



Reporting Services

White Paper

Published: August 2007

Updated: July 2008

Summary: Microsoft SQL Server 2008 Reporting Services provides a complete server-based platform that is designed to support a wide variety of reporting needs including managed enterprise reporting, ad-hoc reporting, embedded reporting, and web based reporting to enable organizations to deliver relevant information where needed across the entire enterprise. Reporting Services 2008 provides the tools and features necessary to author a variety of richly formatted reports from a wide range of data sources and provides a comprehensive set of familiar tools used to manage and secure an enterprise reporting solution. Reports are processed and delivered quickly and effectively, enabling users to either receive reports automatically through subscriptions, to access reports from a central report repository on an ad-hoc basis, or to consume reports in context of their business processes through reports directly embedded into their business or web applications.

Contents

Introduction	1
Authoring Reports	1
Using Report Development Tools	2
Accessing Data Sources for Report Creation	4
Creating Compelling Reports.....	5
Managing Reporting Services	7
Extending Management Capabilities	7
Configuring a Reporting Services Instance.....	8
Microsoft Office SharePoint Services Integration.....	9
Securing Reporting Services	9
Delivering Reports.....	10
High Performance Report Processing	10
Caching.....	10
Snapshots	10
Multiple File Formats	11
Delivering Reports through Subscriptions.....	11
Embedding Reports into Business Applications.....	11
Conclusion	12

Introduction

Although companies often collect and store large amounts of data, it is sometimes difficult to present that data in a meaningful way and provide insight into what is going on in the business to enable business decision makers to make relevant and timely decisions for the company.

To be able to make effective business decisions, all types of users across the company from business managers to information workers need to have easy access to informative and intuitive reports that combine data from the many data sources throughout the enterprise. In most organizations, these reports need to combine detailed numeric and text-based reports that provide a comprehensive account of business activity, with graphic visualization that help provide a better view of trends and comparisons.

Reporting Services provides companies with the ability to fill a variety of reporting scenarios.

- **Managed Reporting.** Also often referred to as enterprise reporting - supports the creation of reports that span all aspects of the business and delivers them across the enterprise to provide every employee real time access to information relevant for their business area and enable better decision making
- **Ad-Hoc Reporting.** Enables users to create their own reports on an ad-hoc basis and provides them with the flexibility to quickly get the information that they need, in the format that they need it without submitting a request and waiting for a report developer to create the report for them.
- **Embedded Reporting.** Enables organizations to embed reports directly into business applications and web portals, enabling users to consume reports within the context of their business process. Deep integration with Microsoft Office SharePoint Server 2007 also enables organizations to deliver reports through a central report library or to use new web parts for thin rendering of reports directly within SharePoint enabling easy creation of dashboards. In this way organizations are able to bring all business critical data, structured as well as unstructured, from across the company together in one central location providing one common experience for information access so that users can see key business performance information at a glance.

This whitepaper describes how SQL Server 2008 Reporting Services provides a complete, server-based platform to easily support this wide variety of reporting needs.

Authoring Reports

Report authoring is a major activity in many organizations. Executives, business analysts, managers, and increasingly information workers throughout the enterprise rely on timely and accurate information from easy to understand reports to perform their job effectively. SQL Server 2008 Reporting Services includes comprehensive report authoring tools, and a range of report format innovations that make it easy to create reports that bring data to life and

provide the information that employees need in whatever format is most effective for your organization.

Using Report Development Tools

In most organizations, there are two distinct groups of people who create reports; experienced business intelligence solution developers who are used to working in a comprehensive development environment, and business users who are unfamiliar with database schema designs and need an intuitive report design environment that abstracts the underlying technical complexities.

SQL Server 2008 meets both of these needs by providing distinct report development tools specifically designed to meet the needs for these two audiences. This enables developers to create sophisticated reporting solutions for the entire enterprise, while making it easy for business users to focus on the specific data relevant for their business area.

Report Designer

Report Designer provides a highly flexible and productive report authoring environment for developers and advanced report authors as shown in Figure 1. Report Designer is a component of SQL Server Business Intelligence Development Studio, a Business Intelligence development environment based on the common development interface of Microsoft Visual Studio. The design interface enables developers to easily define datasets for a report from various data sources, design the layout of the report, and then preview the report directly in the design environment before deploying it to the Report Server. Developers can create reports quickly and easily by using the Report Wizard or they can build more extensive reports by using the visual design environment in Report Designer. The built-in query designer simplifies the task of retrieving the datasets used by reports, and the intuitive drag and drop design interface makes it easy to lay out report elements for display. When a developer has completed a report, they can preview it and deploy it to the report server directly from Report Designer.

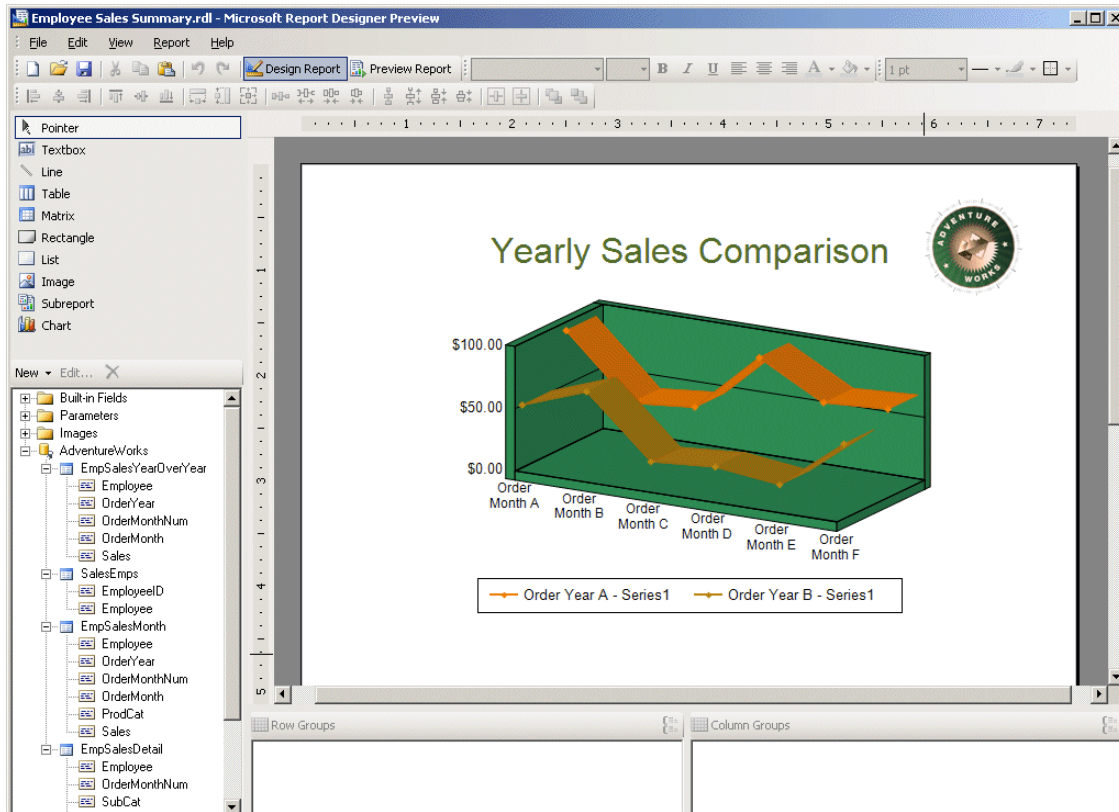


Figure 1: Report Designer

Report Builder

Having developers develop powerful reports that can be deployed to thousands of users is cost efficient and brings the benefits of consistent, professionally designed reports to the whole enterprise. However, sometimes individual business users need to either modify reports or even create their own reports as specific requirements occur. Report Builder is an easy to use, business-focused report query tool that abstracts underlying business data through report models that reflect business entities such as customers, products, sales, and so on. Report Builder is shown in Figure 2. By creating a model that reflects this business-oriented view of the data, Report Builder enables business users to create robust reports without having extensive knowledge of underlying the database structure or the query syntaxes.

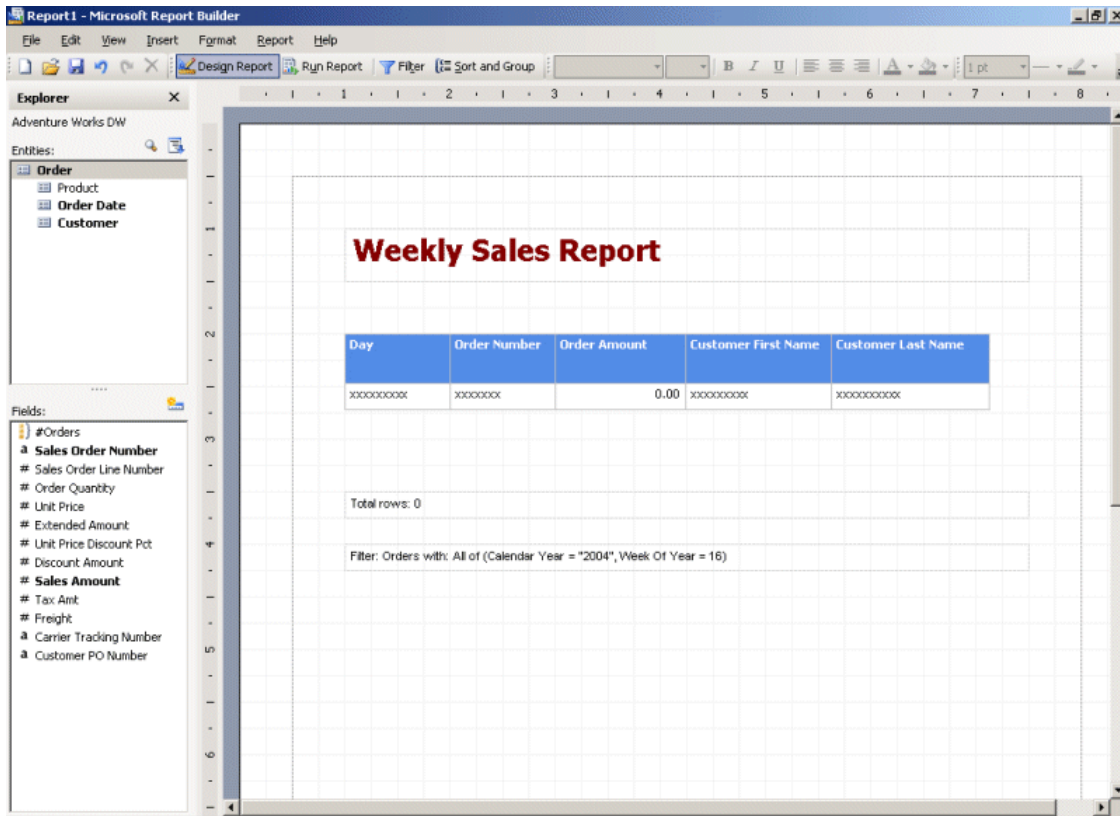


Figure 2: Report Builder

Accessing Data Sources for Report Creation

Most companies have data spread across a variety of applications and databases. To get a full understanding of business performance, users need a reporting solution that can pull data from multiple disparate data sources across the organization. For example, sales executives within the company might need a report that incorporates financial data from the accounting department's Oracle database with customer data from a Microsoft SQL Server-based CRM application. SQL Server 2008 Reporting Services supports access a variety of data sources for report creation, including SQL Server relational databases, SQL Server Analysis Services, Oracle, DB2, and Hyperion Essbase; and enables the report author to combine data from multiple sources in a single report. Additionally, with SQL Server 2008 Reporting Services report authors can generate report models for use with Report Builder from SQL Server relational databases, SQL Server Analysis Services cubes, and Oracle databases.

Creating Compelling Reports

Business users need the flexibility to view and present different types of data in a variety of formats. Reporting Services fills this need by providing a strong set of compelling report authoring features including multiple display and parameter options, interactivity, rich text formatting, and a broad variety of charting capabilities.

Users have different requirements and expectations when viewing and interacting with reports, and a report format that works well for one kind of data might not be appropriate for other kinds of data. SQL Server Reporting Services supports a number of display options, enabling you to create reports that display data in the most suitable layout for the users who will consume them.

SQL Server Reporting Services uses data regions to define the layout for the data in reports. You can use the following data regions to display data:

- **List.** A freeform layout for data.
- **Chart.** A graphical format that displays data in the form of a chart.
- **Table.** A tabular format in which data is displayed in rows and columns. You can create a hierarchy of rows to reflect groupings in your data and display group totals.
- **Matrix.** A format that supports row and column groups, and which can display aggregated summary data in the cells where row groups and column groups intersect one another, similarly to a pivot table or crosstab.
- **Tablix.** A hybrid between Table and Matrix, introduced with the release of SQL Server 2008 Reporting Services.

Tablix

As mentioned above, SQL Server 2008 introduces a new data region type called *Tablix*. As the name suggests Tablix provides a combination of the best features of table and matrix data regions. Tablix enables you to create reports that use a flexible layout with multiple row and column groups, making it easy to build highly versatile report layouts that were difficult or impossible to implement using any report authoring tools available in the market today. With Tablix, you can build reports that include:

- **Fixed and dynamic columns and rows.** In a Table, a report can have dynamic rows in which rows repeat and multiple nesting of groups is allowed, but columns are always fixed. In a Matrix, a report can have dynamic rows as well as columns, but no fixed rows and columns can be included. With Tablix these restrictions do not apply and reports can include both dynamic and fixed rows and well as dynamic and fixed columns.
- **Arbitrary nesting on each axis.** When using a Table in previous version of Reporting Services, arbitrary nesting was available on rows only. With the introduction of Tablix arbitrary nesting is available on both rows and columns.

- **Optional omission of row or column headers.** Because Tablix offers everything that is available with both Table and Matrix, a report author can omit the row and column headers. This technique was previously only possible with a Matrix, but with Tablix it can also be applied to Tables.
- **Multiple parallel row or column members at each level.** The ability to apply multiple parallel rows and column members within the same report is a new unique design feature that can significantly change the way information is represented in reports. For example, when designing a sales report the report author can choose to have one dynamic hierarchy for a column showing *Year*, followed by another dynamic column showing *Product* as illustrated below. Note that the two hierarchies are displayed next to each other independently, and yet still share the same set of row values.

Regional Sales Report		Year		Product	
		2005	2006	Table	Chair
Washington	Seattle	50	60	20	30
	Spokane	30	40	10	20
Oregon	Portland	40	50	10	10
	Eugene	20	30	25	5

Charts

Large amounts of complex data in text format can be difficult for readers to make sense of. With increased investments in new visualization capabilities in Reporting Services SQL Server 2008 offers an enhanced charting engine, including a large variety of enriched charts that enable decision makers to view data in a friendly format that immediately enables them to see the big picture. Reporting Services provides users with a large variety of chart types including traditional bar, area, column, doughnut, line, and pie charts. Reporting Services 2008 also includes support for per-series plot type charts with multiple chart areas and legends, range charts, polar, radar, pyramid and funnel charts. Chart enhancements such as annotations and support for large data volumes provide greater flexibility to your reporting environment.

Interactive Features

SQL Server 2008 Reporting Services offers a number of report design features that enable developers to create highly dynamic reports that provide users with the ability to interact with the data and explore it in greater depth. Interactive elements that can be added to reports include:

- **Parameters.** Adding parameters to reports enables users to provide specific values when the report is processed in order to filter the information represented in the report. For example, a user could provide a *Region* parameter value for a sales report to view only sales in the specified region. Parameters help report authors reach a wider audience with a single report, and empower users with a more flexible and interactive experience.

-
- **Interactive sorting.** Applying sort capabilities to a report enables users to sort the data by any of the columns the report contains in ascending or descending order. For example, a user viewing a report containing a list of sales might sort the data first by date, and then by sales amount.
 - **Actions.** Developers can add links to a report that enable users to perform actions, such as jumping to another area of the report, opening an external Web page, or opening another, related report in order to perform drill-through analysis. For example, a sales report might list the order number, customer name, and sales amount for orders placed in the current month. The report developer could add a drill-through action to the report so that by clicking the order number for a given order, users can open a report showing the details of that order.
 - **Document maps.** Users can use a document map to navigate a report easily. The document map is displayed as a side pane in the report, and users can use the hierarchy of links it contains to move around the various sections of the report.
 - **Sub-reports.** Developers can embed one report inside another as a sub-report; a powerful feature of Reporting Services that enables developers to display information from one report as a sub-region directly within another report. For example, a developer might create a report that shows each sales region in the organization with a sub-report showing the sales performance for each region.

Managing Reporting Services

Administrators can manage a SQL Server 2008 Reporting Services solution in one of two modes; Native mode and SharePoint Integrated Mode. When running in Native Mode, administrators can use a comprehensive set of management and configuration tools that provide a uniform experience across all SQL Server 2008 technologies. Database Administrators can take advantage of the SQL Server Management Studio to integrate management of a Reporting Services instance with SQL Server database engine management, and can delegate Reporting Services service management to system administrators by creating a custom Microsoft Management Console with the SQL Server Configuration Manager MMC snap-in.

When running SharePoint Integrated Mode administrators can leverage the integrated management capabilities of SharePoint and benefit from a single integrated management experience across the two server components directly within SharePoint.

Extending Management Capabilities

Report Manager, shown in Figure 3, is a Web based tool that can be used in Native Mode to access and manage Reporting Services over the Web. With Report Manager, administrators can perform a wide variety of tasks including, viewing and managing reports and subscriptions, configuring report processing options, and configuring security from any computer with a browser.

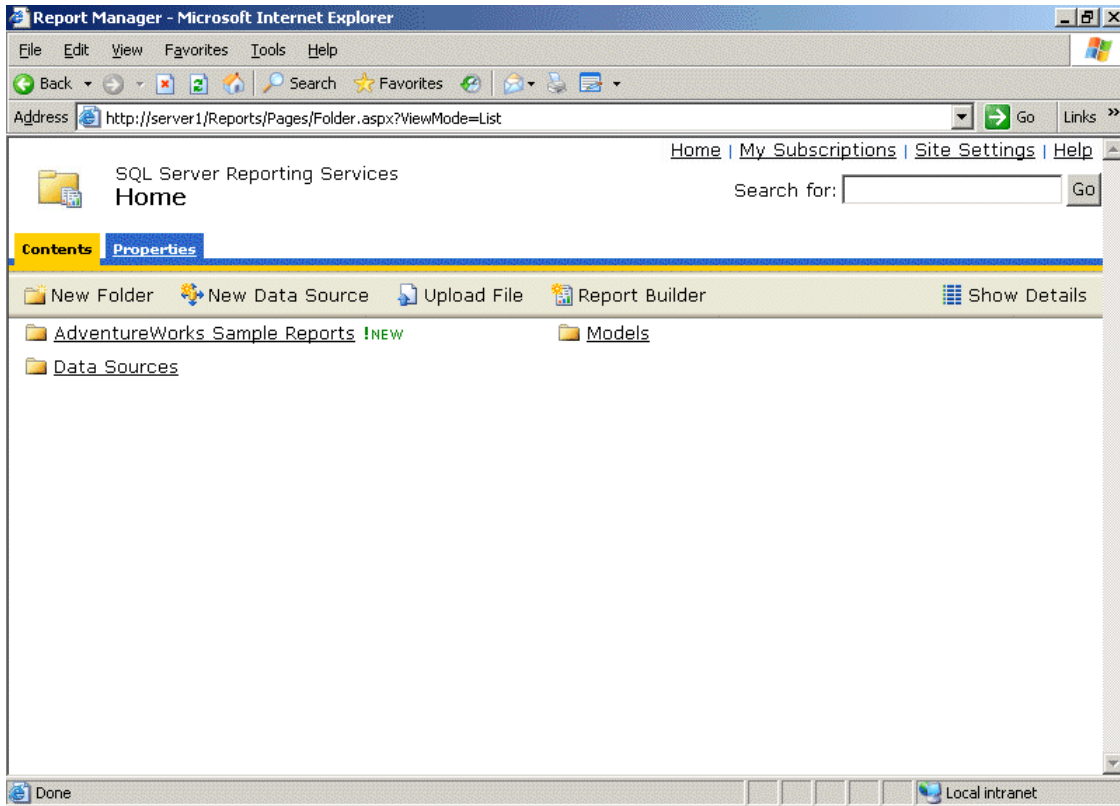


Figure 3: Report Manager

Configuring a Reporting Services Instance

Choosing the default configuration when installing Reporting Services by using the SQL Server setup program, automates the entire configuration process for Reporting Services. However, many organizations often add and change Reporting Services configuration options after the initial installation, for example to deploy a Reporting Services solution as a scalable Web farm. Reporting Services Configuration Manager, shown in Figure 4, makes it easy for administrators to perform each of the tasks required to configure a Reporting Services deployment, which results in reduced downtime and faster delivery of reporting solutions.

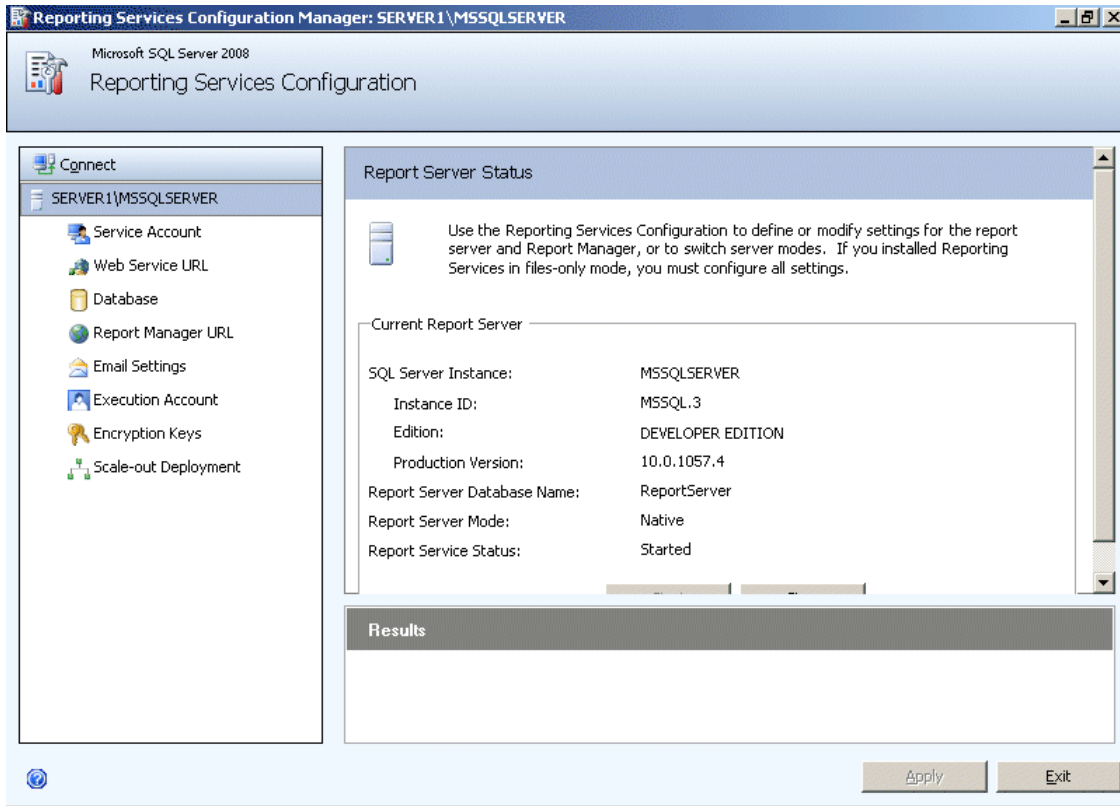


Figure 4: Reporting Services Configuration Manager

Microsoft Office SharePoint Services Integration

SharePoint Integrated mode provides access to report server content and operations through a SharePoint site. SharePoint integration provides shared storage, shared security, and same site access for all business documents including reports, report models, and shared data sources. Document collaboration and management features, such as checking reports in and out for modification, alerts to be notified of changes to reports, and the workflow capabilities of SharePoint can all add to greater accountability and flexibility in your reporting environment. Furthermore reports can be embedded directly into SharePoint pages for a seamless, contextualized reporting experience for users.

Securing Reporting Services

Security is always one of the top concerns of data administrators. Reporting Services provides flexible role based security grant administrative permissions and to secure access to shared data sources, reports, folders, and resources. Administrators can use both built-in as well as customized role definitions to fine tune the security of their reporting environment. A user can belong to different types of roles for different items. For example, a user can be a member of the Content Manager role for a report managed by their department

and also be a member of the Browser role for a report developed for another department within the company.

Administrators can further enhance security administration through SharePoint integration. When Reporting Services is running in SharePoint Integrated Mode, SharePoint permissions can be used to manage reports and folders within the SharePoint library.

Delivering Reports

Organizations are realizing the benefit of making business intelligence available to a much broader audience with the company; enabling more users to make decisions and perform tasks based on information that traditionally has only been available to a small group of business analysts. One of the challenges that exist today is delivering this data to the thousands of business users that can benefit from access to this data. SQL Server 2008 Reporting Services has the capability to effectively deliver richly formatted reports to the users who need them.

High Performance Report Processing

In many reporting solutions, very large reports can consume the reporting server's available memory causing other smaller reports to fail. SQL Server 2008 Reporting Services includes a scalable report engine that limits memory usage and conflicts with non-memory bound reports. Reporting performance is also optimized through on-demand processing and instance-based rendering.

Caching

Reporting Services further enhances performance by enabling the caching of reports. By caching reports that are frequently accessed using the same parameter values, organizations can reduce the processing overhead required to render reports. After the report has been processed the cached copy is subsequently available to other users who access the same report without requiring any further processing. With this approach, if multiple users open the report, only the first request results in report processing. The report is subsequently cached, and the remaining users view the cached report.

Snapshots

Reporting Services supports the creation of snapshot reports, which are rendered at a scheduled interval and made available for users to view. Snapshots are similar to cached reports; the main difference is that Snapshots are often created on a regular schedule. Like cached reports, Snapshots can improve reporting performance by periodically generating reports that require complex or time-consuming queries, and enabling users to view the pre-generated snapshot instead of processing the report on-demand. Snapshot

reports are also a useful way to maintain historical reports, because each instance of the report reflects the data as it was when the snapshot was generated.

Multiple File Formats

Users need to be able to access and share reports in the format with which they are most familiar. Reporting Services 2008 supports rendering to most common file formats including HTML, PDF, CSV, XML, and Image (TIFF) and also offers complete new rendering in Microsoft Office Word as well as improved rendering in Microsoft Office Excel. Reports in Microsoft Office formats are fully editable, enabling users to create custom documents in based on these reports.

Delivering Reports through Subscriptions

Reporting Services supports standard as well as data-driven subscriptions, enabling the automatic delivery of reports to the right users by pushing them directly to each user via email or by sending them to a file share from where users can pull the reports. Users can use standard subscriptions to have reports delivered to them based on a custom schedule and explicitly specified report parameter values. Centrally managed data-driven subscriptions, which retrieve subscription properties at run-time from an external data source, provide a way to deliver the same report to multiple users, each with their own individual report parameter and rendering format requirements.

By default, reports can be sent to subscribers via email or file shares. Developers can enhance the environment by adding additional delivery extensions that route reports to other locations.

Embedding Reports into Business Applications

Reports can be embedded into business applications by using the ReportViewer control; a Reporting Services control provided in Visual Studio. This control enables you to display a report in a .NET application, such as an ASP.NET Web page or a Windows desktop application.

The ReportViewer control supports *local* mode, in which reports are processed in the client application, and *remote* mode, in which the reports are run on a remote Reporting Services report server. A small company can start with the more limited capabilities of a local mode implementation and upgrade to a full Reporting Services integrated implementation as their usage and reporting needs grow.

Many organizations use SharePoint solutions to boost cooperative work between employees, partners and even customers. Configuring Reporting Services to run in SharePoint Integrated Mode enables user to publish or upload reports, report models, resources, and shared data source files to a SharePoint library. This enables you to embed reporting into your SharePoint-based Intranet solution by creating report dashboards that display reports from Reporting Services.

Conclusion

SQL Server 2008 Reporting Services provides a complete enterprise reporting solution that can support thousands of users by providing them with the reports that they need when and where they need them. These reports can bring together data from the many different data sources that exist within an organization, and deliver them in rich rendering formats to users across the organization and beyond.

For more information:

<http://www.microsoft.com/sql/technologies/reporting/default.mspx>

Please give us your feedback:

Did this paper help you? Tell us on a scale of 1 (poor) to 5 (excellent), how would you rate this paper and why have you given it this rating? For example:

- Are you giving it a high rating because it has good examples, excellent screenshots, clear writing, or another reason?
- Are you giving it a low rating because it has poor examples, fuzzy screenshots, unclear writing?

This feedback will help us improve the quality of white papers we release.

[Send feedback.](#)

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2008 Microsoft Corporation. All rights reserved.

Microsoft, PowerShell, SharePoint, SQL Server, Visual Basic, Visual C#, Visual Studio, Windows, Windows Server, and the Server Identity Logo are trademarks of the Microsoft group of companies.

All other trademarks are property of their respective owners.