of the Reporting Services installation and is used to hold information about Reporting Services, such as deployed project types, website, directory folder structure and scheduled jobs
- **Report Server Temp Database**  
Another database created during the installation. It is used to hold information briefly, such as current users logged in and recently run reports. It does not hold onto report information for long and most report data is stored in a separate file for the long-term

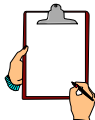
## Report Design Options

Reporting Services offers three design options for developing reports or project types. The choices are Visual Studio 2005, SQL Server Business Intelligence Development Studio (Development Studio) and Report Builder. The first two options offer the exact same capabilities.

### Visual Studio 2005 and Development Studio

If you are a Visual Studio developer building custom applications, then use Visual Studio. If you are not a VS developer, but want to have access to the most comprehensive designer, then use Development Studio. Development Studio is available as part of Reporting Services with no additional costs.

The design view for either Visual Studio or Development Studio looks identical and both products will usually display the words “Visual Studio” in the title bar:

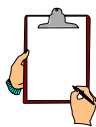
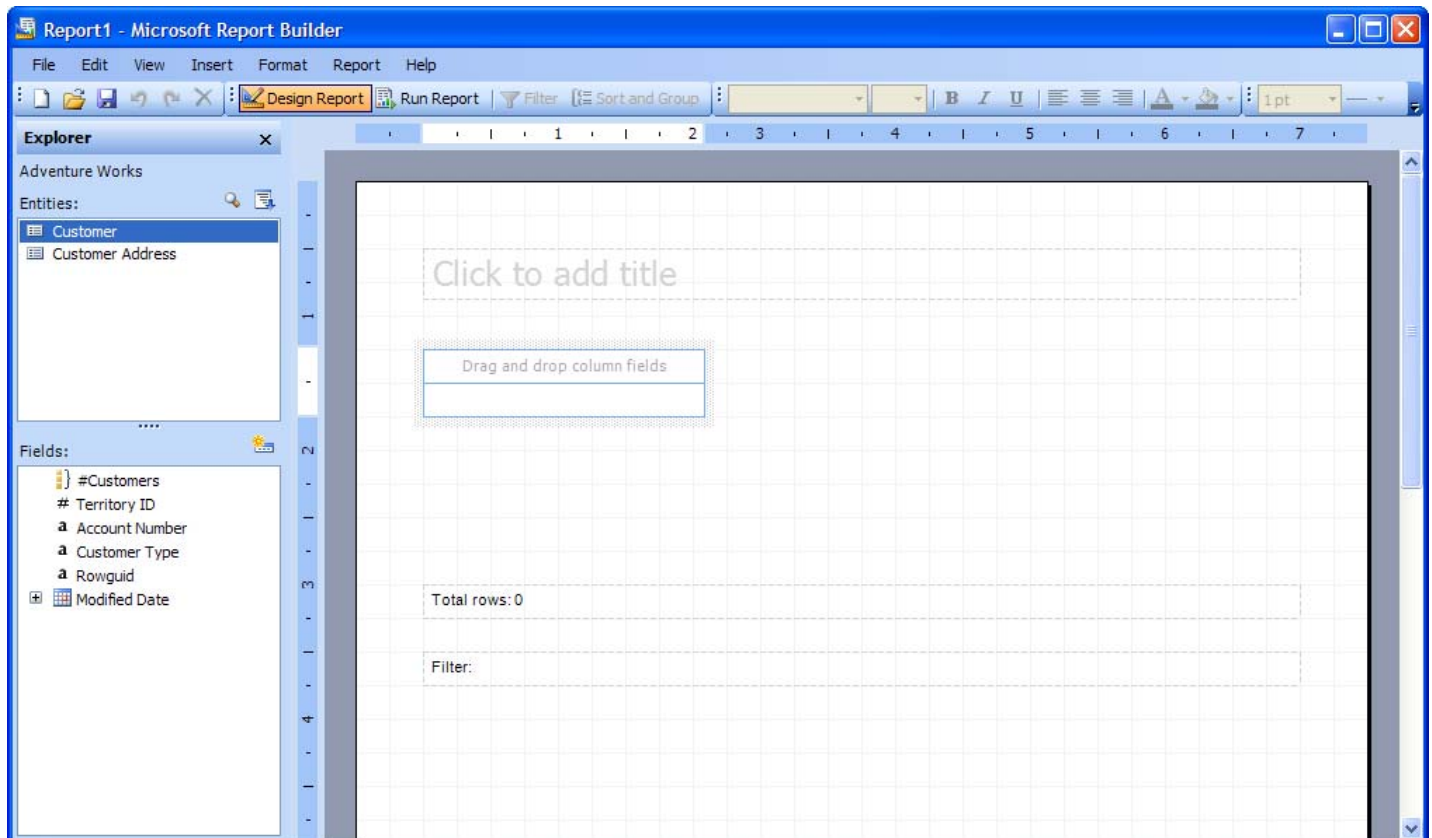


**NOTE:** The remainder of screen shots and references will be done using Visual Studio 2005 for this courseware. Please bear in mind all of the examples and screen views will be the same if you are using Development Studio.

## Lesson 1: The Big Picture

## Report Builder

Report Builder is a unique designer due to its purposeful simplicity. It is a tool for general end users requiring an ad-hoc report designer, but do not wish to configure underlying data sources. The user interface is modeled on familiar Microsoft Office paradigms such as Excel and PowerPoint. Report Builder is launched from a browser and will load automatically as a client application on the user's desktop.



**NOTE:** Before being able to use Report Builder, Report Models need to be created and published using the Model Designer in Visual Studio 2005 or Development Studio. This topic is cover later in the course.

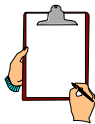
## Data Processing Extensions

Data processing extensions in Reporting Services enable you to connect to a data source and retrieve data. They also serve as a bridge between a data source and a dataset. Reporting Services data processing extensions are modeled after a subset of the Microsoft .NET Framework data provider interfaces.

### Included Extensions

The following lists the data processing extensions included with Reporting Services:

- **SQL Server 2005**  
A data provider for SQL Server to connect to and retrieve data from the SQL Server Database Engine
- **OLE DB**  
A data provider for OLE DB. With this extension, the report server can query any data source that has an OLE DB provider
- **Oracle**  
A data provider for Oracle. With this extension, the report server can access Oracle data sources through Oracle client connectivity software
- **ODBC**  
A data provider for ODBC. With this extension, the report server can access data in any database for which there is an ODBC driver



**NOTE: Each of the above data processing extensions use the .NET Framework Data Provider.**

### Custom Data Processing Extensions

You can develop data processing extensions for your own data sources. This means you can deploy your own custom data processing extensions into the existing report servers of your organization. You can also use them as part of a custom reporting suite that you provide to your report designers.

Two major advantages to consider for using custom extensions are:

- Simplified data access architecture for easier manageability and improved performance
- Exposing extension-specific functionality to end users

## ***Deployment Options***

Reporting Services offers several options for deploying reports. Depending on the option chosen and the data source being reported from, some consideration of setup is required. Database login credentials and possible database client software being installed on the Reporting Services server are the most common issues to resolve. The following ways to deploy reports are:

- **Report Manager**  
Use the default web application installed with Reporting Services to deploy your reports. Users can access the site to run and subscribe reports
- **Embedded URLs**  
Determine the URL location of a published report and run the report completely within your existing portal
- **SharePoint Web Parts**  
Use your existing SharePoint Portal to view Reporting Services folder hierarchy and even run reports

### **Key Considerations**

- **Oracle Reports**  
To deploy a report using Oracle as a data source, the Oracle client tools must be installed and configured correctly on the report server. Without doing so, it is likely the reports will work on the developers machine, but fail upon deployment
- **Staging Development vs. Production Reports**  
Create a separate folder on the report server for testing purposes and then upon success, move the reports to a production folder. Through assignment of user rights you can keep reports in development out of view of general users. Using the same report server for testing and production also minimizes troubleshooting
- **Configure Data Sources**  
If a report needs to support user subscriptions, scheduled report history or it runs as a report execution snapshot, you must supply it with stored credentials after it has been published

## Distribution Options

So far we have mentioned ways to run reports directly in real-time views through the use of Report Manager, embedded URLs and SharePoint. It is possible to distribute the reports in a scheduled delivery process called subscriptions. Subscriptions are covered in complete detail in a following lesson, but as part of the big picture it is important to understand subscriptions offers three delivery options:

- **E-mail**

This delivery option can be performed in several ways

- A generated email with a hyperlink to the generated report
- Include the notification in the subject line using variables such as @ReportName and @ExecutionTime to properly identify the subscription
- Embed the report as an inline or attached report. The rendered report's format and email client will determine if the report is embedded or not

- **File Share**

To send a report to a file share, the location must already exist because Reporting Services will not create a directory. You also must use Uniform Naming Convention (UNC) to specify the location. For example:

```
\\<servername>\c$\reportarchive\operations\2003
```

Lastly, it is important to ensure the report server has access to the file share location

- **Custom Delivery Extensions**

Reporting Services offers a delivery extension API for many types of operations such as receiving notifications and status.

## Rendering Extensions

A rendering extension is a module of the report server able to export report data and layout information into various formats such as HTML, Excel, CSV, XML, Image, and PDF. You can create additional rendering extensions to generate reports in other formats.



**To determine which rendering extensions are available, you can view the list of installed extensions in the *RSReportServer.config* file.**

## Lesson 1: The Big Picture

The following describes the report export options available in reporting Services:

<b>Extension Name</b>	<b>Description</b>
<b>XML</b>	Renders a report in XML. This extension can be modified to create a simple solution for custom rendering extensions
<b>CSV</b>	Renders a report in comma-delimited format
<b>Image</b>	Renders a report in a page-oriented TIFF format
<b>PDF</b>	Renders a report in the Adobe Acrobat Reader
<b>Excel</b>	Renders a report in Microsoft Excel
<b>HTML 4.0</b>	HTML is the format used to initially render the report. Only for browsers supporting this HTML 4.0 standard
<b>HTML 3.2</b>	HTML is the format used to initially render the report. Only for browsers not supporting this HTML 3.2 standard
<b>MHTML</b>	Renders a report in MHTML. The report opens in Internet Explorer. The format is shown as Web Archive in the Export drop-down of the report toolbar
<b>HTMLLOWC</b>	Renders a report in an Office Web Component that loads within the browser window. The format is shown as HTML with Office Web Components in the Export drop-down of the report toolbar
<b>Null</b>	Does not render a report to a specific format. This rendering extension is useful for placing reports in cache. Null rendering should be used in conjunction with a scheduled execution or delivery.

# NOTES

# NOTES