TRAVERSE®

Reporting Services

Training Manual

ETMRS11

Document Number 10203.RSTRN

July 2011, Release 11

This document has been prepared to conform to the current release version of TRAVERSE Accounting Business Software for Windows. Because of our extensive development efforts and our desire to further improve and enhance the product, inconsistencies may exist between the software and the documentation in some instances. Call your customer support representative if you encounter an inconsistency.

CONTENTS

CONTENTS

1

Reporting Services

OVERVIEW

Microsoft SQL Server Reporting Services is a server-based solution for building enterprise reports that draw content from a variety of relational and multidimensional data sources, publishing reports that can be viewed in various formats, and centrally managing security and subscriptions. The reports that you create can be viewed over a Web-based connection or as part of a Microsoft Windows application or SharePoint portal.

Reporting Services includes graphical tools and wizards for creating and publishing reports and report models; report server management tools for administering Reporting Services; and application programming interfaces (APIs) for programming against and extending the Reporting Services object model.

Reporting Services includes the following core components:

- A complete set of tools that you can use to create, manage, and view reports.
- Report Server component that hosts and processes reports in a variety of formats. Output formats include HTML, PDF, TIFF, Excel, CSV, and more.
- An API (*application programming interface*) that allows developers to integrate or extend data and report processing in custom applications, or create custom tools to build and manage reports.

The reports that you build can be based on relational or multidimensional data from SQL Server, Analysis Services, Oracle, or any Microsoft .NET data provider such as ODBC or OLE DB. You can create tabular, matrix, and free-form reports. You can also create ad hoc reports that use predefined models and data sources.

Visually and functionally, the reports that you build in Reporting Services surpass traditional reporting by including interactive and Web-based features. Some examples of these features include drill-down reports that enable navigation through layers of data, parameterized reports that support content filtering at run time, free-form reports that support content in vertical, nested, and side-by-side layouts, links to Web-based content or resources, and secure, centralized access to reports over remote or local Web connections.

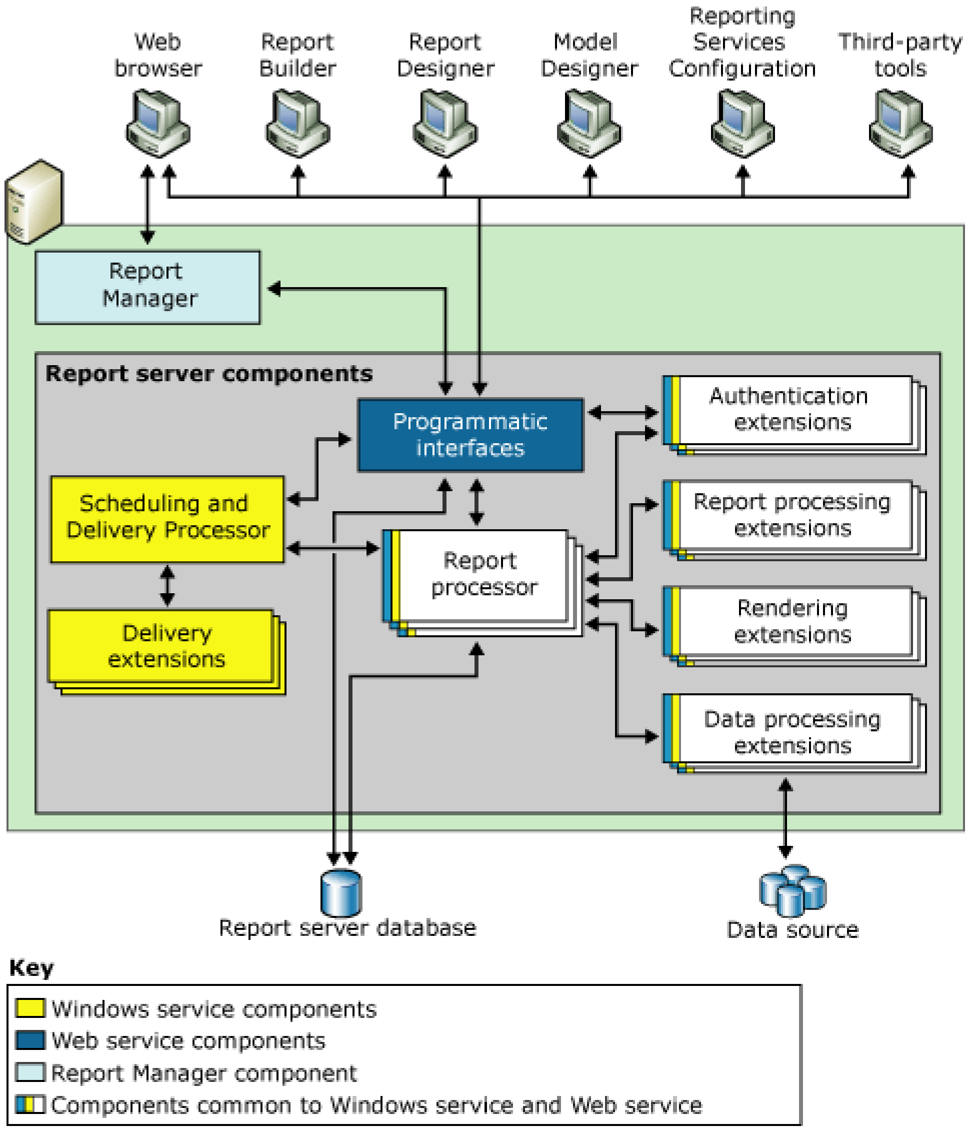
Although Reporting Services integrates with other Microsoft technologies out-of-the-box, developers and third-party vendors can build components to support additional report output formats, delivery formats, authentication models, and data source types. The development and run-time architecture was purposely created in a modular design to support third-party extension and integration opportunities.

SQL Server Reporting Services is a set of processing components, tools, and programmatic interfaces that support the development and use of rich reports in a managed environment. The tool set includes development tools, configuration and administration tools, and report viewing tools. Programmatic interfaces include Simple Object Access Protocol (SOAP), URL endpoints, and Windows Management Instrumentation (WMI) for easy integration with new or existing applications and portals.

Processing is distributed across multiple components. Central and specialized processors are used to retrieve data, process report layout, render presentation formats, and deliver to target destinations. Presentation processing occurs after the data is retrieved and is decoupled from data processing, allowing multiple users to review the same report simultaneously in formats designed for different devices or quickly change the viewing format of the report, from HTML to PDF or Microsoft Excel or XML, with a single click. The modular architecture is designed for extensibility. Developers can include reporting functionality in custom applications or extend reporting functionality to support custom features.

The following diagram shows the Reporting Services components and tools. The diagram also shows how custom tools fit into the overall design. It shows the flow of requests and data among the server components and which components send and retrieve content from a data store.

Reporting Services architecture diagram



Report Features

SQL Server Reporting Services provides a middle-tier server that runs under Microsoft Internet Information Services (IIS), allowing you to build a reporting environment on top of an existing Web server infrastructure. You can build reports that draw data from the data servers that you have in place for any data source type that has a Microsoft .NET Framework-managed data provider, OLE DB provider, or ODBC data source. You can build a wide range of reports that combine the strengths of Web-based features and traditional reporting. You can create interactive, tabular, or free-form reports that retrieve data at scheduled intervals or on-demand when the user opens a report. Matrix reports can summarize data for high-level reviews, while providing supporting detail in drilldown reports. Parameterized reports can be used to filter data based on values that are provided at run time. Reports can be rendered in both desktop and Web-oriented formats. You can choose from a variety of viewing formats to render reports on demand in preferred formats for data manipulation or printing. Reporting Services is server-based and thus provides a way to centralize report storage and management, provide secure access to reports and folders, control how reports are processed and distributed, and standardize how reports are used in your business. Reporting Services can be configured for high availability. You can install report servers on single-server, distributed, and clustered configurations.

This section describes the benefits of report definition and design, configuration and deployment, access and delivery, and programming features of Reporting Services. Developers who want to embed report functionality into Windows or Web applications might want to consider the **ReportViewer** controls as an alternative reporting solution. For more information about the controls, see Reporting Services and ReportViewer Controls in Visual Studio.

- **Relational, multidimensional, and XML data sources**. You can create reports that use relational and multidimensional data from SQL Server and Analysis Services. You can also use .NET Framework data providers to get data from Oracle and other databases. ODBC and OLE DB providers are also supported. You can use an XML data processing extension to retrieve data from any XML data source.
- **Tabular, matrix, chart, and free-form report layouts**. You can build tabular reports for column-based data, matrix reports for summarized data, chart reports for graphical data, and free-form reports for everything else. Free-form report layout is based on data regions, which enclose controls and fields in a container that you can move, nest, or arrange in a side-by-side layout. You can combine tabular, matrix, and charted information in a single report.
- **Ad hoc reports**. You can create and save reports directly to a report server using a ClickOnce application called Report Builder. Ad hoc reporting is supported through a thin client that is downloaded from the report server.
- **Drill-through reports and interactivity**. You can add interactive features by providing links to related reports and reports that provide supporting details. You can add scripted expressions in Microsoft Visual Basic.
- **Parameterized reports**. You can add parameters to refine a query or filter a dataset. Dynamic parameters get values at run time based on user selections (the selection of one parameter builds the value list for a second parameter).
- **Presentation formats**. Choose a presentation format when you open the report, or after you open the report. You can choose Web-oriented, page-oriented, and desk-

top application formats. Formats include HTML, MHTML, PDF, XML, CSV, TIFF, and Excel.

- **Custom controls or report items**. You can embed custom controls or report items that you create or purchase from a third-party vendor. A custom control requires a custom report processing extension.
- **Navigation**. You can add bookmarks and document maps to provide navigation options within a large report.
- **Aggregations**. You can aggregate and summarize data using controls and expressions. Aggregates include sum, average, min, max, count, and running totals.
- **Graphical elements.** You can embed or reference images and other resources that contain external content.

Report and Model Design

- **Report Designer**. Create reports in a full-featured report authoring application that is hosted within Business Intelligence Development Studio. You can use Report Designer to work with data, define a layout, preview a report, and publish a report to a test or production server. Report Designer includes query builders, an expression editor, and wizards so that you can follow step-by-step instructions to create a report. Report Designer also supports advanced reporting features for professional report designers who understand query languages and expression languages.
- **Model Designer**. Create models that support ad hoc reporting in Reporting Services. You can generate models automatically based on an existing schema or view, refine the model, and then publish it to a report server.
- **Report Builder.** Create ad hoc reports using templates and pre-organized data and save them directly to a report server. Reports that you create in Report Builder can be accessed, distributed, and managed just as you would any other report.

Deployment and Administration

- **Reporting Services Configuration**. Deploy and maintain an existing report server using a graphical user interface to configure service accounts, virtual directories, report server databases, encryption keys, and Web farm deployment.
- **Report Manager**. Configure role-based security and manage report server content by setting properties on data sources, reports, folders, resources, and report models. You can configure report execution, report history, and set limits on processing time, monitor and cancel pending or in-process reports, and create and manage data source connections and schedules independently of the reports that they are associated with.

- **Integration with SQL Server Management Studio, SQL Server Configuration Manager, Surface Area Configuration tools**. Report server administrators can use the tools provided in SQL Server 2005 to manage a Reporting Services installation. You can record scripts in Management Studio to replay routine maintenance tasks on other report servers.
- **Command-line utilities**. Command-line utilities support report server configuration, key management, and scripted operations.
- **Role-based security.** Use role-based security to control access to folders, reports, and resources. Security settings follow an inheritance pattern through the folder structure. You can vary security at any branch to redefine user access down to the item level.

Report Access and Delivery Features

- **On-demand access over Web connections**. Use a browser to navigate a folder hierarchy to find and work with reports and other items. You can reference reports from a Favorites list in a Web browser, or link from a Web portal.
- **Sharepoint Web parts**. Reporting Services provides two Web parts for report viewing and report server folder navigation. You can embed the Web parts in a Microsoft SharePoint Portal site to easily integrate with a report server deployment.
- **My Reports and My Subscriptions**. In Report Manager, you can store and manage reports and subscriptions in a personal workspace.
- **Subscriptions for E-mail or File share Delivery**. Automate report delivery through a standard subscription and set report presentation preferences. Users who prefer to view a report in Microsoft Excel, for example, can specify that format in a subscription. Deliver a rendered report to an e-mail inbox. Set delivery options that control whether the report is delivered as a link or attachment. Deliver a rendered report to a shared folder. Set delivery options that control whether the report is overwritten or added to an existing folder.
- **Data-driven subscriptions**. Automate report distribution through data-driven subscriptions, which generate a recipient list and delivery instructions at run time from an external data source. Use a query and column-mapping information to customize report output for a large number of users.

Programmabilty and Extensibility

- **Report Definition Language (RDL)**. RDL describes all possible elements of a report using an XML grammar that is validated by an XML schema. The report definition of an individual report is based on RDL and contains instructions for rendering the design at run time. RDL is extensible. You can add support for elements or features that are not present in the existing RDL schema, and then build custom tools and report rendering extensions to handle the features you create.

- **SOAP API**. Use Report Server Web service methods to access a report server and Report Server Web service programmatically.
- **URL access**. You can access report server items through parameterized URL strings. All reports and items stored in a report server are addressable through the report server namespace.
- **WMI provider**. Reporting Services includes a Windows Management Instrumentation (WMI) provider that you can use to manage the Report Server Windows service.
- **Extensible delivery, data processing, rendering, security, and report processing**. You can create custom delivery extensions to route reports to file shares, internal archive stores, or internal applications. You can extend data processing to query, convert, or transform data from new data source types. You can create custom rendering extensions to support report presentation in application formats or Web-oriented formats that are not provided with the product. You can build or integrate a security extension that provides an alternative to the Windows authentication model.

Microsoft SQL Server 2005/2008 Express Edition

Microsoft SQL Server 2005/2008 Express Edition with Advanced Services (SQL Server Express) is a free version of SQL Server 2005/2008 Express Edition that includes Reporting Services functionality. Reporting Services in SQL Server Express is different from other editions of Reporting Services in the following ways:

- Reporting Services in SQL Server Express includes a subset of the features that are available in other editions of SQL Server 2005/2008. You can use this topic to learn about the features that are part of the Express edition (features are based on the SQL Server 2005 Service Pack 1 version of Reporting Services).
- Product documentation for this edition is included in the SQL Server 2005/2008 Books Online Documentation Refresh 2. To get the documentation refresh, you must download it from a Microsoft Web site. After you download and install it, you can use the **SQL Server Express** filter in Books Online to hide the Reporting Services content that does not apply to this edition.
- In contrast with other editions of SQL Server 2005/2008, the AdventureWorks sample database is not installed or attached automatically. Setup does not install or attach the database for you. To get the database, you must download it from a Microsoft Web site. After you download and install it, you must attach it to the local SQL Server Express Database Engine instance. You can use SQL Server Management Express to

attach the database. For more information, see Installing Sample Databases for Express Editions.

- Data source connection strings and Reporting Services URLs have different default values in a SQL Server Express installation. SQL Server Express always installs as a named instance. Any URLs or connection strings must include the instance name. The following examples illustrate the syntax you should use

Item	**Example syntax**
Connection string to the sample AdventureWorks database hosted on a local SQL Server Express instance	Data Source=localhost\SQLExpress; Initial Catalog=AdventureWorks
URL to a report server and report server endpoint	http://localhost/reportserver$SQLExpress
URL to Report Manager	http://localhost/reports$SQLExpress

Important:
SQL Server Express includes SQL Server Management Studio Express. SQL Server Management Studio Express cannot be used to administer a report server. Use Report Manager and the Reporting Services Configuration tool instead.

Requirements

Reporting Services in SQL Server Express has all of the same software requirements as other editions of Reporting Services. Because the report server runs as an ASP.NET worker process in IIS, you must have a local instance of IIS 5.0 or later with ASP.NET 2.0 enabled on your computer. In addition, the Reporting Services Configuration tool requires Windows Management Instrumentation (WMI). If you disabled WMI, you cannot use the configuration tool. For more information about product requirements, see Hardware and Software Requirements (SQL Server Express).

Reporting Features in SQL Server Express

SQL Server Express provides the following Reporting Services functionality:

- On-demand report processing for each user who views a report. When a user opens a report, the report is initialized, the query is processed, data is merged into the report layout, and the report is rendered into a presentation format.
- Rendering formats are available for HTML, Acrobat, and Excel.
- Report data sources must be SQL Server relational databases that run locally in SQL Server Express.
- Report server management and report viewing are supported through Report Manager.
- Configuration is supported through the Reporting Services Configuration tool.

- Rs.exe, rsconfig.exe, and rskeymgmt.exe command line utilities are available in SQL Server Express.
- Windows authentication and predefined roles are used to map existing group and user accounts to a named collection of operations.

Unsupported Features

Other editions of SQL Server 2005/2008 include a larger set of Reporting Services features. The following list describes the features that are documented in SQL Server Books Online, but cannot be used in this edition:

- Scheduled report processing, caching, snapshots, subscriptions, and delivery are not supported.
- Analysis Services, Oracle, XML, SAP, SQL Server Integration Services (SSIS), OLE DB, and ODBC data sources are not supported.
- Remote data sources are not supported. Reports that are hosted in a SQL Server Express report server must retrieve SQL Server relational data from a local SQL Server Express Database Engine instance.
- Ad hoc reporting through semantic models and Report Builder is not supported.
- TIFF (Image), XML, and CSV rendering extensions are not supported.
- The Reporting Services API extensible platform for delivery, data processing, rendering, and security is not supported.
- Custom authentication extensions and custom role assignments are not supported. You must map existing Windows domain user and group accounts to predefined role definitions.
- Custom report items are not supported.
- Managing a long-running report process is not supported. Specifically, you cannot use the Manage Jobs feature in Report Manager to stop report processing.
- Scale-out deployment is not supported.

If you are using the User Instances feature (also known as "Run As Normal User" or RANU) in SQL Server Express, remember that the data source will not be accessible over remote connections. This means that if you configure a report data source for the User Instances feature, remote users who access the report will get the following error:

"An error has occurred during report processing. Cannot create a connection to data source '<datasourcename>'. For more information about this error navigate to the report server on the local server machine, or enable remote errors."

In addition, the report server log file will contain the following entry:

```
Cannot create a connection to data source '<datasourcename>'. --->
System.Data.SqlClient.SqlException: User does not have permission to perform this action.
```

To avoid these errors, do not set the **User Instance** property in the connection string or in the **Advanced Properties** dialog box when defining the connection.

How to Create Reports

To create reports, you must install the SQL Server Express Toolkit. It includes the version of Business Intelligence Development Studio that is used with the SQL Server Express edition. The SQL Server Express Toolkit is installed separately from other SQL Server Express components. For more information, search the Microsoft Web site for installation instructions for SQL Server Express with Advanced Services.

The reports that you create in the Express edition of Business Intelligence Development Studio can use all of the report definition features that you find in other editions of Reporting Services. For example, you can create drillthrough reports, subreports, and parameterized reports that include charts, tables, matrices, and lists.

You can reuse or copy report definitions that you created in other editions. Be aware that if you publish a report that you created in an earlier version of Reporting Services, the report will be upgraded to use the most recent SQL Server report definition format.

After you create a report definition, you can publish it to a report server. To make a published report available to users, you must use Report Manager to create role assignments that grant access to the report. For more information about creating, publishing, and securing reports, see Designing and Creating Reports, Publishing Reports to a Production Environment, and Creating, Modifying, and Deleting Role Assignments.

How to Deploy Reports and Use Report Server Functionality

To view published reports, you can use a browser or Report Manager, or create a custom application that uses a ReportViewer control to host reports.

Using Report Manager or a Browser

SQL Server Express includes Report Manager, a Web application that can be used by any user who wants to view published reports. Report Manager is accessed through a report server virtual directory that is configured on a local Web server. Users who have access to that directory can run Report Manager.

To view reports in Report Manager, you must define role assignments that allow users to view reports and navigate folders without granting access to server management features that a report server administrator might use. You can also use a Web browser to view a single published report.

Using Report Viewer controls

If you are a developer, you can use SQL Server Express with the Report Viewer controls, which are available through Microsoft Visual Studio. There are two controls: one for Windows Forms applications and one for ASP.NET applications. The controls are freely distributable with your application. You can configure the controls to run remote server reports that run on SQL Server Express.

The Report Viewer controls support other processing modes and deployment models in addition to SQL Server Express. For more information, see “ReportViewer Controls (Visual Studio)” in the Visual Studio product documentation on MSDN.

Programmability in SQL Server Express

The Report Server Web service can be accessed programmatically. Although you can access all of the report server SOAP endpoints, you cannot use all of the features. If you cannot use a method, Reporting Services returns an error.

SETTING UP REPORTING SERVICES

2

BASIC SQL 2008 EXPRESS INSTALLATION

This chapter will go through a basic SQL Server 2008 Express installation using the TRAVERSE 11 DVD. SQL 2008 Express is also available for download from Microsoft. For installation requirements, support and download information please visit the following Microsoft websites.

Support and System Requirements: http://www.microsoft.com/express/support/

Downloads: http://www.microsoft.com/express/sql/download/

Before you start, you will want to verify the following are installed or the installation will be stopped:

.Net 3.5 SP1

http://www.microsoft.com/downloads/details.aspx?FamilyID=AB99342F-5D1A-413D-8319-81DA479AB0D7&displaylang=en

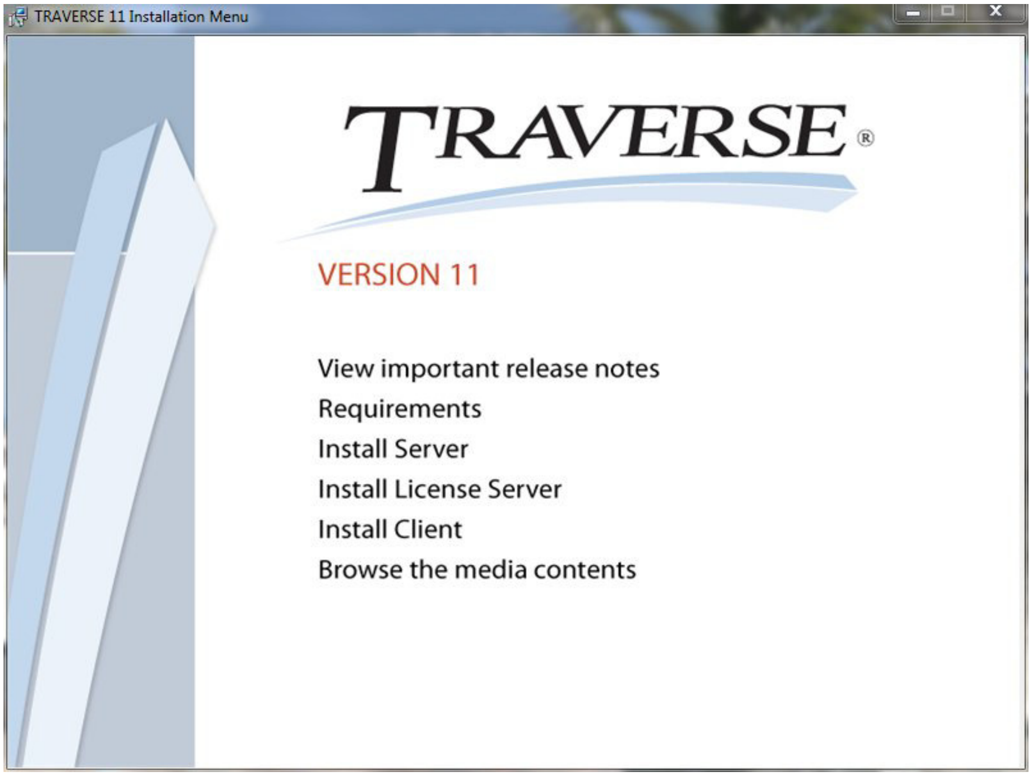
Windows Installer 4.5

http://www.microsoft.com/downloads/details.aspx?familyid=5A58B56F-60B6-4412-95B9-54D056D6F9F4&displaylang=en

Windows Powershell 1.0

http://www.microsoft.com/windowsserver2003/technologies/management/powershell/download.mspx

1. To install SQL Server Express from the TRAVERSE DVD, insert the DVD into the server DVD drive. The **TRAVERSE Startup** screen should appear.



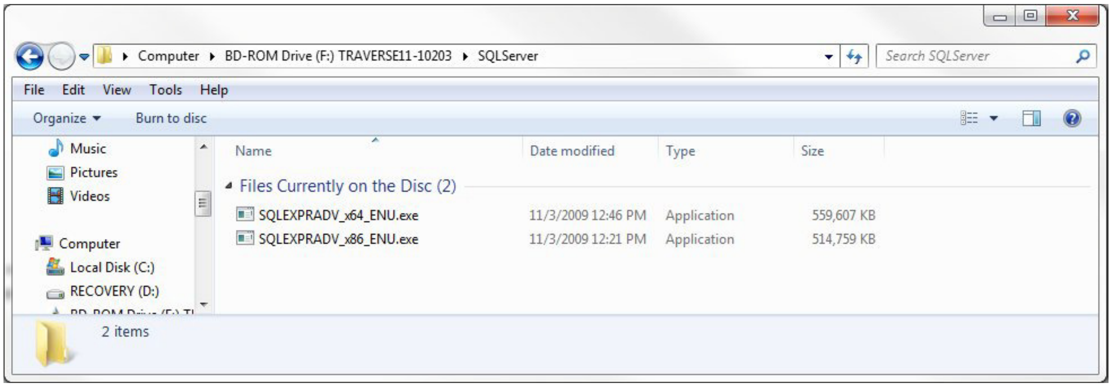
TRAVERSE 11 Installation Menu

TRAVERSE®

VERSION 11

View important release notes
Requirements
Install Server
Install License Server
Install Client
Browse the media contents

2. After reviewing the requirements and other documentation, click on **Browse the media contents** option and go to the **SQLServer** directory.



Computer ▸ BD-ROM Drive (F:) TRAVERSE11-10203 ▸ SQLServer

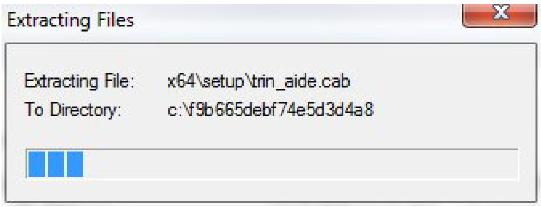
Files Currently on the Disc (2)

Name	Date modified	Type	Size
SQLEXPRADV_x64_ENU.exe	11/3/2009 12:46 PM	Application	559,607 KB
SQLEXPRADV_x86_ENU.exe	11/3/2009 12:21 PM	Application	514,759 KB

2 items

3. Double click on the appropriate version of SQL Server Express to install. Choose **SQLEXPRADV_x64_ENU.exe** if running on a server with a 64 bit OS. Choose **SQLEXPRADV_x86_ENU.exe** for servers running on a 32 bit OS.

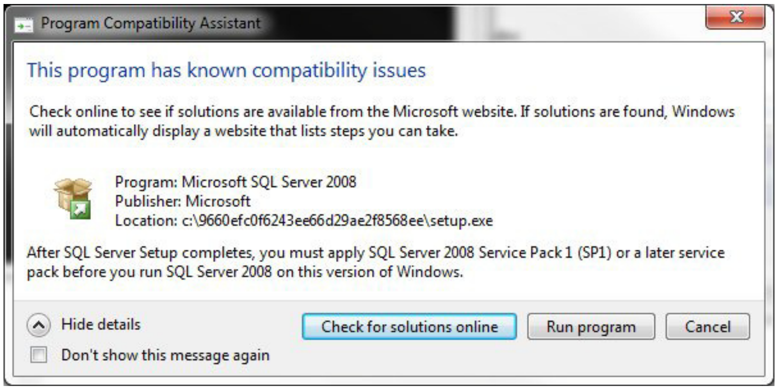
4. The installation should now start extracting files to the server.



Extracting Files

Extracting File: x64\setup\trin_aide.cab
To Directory: c:\f9b665debf74e5d3d4a8

5. Choose **Run Program** to start the installation.



Program Compatibility Assistant

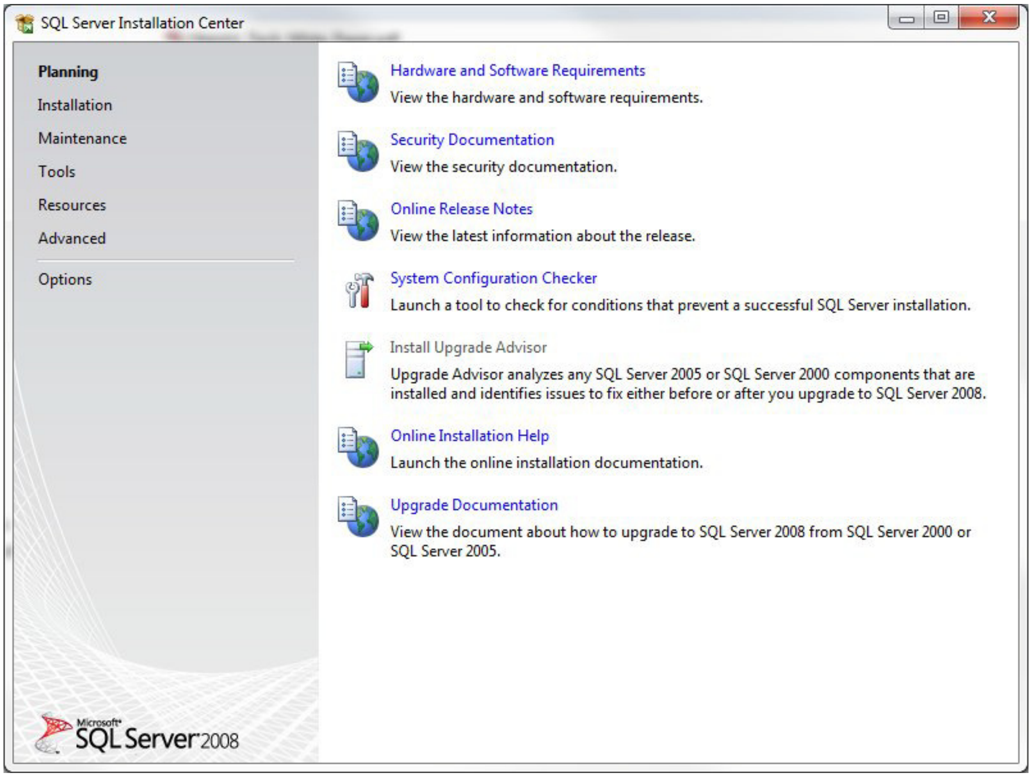
This program has known compatibility issues

Check online to see if solutions are available from the Microsoft website. If solutions are found, Windows will automatically display a website that lists steps you can take.

Program: Microsoft SQL Server 2008
Publisher: Microsoft
Location: c:\9660efc0f6243ee66d29ae2f8568ee\setup.exe

After SQL Server Setup completes, you must apply SQL Server 2008 Service Pack 1 (SP1) or a later service pack before you run SQL Server 2008 on this version of Windows.

6. Once the server is ready for installation, you should get the **SQL Server Installation Center** screen.



Hardware and Software Requirements
View the hardware and software requirements.

Security Documentation
View the security documentation.

Online Release Notes
View the latest information about the release.

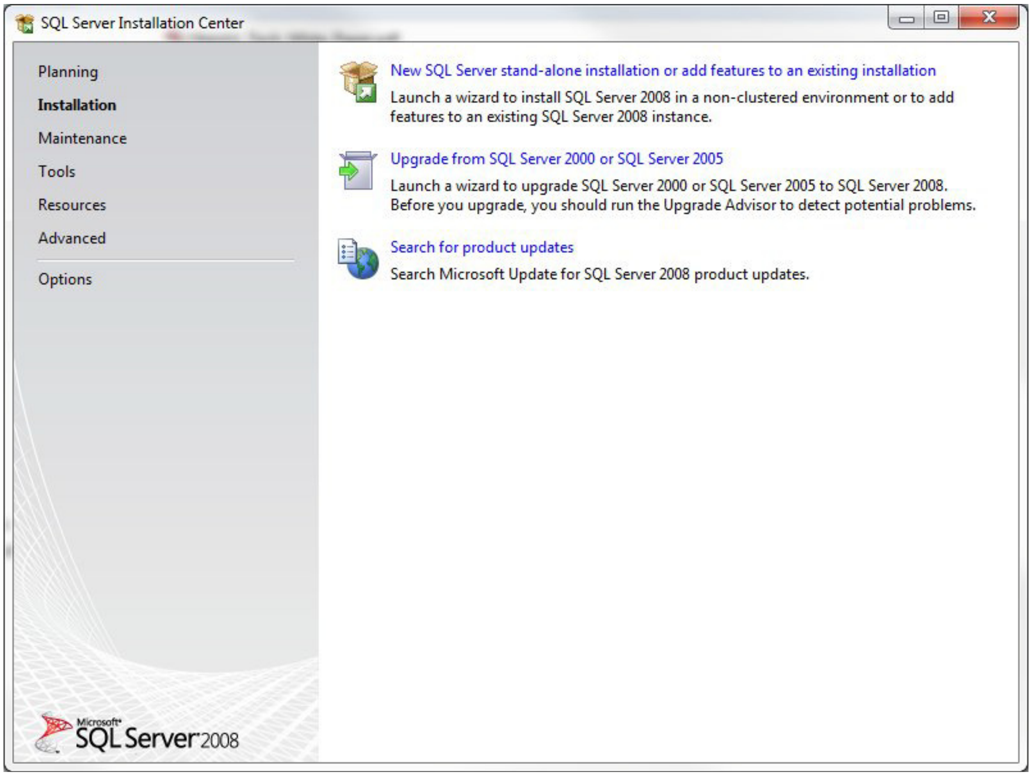
System Configuration Checker
Launch a tool to check for conditions that prevent a successful SQL Server installation.

Install Upgrade Advisor
Upgrade Advisor analyzes any SQL Server 2005 or SQL Server 2000 components that are installed and identifies issues to fix either before or after you upgrade to SQL Server 2008.

Online Installation Help
Launch the online installation documentation.

Upgrade Documentation
View the document about how to upgrade to SQL Server 2008 from SQL Server 2000 or SQL Server 2005.

7. Review the available documentation in the Planning section, then click on **Installation**.



SQL Server Installation Center

Planning
Installation
Maintenance
Tools
Resources
Advanced
Options

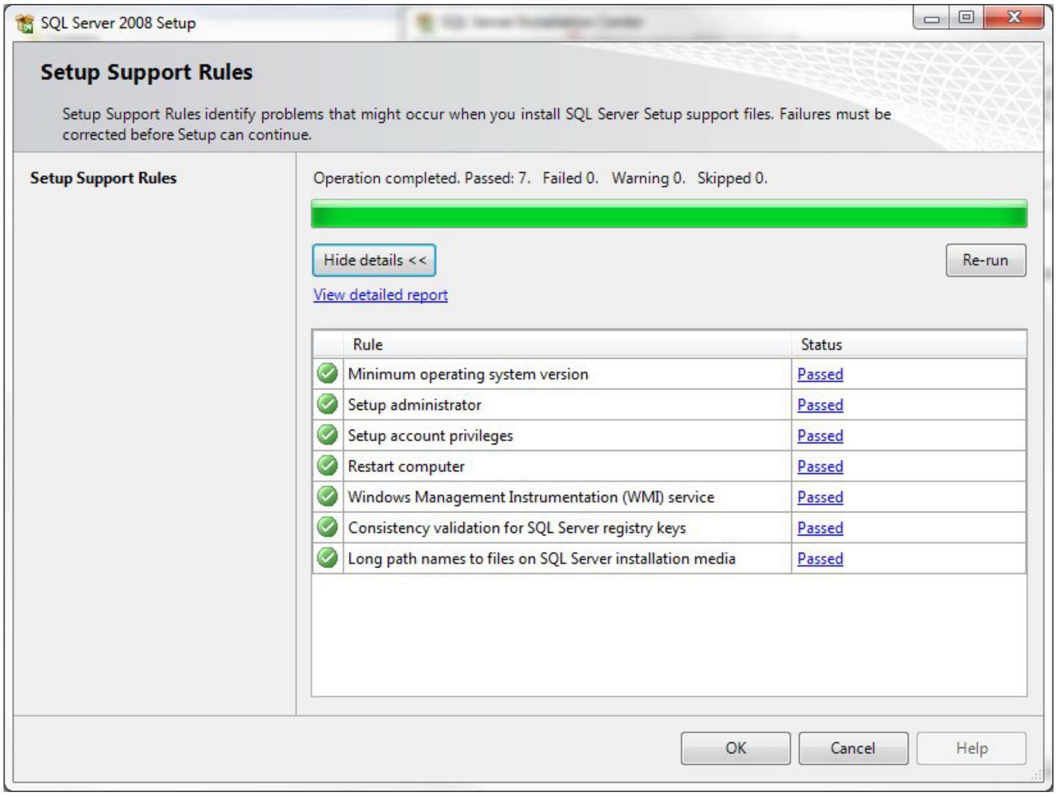
New SQL Server stand-alone installation or add features to an existing installation
Launch a wizard to install SQL Server 2008 in a non-clustered environment or to add features to an existing SQL Server 2008 instance.

Upgrade from SQL Server 2000 or SQL Server 2005
Launch a wizard to upgrade SQL Server 2000 or SQL Server 2005 to SQL Server 2008. Before you upgrade, you should run the Upgrade Advisor to detect potential problems.

Search for product updates
Search Microsoft Update for SQL Server 2008 product updates.

8. Choose **New SQL Sever stand-alone installation or add features to an existing installation** to begin the installation.

9. The **Setup Support Rules** screen should appear. Click on **View Details** to verify that all tests passed. Once all applicable tests have passed click **OK** to continue.



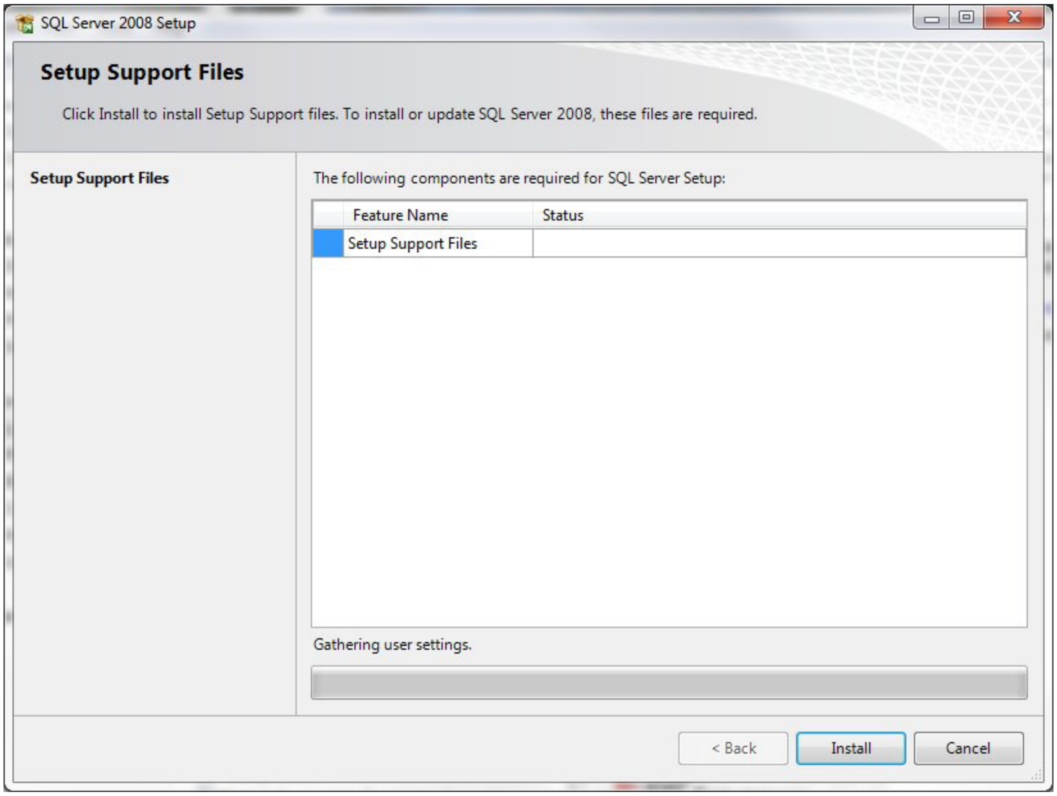
Setup Support Rules

Setup Support Rules identify problems that might occur when you install SQL Server Setup support files. Failures must be corrected before Setup can continue.

Operation completed. Passed: 7. Failed 0. Warning 0. Skipped 0.

	Rule	Status
✓	Minimum operating system version	Passed
✓	Setup administrator	Passed
✓	Setup account privileges	Passed
✓	Restart computer	Passed
✓	Windows Management Instrumentation (WMI) service	Passed
✓	Consistency validation for SQL Server registry keys	Passed
✓	Long path names to files on SQL Server installation media	Passed

10. The **Setup Support Files** install screen appears. Click **Install** to install the support files.



SQL Server 2008 Setup

Setup Support Files

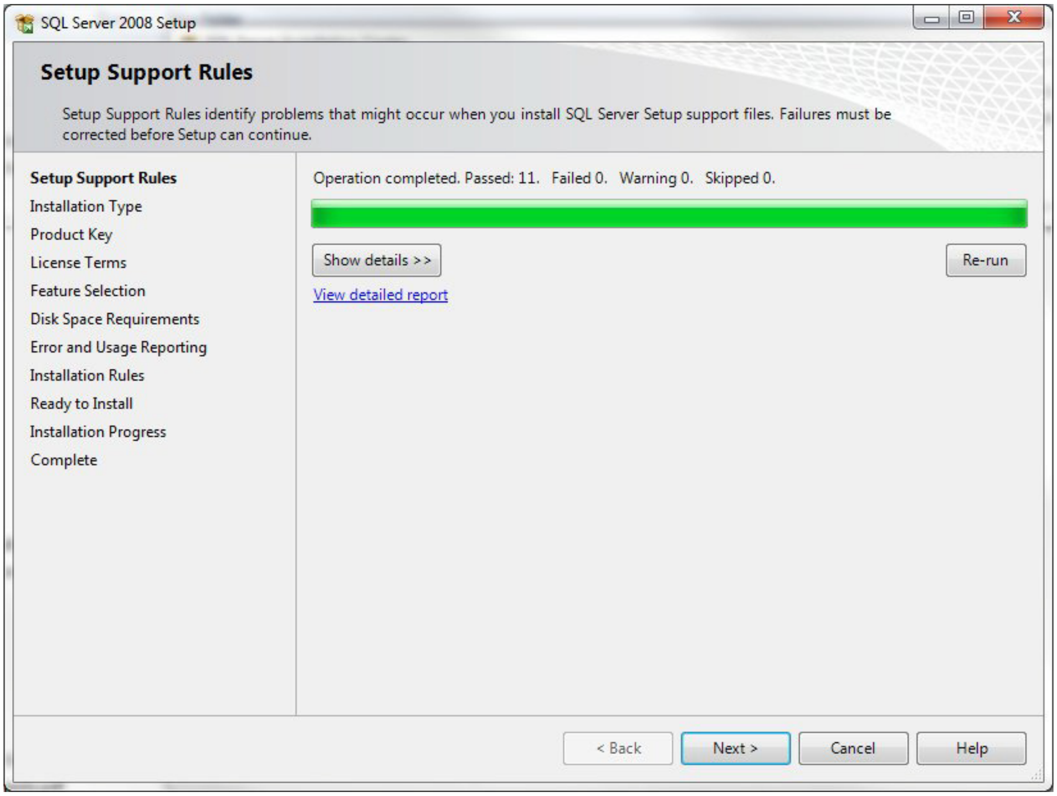
Click Install to install Setup Support files. To install or update SQL Server 2008, these files are required.

The following components are required for SQL Server Setup:

	Feature Name	Status
	Setup Support Files	

Gathering user settings.

11. The **Setup Support Rules** screen appears. Click **Next**.



SQL Server 2008 Setup

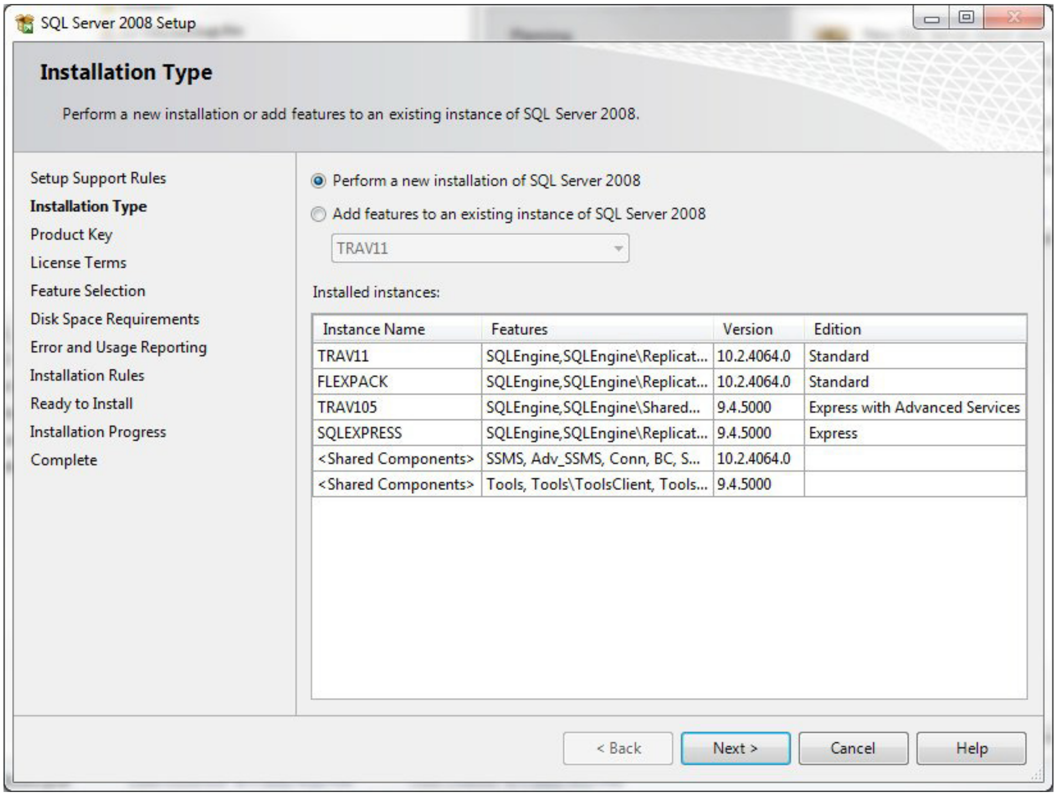
Setup Support Rules

Setup Support Rules identify problems that might occur when you install SQL Server Setup support files. Failures must be corrected before Setup can continue.

Setup Support Rules
Installation Type
Product Key
License Terms
Feature Selection
Disk Space Requirements
Error and Usage Reporting
Installation Rules
Ready to Install
Installation Progress
Complete

Operation completed. Passed: 11. Failed 0. Warning 0. Skipped 0.

12. The **Installation Type** screen appears.



Installation Type

Perform a new installation or add features to an existing instance of SQL Server 2008.

● Perform a new installation of SQL Server 2008

○ Add features to an existing instance of SQL Server 2008

TRAV11

Installed instances:

Instance Name	Features	Version	Edition
TRAV11	SQLEngine,SQLEngine\Replicat...	10.2.4064.0	Standard
FLEXPACK	SQLEngine,SQLEngine\Replicat...	10.2.4064.0	Standard
TRAV105	SQLEngine,SQLEngine\Shared...	9.4.5000	Express with Advanced Services
SQLEXPRESS	SQLEngine,SQLEngine\Replicat...	9.4.5000	Express
<Shared Components>	SSMS, Adv_SSMS, Conn, BC, S...	10.2.4064.0	
<Shared Components>	Tools, Tools\ToolsClient, Tools...	9.4.5000	

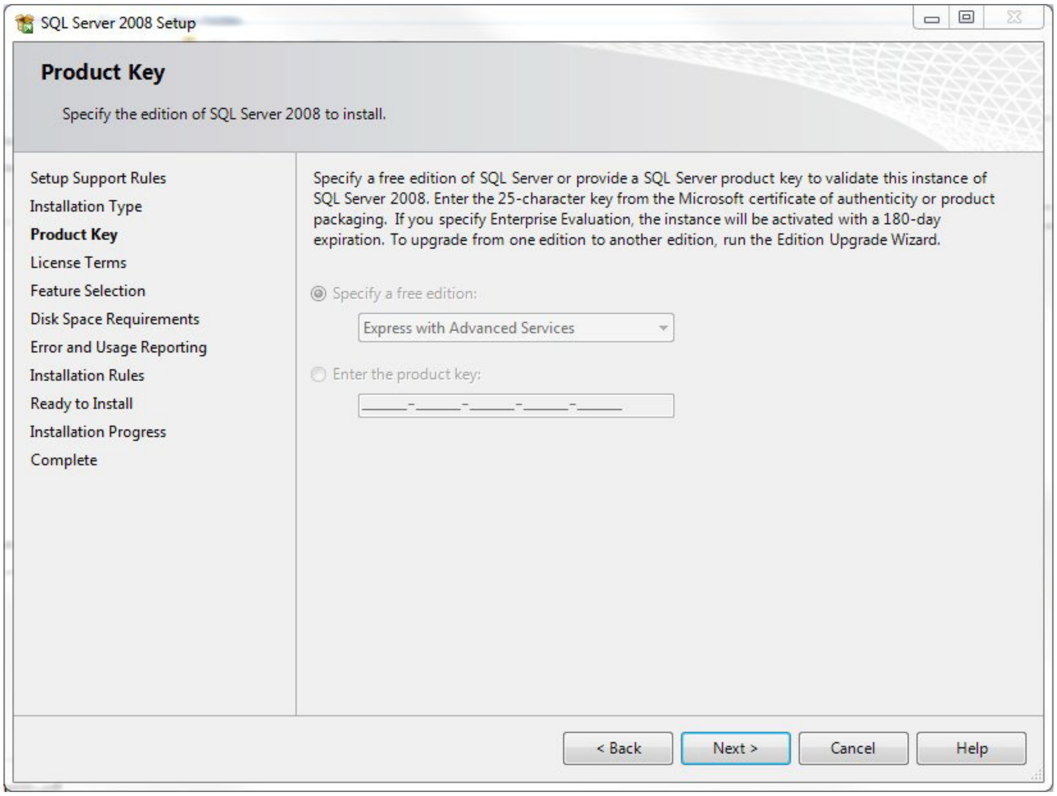
13. Select **"Perform a new installation of SQL Server 2008"** to install a new instance of SQL.

14. A list of the current instances of SQL on this computer is listed.

15. If you want to just add features to an existing instance select **"Add features to an existing instance of SQL Server 2008"** and select an instance.

16. Click **Next**.

17. The **Product Key** screen should appear.



SQL Server 2008 Setup

Product Key

Specify the edition of SQL Server 2008 to install.

Specify a free edition of SQL Server or provide a SQL Server product key to validate this instance of SQL Server 2008. Enter the 25-character key from the Microsoft certificate of authenticity or product packaging. If you specify Enterprise Evaluation, the instance will be activated with a 180-day expiration. To upgrade from one edition to another edition, run the Edition Upgrade Wizard.

● Specify a free edition:

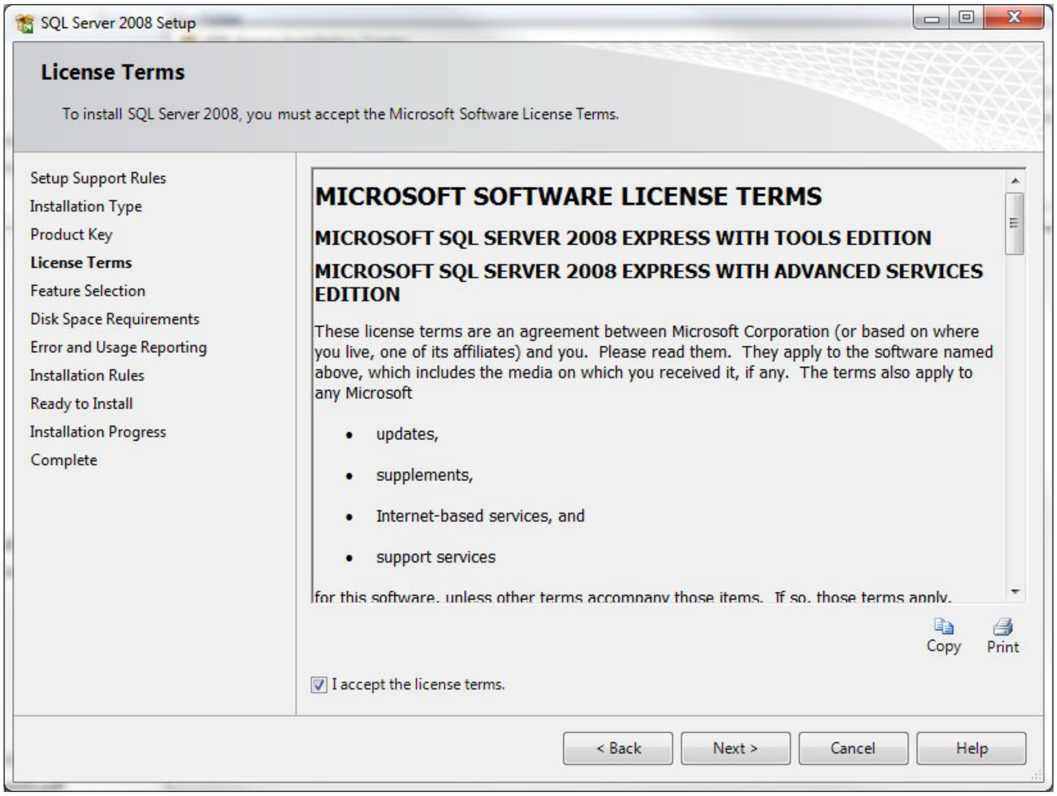
Express with Advanced Services

○ Enter the product key:

_____-_____-_____-_____-_____

18. This screen should be grayed out and the Specify a free edition should be automatically selected. Click **Next** to continue.

19. The **License Terms** screen should appear.



SQL Server 2008 Setup

License Terms

To install SQL Server 2008, you must accept the Microsoft Software License Terms.

MICROSOFT SOFTWARE LICENSE TERMS

MICROSOFT SQL SERVER 2008 EXPRESS WITH TOOLS EDITION

MICROSOFT SQL SERVER 2008 EXPRESS WITH ADVANCED SERVICES EDITION

These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

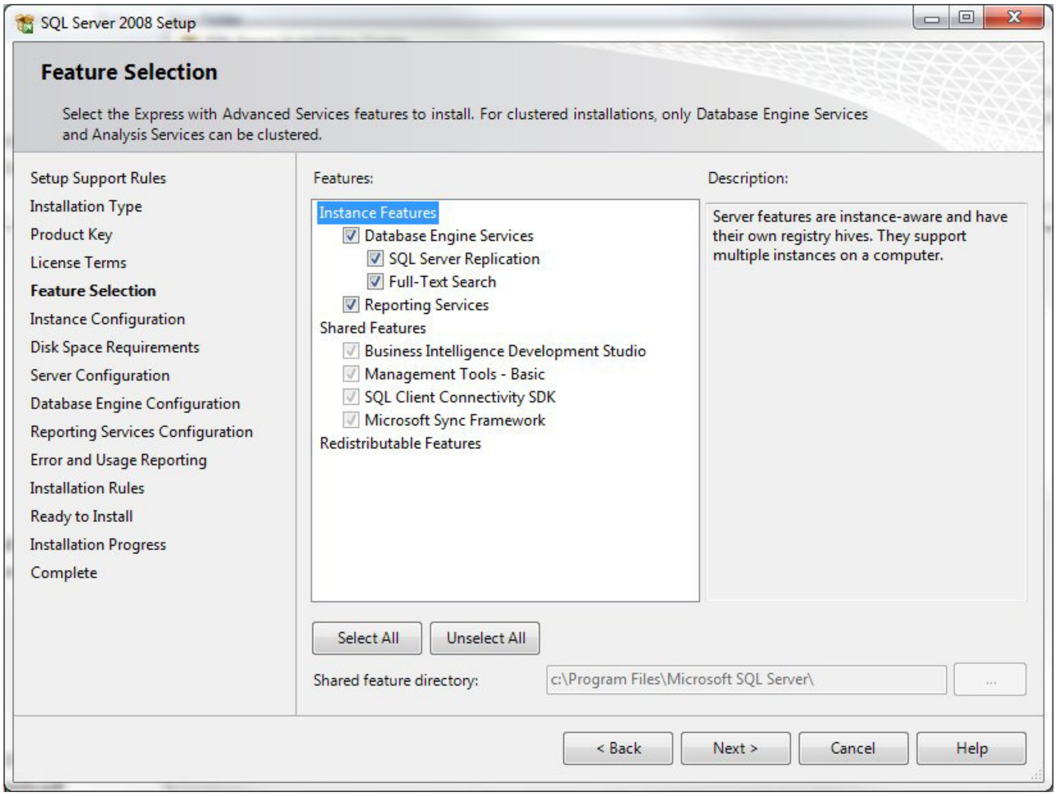
- updates,
- supplements,
- Internet-based services, and
- support services

for this software, unless other terms accompany those items. If so, those terms apply.

☑ I accept the license terms.

20. Review and **accept** the Terms and click **Next** to continue.

21. The **Feature Selection** screen appears.



SQL Server 2008 Setup

Feature Selection

Select the Express with Advanced Services features to install. For clustered installations, only Database Engine Services and Analysis Services can be clustered.

Setup Support Rules
Installation Type
Product Key
License Terms
Feature Selection
Instance Configuration
Disk Space Requirements
Server Configuration
Database Engine Configuration
Reporting Services Configuration
Error and Usage Reporting
Installation Rules
Ready to Install
Installation Progress
Complete

Features:

Instance Features
- ☑ Database Engine Services
  - ☑ SQL Server Replication
  - ☑ Full-Text Search
- ☑ Reporting Services

Shared Features
- ☑ Business Intelligence Development Studio
- ☑ Management Tools - Basic
- ☑ SQL Client Connectivity SDK
- ☑ Microsoft Sync Framework

Redistributable Features

Description:

Server features are instance-aware and have their own registry hives. They support multiple instances on a computer.

Shared feature directory: c:\Program Files\Microsoft SQL Server\

22. Check the **Database Engine Services** option. The Database Engine is the actual SQL install and is required if installing a new instance of SQL Express.

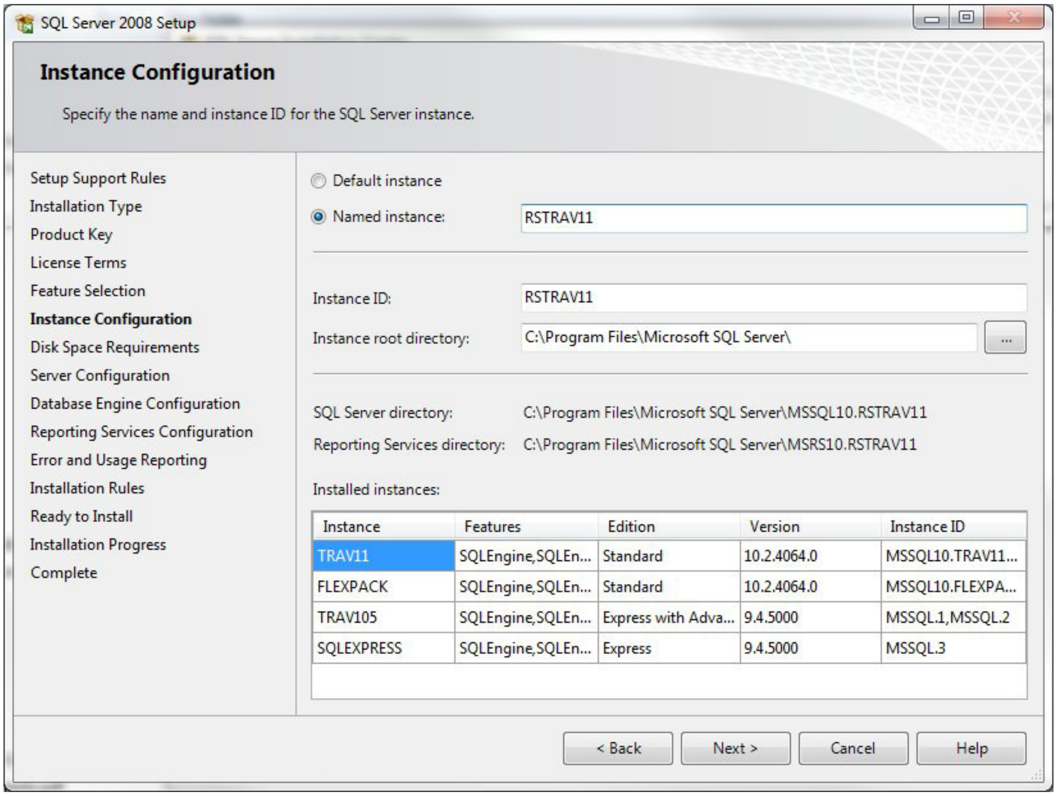
23. It is also advised to select the **Management Tools-Basic** option, as that will install the SQL Server Management Studio tool. This is not required, but it can be a useful tool as it allows a user interface for SQL Server management. It also allows access to the databases and the data within them, as well as offering a utility for managing SQL users, backups and security.

24. If you have **IIS services** installed, install **Reporting Services** if you plan on creating reports using SQL Reporting services.

25. The other options are advanced SQL tools and should only be selected if you are sure they are going to be required for your installation. SQL administrator knowledge could be required to use these tools. Each option will also use more resources on the server. Consult the Microsoft knowledgebase for more information on these options.

26. Click **Next** to continue.

27. The **Instance Configuration** screen appears.



Instance Configuration

Specify the name and instance ID for the SQL Server instance.

○ Default instance

● Named instance: RSTRAV11

Instance ID: RSTRAV11

Instance root directory: C:\Program Files\Microsoft SQL Server\ ...

SQL Server directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11

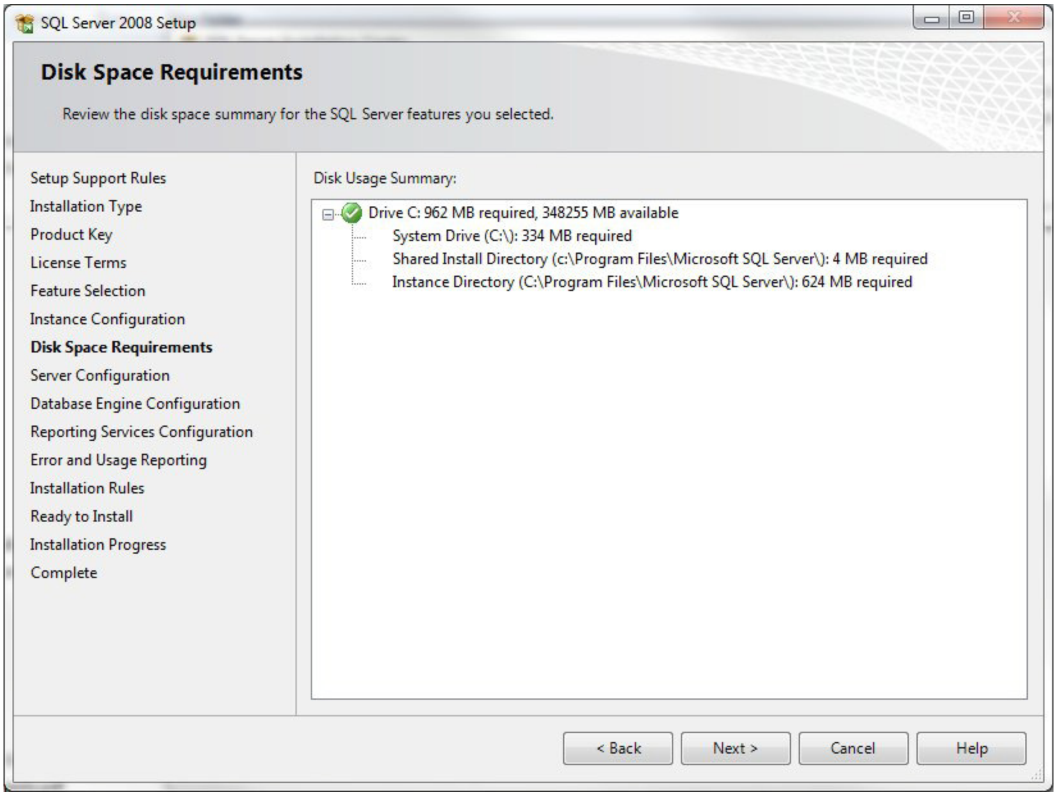
Reporting Services directory: C:\Program Files\Microsoft SQL Server\MSRS10.RSTRAV11

Installed instances:

Instance	Features	Edition	Version	Instance ID
TRAV11	SQLEngine,SQLEn...	Standard	10.2.4064.0	MSSQL10.TRAV11...
FLEXPACK	SQLEngine,SQLEn...	Standard	10.2.4064.0	MSSQL10.FLEXPA...
TRAV105	SQLEngine,SQLEn...	Express with Adva...	9.4.5000	MSSQL.1,MSSQL.2
SQLEXPRESS	SQLEngine,SQLEn...	Express	9.4.5000	MSSQL.3

28. If using the **Default instance**, the SQL server name when logging into TRAVERSE will be the **Windows Server Name\SqlExpress**. If using a Named Instance(recommended), the SQL server name when logging into TRAVERSE will be the Windows Server Name\Named instance, e.g. **Server\Trav11**.

29. The **Disk Space Requirements** screen appears.



Disk Space Requirements

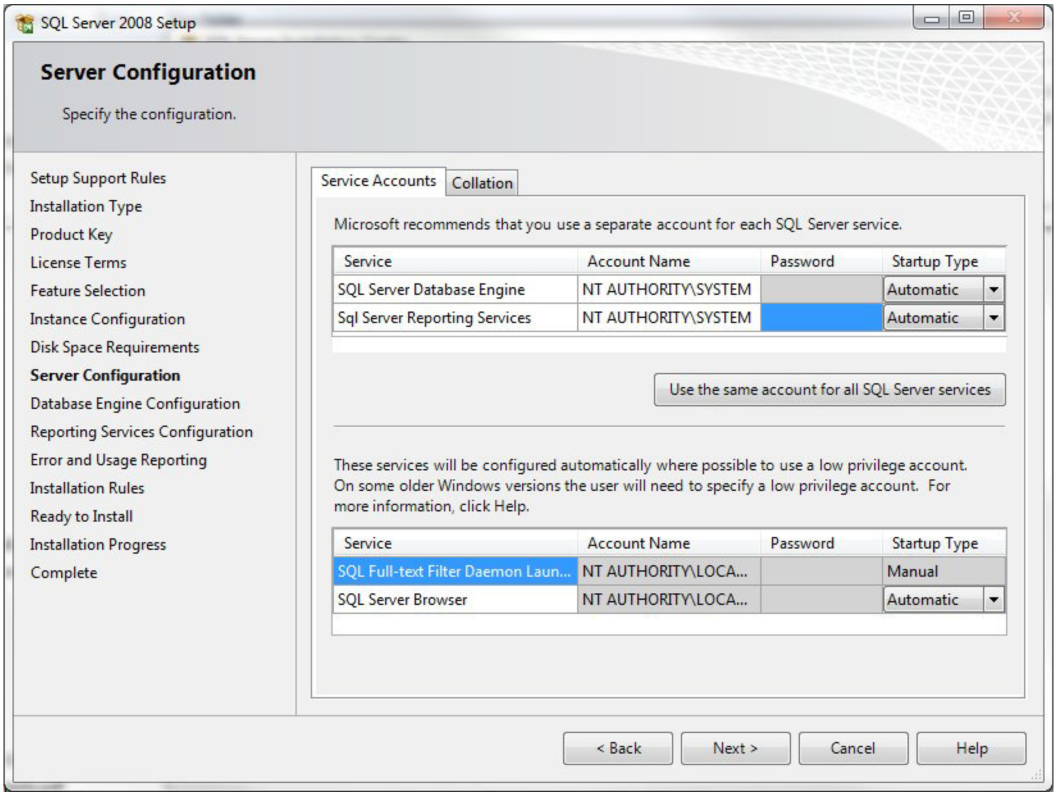
Review the disk space summary for the SQL Server features you selected.

Disk Usage Summary:

- ✓ Drive C: 962 MB required, 348255 MB available
  - System Drive (C:\): 334 MB required
  - Shared Install Directory (c:\Program Files\Microsoft SQL Server\): 4 MB required
  - Instance Directory (C:\Program Files\Microsoft SQL Server\): 624 MB required

30. Verify there is enough disk space and click **Next** to continue.

31. The **Server Configuration Screen** appears.



SQL Server 2008 Setup

Server Configuration

Specify the configuration.

Microsoft recommends that you use a separate account for each SQL Server service.

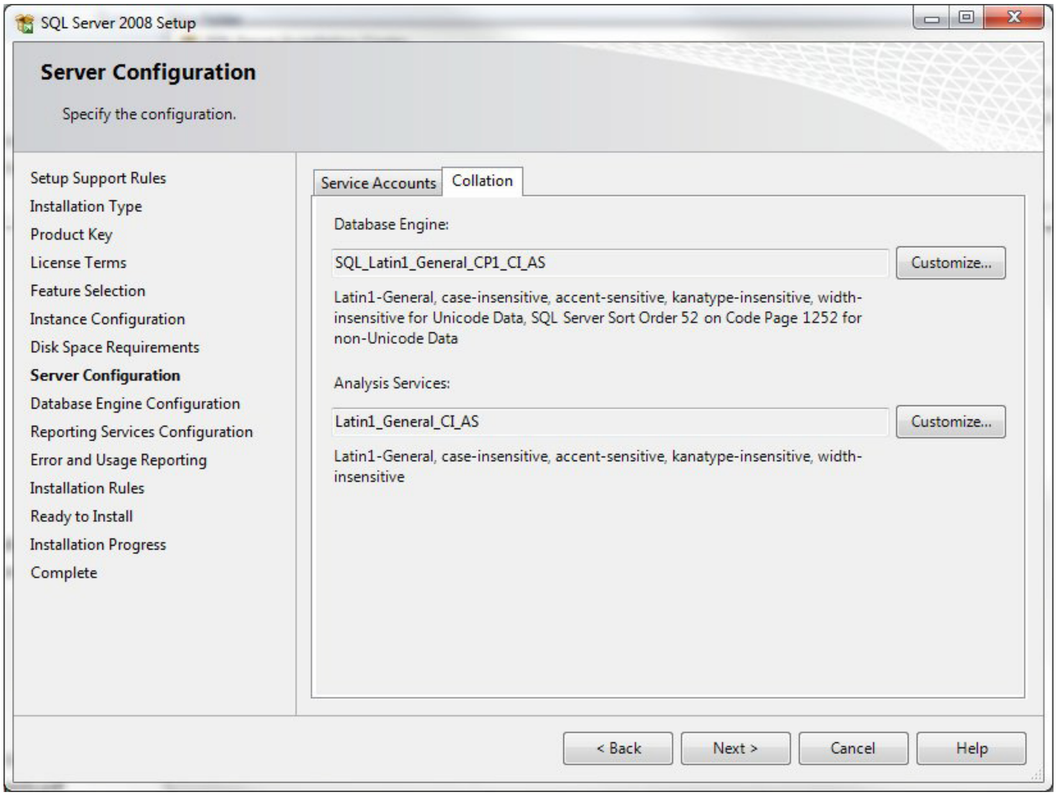
Service	Account Name	Password	Startup Type
SQL Server Database Engine	NT AUTHORITY\SYSTEM		Automatic
Sql Server Reporting Services	NT AUTHORITY\SYSTEM		Automatic

These services will be configured automatically where possible to use a low privilege account. On some older Windows versions the user will need to specify a low privilege account. For more information, click Help.

Service	Account Name	Password	Startup Type
SQL Full-text Filter Daemon Laun...	NT AUTHORITY\LOCA...		Manual
SQL Server Browser	NT AUTHORITY\LOCA...		Automatic

32. Choose the **Account Name** and **Startup Type**. **NT Authority\System** and **Automatic** should be acceptable for most installations.

33. Click on the **Collation** tab to verify that information is correct.



Database Engine:

SQL_Latin1_General_CP1_CI_AS

Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 52 on Code Page 1252 for non-Unicode Data

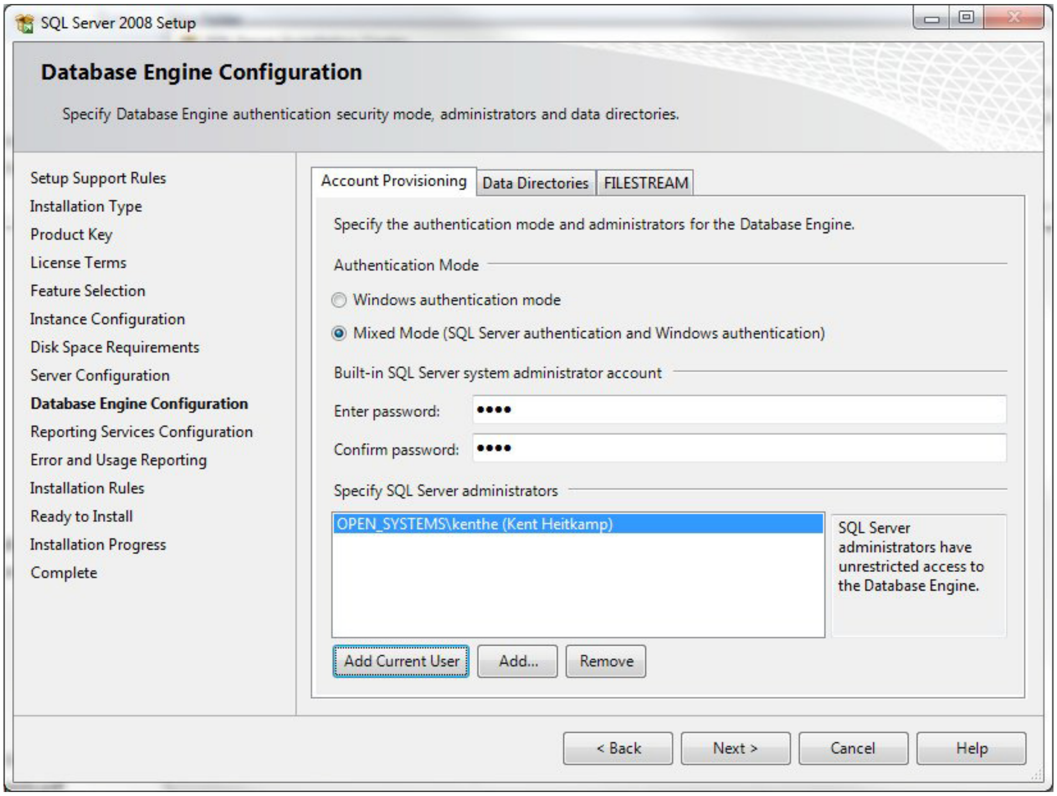
Analysis Services:

Latin1_General_CI_AS

Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive

34. If **SQL_Latin1_General_CP1_CI_AS** is not set for the engine by default, click on **Customize** and choose it. Click **Next** to continue.

35. The **Database Engine Configuration** screen appears.



SQL Server 2008 Setup

Database Engine Configuration

Specify Database Engine authentication security mode, administrators and data directories.

Specify the authentication mode and administrators for the Database Engine.

Authentication Mode

○ Windows authentication mode

● Mixed Mode (SQL Server authentication and Windows authentication)

Built-in SQL Server system administrator account

Enter password: ••••

Confirm password: ••••

Specify SQL Server administrators

OPEN_SYSTEMS\kenthe (Kent Heitkamp)

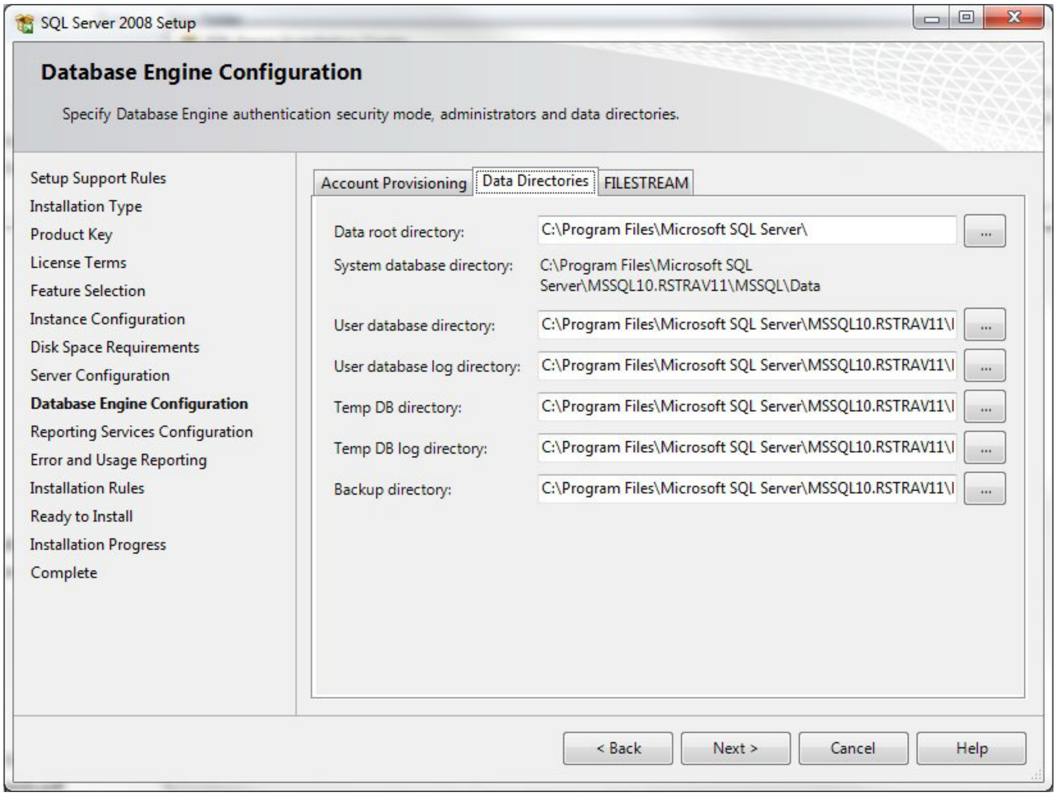
SQL Server administrators have unrestricted access to the Database Engine.

36. Choose **Windows authentication mode** to use only Windows logins for SQL access or Choose **Mixed Mode** to use both Windows and SQL logins. If you choose Mixed Mode, enter an **SA password** in the **Enter password** and **Confirm password** areas. Note this for future use and also note that these passwords may need to conform to your networks password rules.

NOTE: When installing SQL for TRAVERSE you must choose the Mixed Mode option. When setting up the databases for TRAVERSE we require a SQL master login that will be used to log into the SQL server.

37. Click on the **Add Current User** button to add the current Windows user as an administrator. Windows administrators are not considered SQL administrators by default on Windows Vista, Windows 7, Server 2003 or Server 2008 operating systems. If installing on one of these operating systems then you should click on the **Add Current User** button, and also add any other administrative users that should have SQL admin rights. This would be recommended for any users that are going to be responsible for SQL backups, SQL security setup, etc.

38. Click on the **Data Directories** tab.


SQL Server 2008 Setup

Database Engine Configuration

Specify Database Engine authentication security mode, administrators and data directories.

Data root directory: C:\Program Files\Microsoft SQL Server\

System database directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11\MSSQL\Data

User database directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11\I

User database log directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11\I

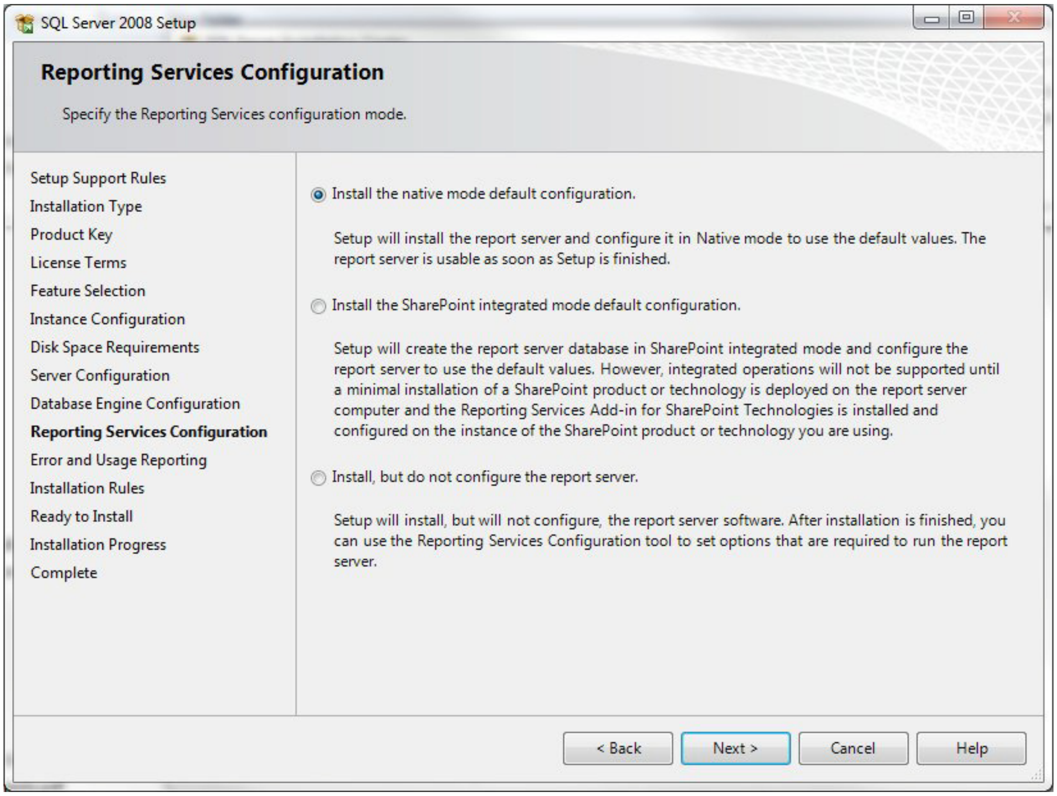
Temp DB directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11\I

Temp DB log directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11\I

Backup directory: C:\Program Files\Microsoft SQL Server\MSSQL10.RSTRAV11\I

39. Setup the **Database Directory** information. The default should be acceptable in most cases, however you may want to change this if you want to store the data on a drive other than C:\. The FILESTREAM tab should not need to have any changes made to it in most installations. See http://msdn.microsoft.com/en-us/library/cc949109.aspx for more information on the Filestream option. Click **Next** to continue.

40. The **Report Services Configuration** screen appears.



SQL Server 2008 Setup

Reporting Services Configuration

Specify the Reporting Services configuration mode.

● Install the native mode default configuration.

Setup will install the report server and configure it in Native mode to use the default values. The report server is usable as soon as Setup is finished.

○ Install the SharePoint integrated mode default configuration.

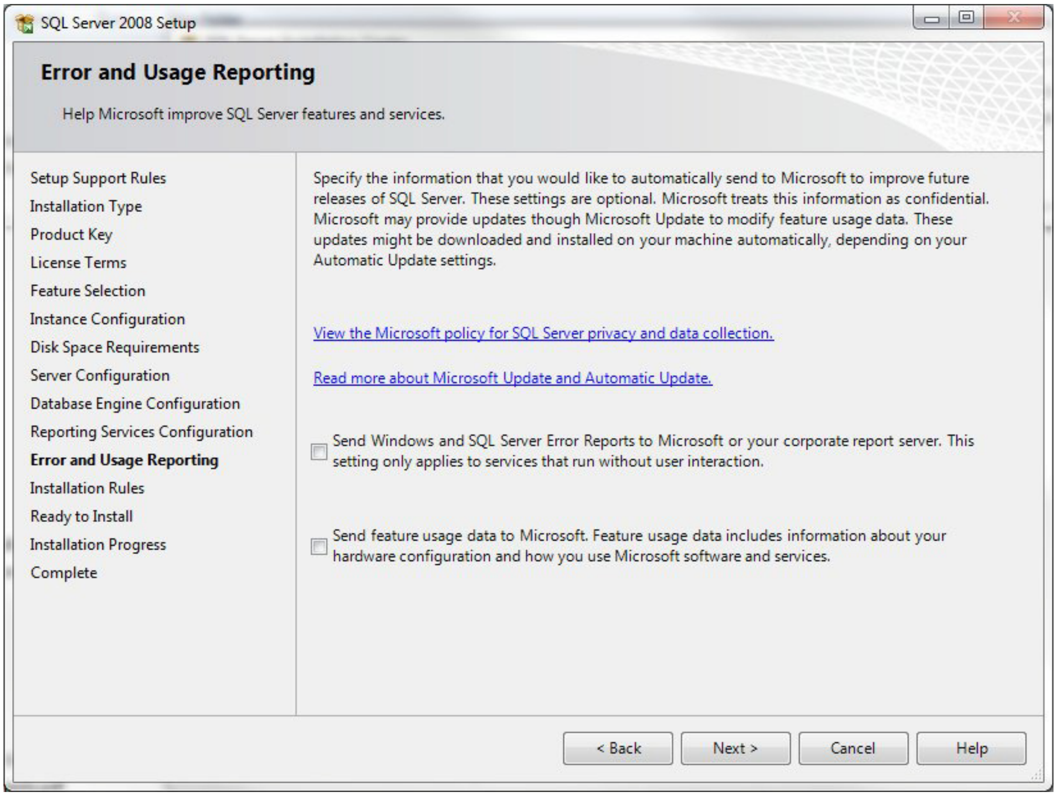
Setup will create the report server database in SharePoint integrated mode and configure the report server to use the default values. However, integrated operations will not be supported until a minimal installation of a SharePoint product or technology is deployed on the report server computer and the Reporting Services Add-in for SharePoint Technologies is installed and configured on the instance of the SharePoint product or technology you are using.

○ Install, but do not configure the report server.

Setup will install, but will not configure, the report server software. After installation is finished, you can use the Reporting Services Configuration tool to set options that are required to run the report server.

41. Select the **Install the native mode default configuration**. Click **Next** to continue.

42. The **Error and usage Reporting** screen appears.



Error and Usage Reporting

Help Microsoft improve SQL Server features and services.

Specify the information that you would like to automatically send to Microsoft to improve future releases of SQL Server. These settings are optional. Microsoft treats this information as confidential. Microsoft may provide updates though Microsoft Update to modify feature usage data. These updates might be downloaded and installed on your machine automatically, depending on your Automatic Update settings.

View the Microsoft policy for SQL Server privacy and data collection.

Read more about Microsoft Update and Automatic Update.

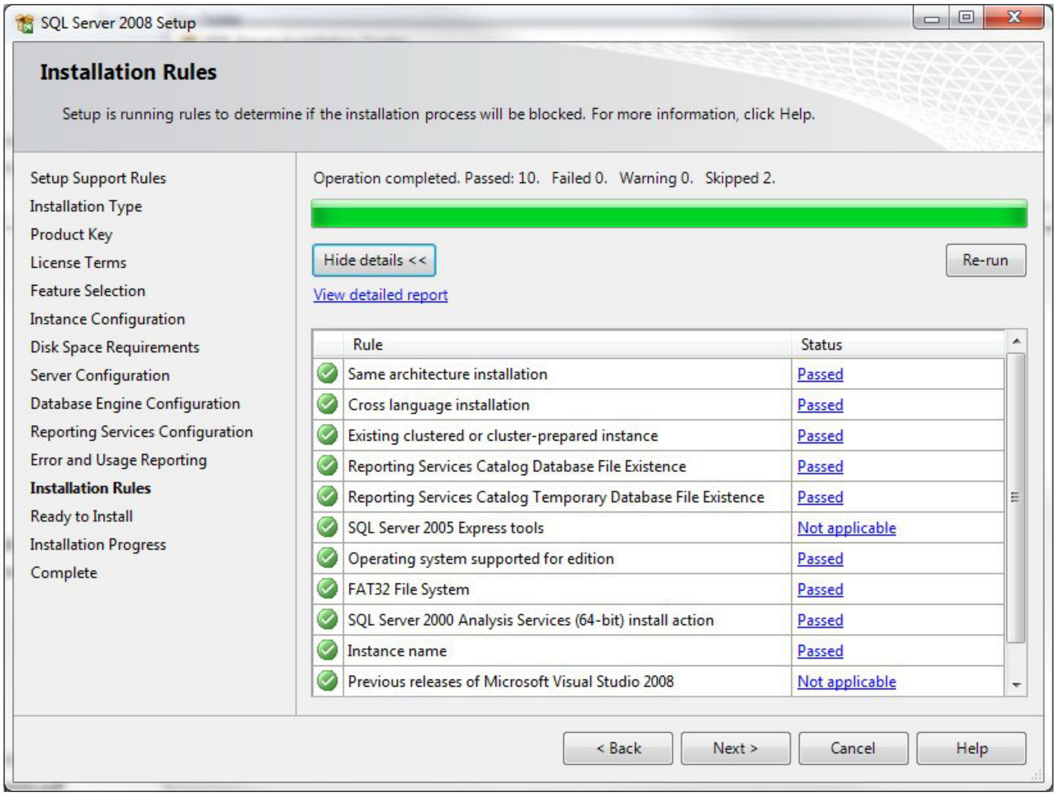
□ Send Windows and SQL Server Error Reports to Microsoft or your corporate report server. This setting only applies to services that run without user interaction.

□ Send feature usage data to Microsoft. Feature usage data includes information about your hardware configuration and how you use Microsoft software and services.

43. Here you can choose to send information to Microsoft about your SQL instance.

44. Click **Next** to continue.

45. The **Installation Rules** screen appears.



SQL Server 2008 Setup

Installation Rules

Setup is running rules to determine if the installation process will be blocked. For more information, click Help.

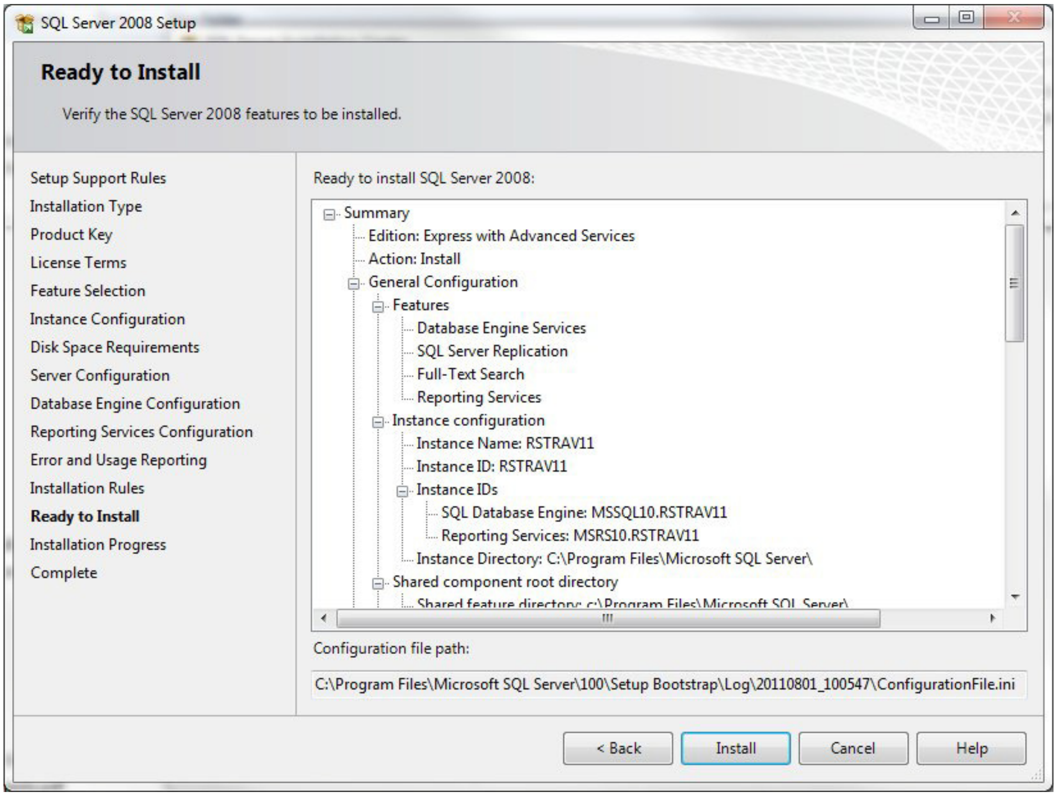
Operation completed. Passed: 10. Failed 0. Warning 0. Skipped 2.

	Rule	Status
✓	Same architecture installation	Passed
✓	Cross language installation	Passed
✓	Existing clustered or cluster-prepared instance	Passed
✓	Reporting Services Catalog Database File Existence	Passed
✓	Reporting Services Catalog Temporary Database File Existence	Passed
✓	SQL Server 2005 Express tools	Not applicable
✓	Operating system supported for edition	Passed
✓	FAT32 File System	Passed
✓	SQL Server 2000 Analysis Services (64-bit) install action	Passed
✓	Instance name	Passed
✓	Previous releases of Microsoft Visual Studio 2008	Not applicable

46. Verify that all applicable rules passed their test. If a test failed, consult the Microsoft Knowledgebase for more information on how to fix. Once all applicable tests have passed, click **Next** to continue.

NOTE: If you are installing the 2008 Management Tools on a server that also has 2005 SQL Management Studio installed, you must remove 2005 Management Studio prior to continuing on in the installation.

47. The **Ready to Install** screen appears.



SQL Server 2008 Setup

Ready to Install

Verify the SQL Server 2008 features to be installed.

- Setup Support Rules
- Installation Type
- Product Key
- License Terms
- Feature Selection
- Instance Configuration
- Disk Space Requirements
- Server Configuration
- Database Engine Configuration
- Reporting Services Configuration
- Error and Usage Reporting
- Installation Rules
- **Ready to Install**
- Installation Progress
- Complete

Ready to install SQL Server 2008:

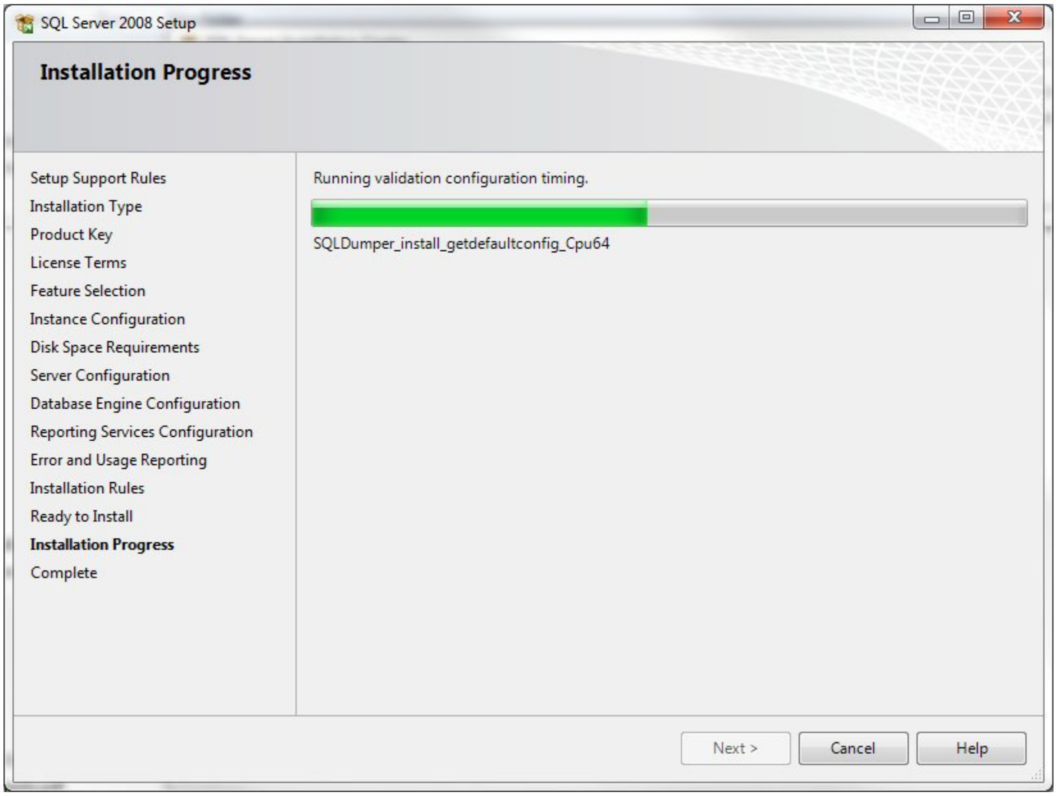
```
⊟ Summary
   Edition: Express with Advanced Services
   Action: Install
  ⊟ General Configuration
    ⊟ Features
       Database Engine Services
       SQL Server Replication
       Full-Text Search
       Reporting Services
    ⊟ Instance configuration
       Instance Name: RSTRAV11
       Instance ID: RSTRAV11
      ⊟ Instance IDs
         SQL Database Engine: MSSQL10.RSTRAV11
         Reporting Services: MSRS10.RSTRAV11
       Instance Directory: C:\Program Files\Microsoft SQL Server\
    ⊟ Shared component root directory
       Shared feature directory: c:\Program Files\Microsoft SQL Server\
```

Configuration file path:

C:\Program Files\Microsoft SQL Server\100\Setup Bootstrap\Log\20110801_100547\ConfigurationFile.ini

48. Verify all of the information is correct. Click **Back** to go back and make necessary changes or click **Install** to continue.

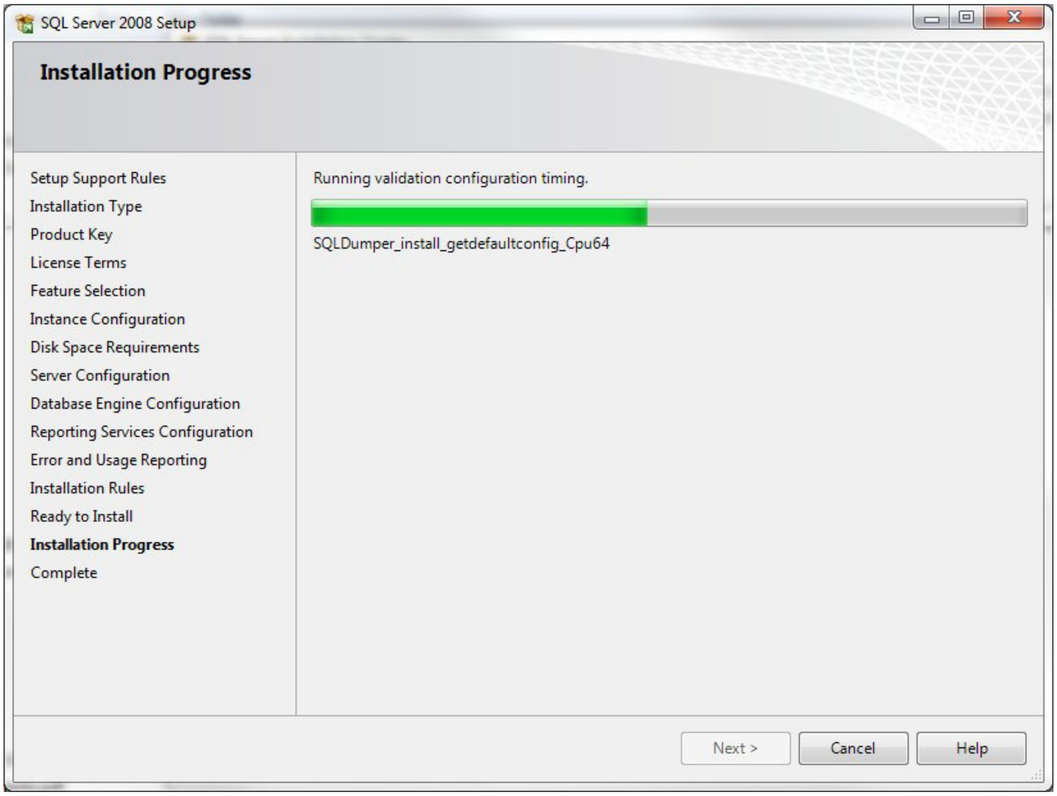
49. The **Installation Progress** screen appears while the installation is in progress. This may take a long time to complete.



Running validation configuration timing.

SQLDumper_install_getdefaultconfig_Cpu64

50. When the progress is complete, verify that all features were successfully installed.



SQL Server 2008 Setup

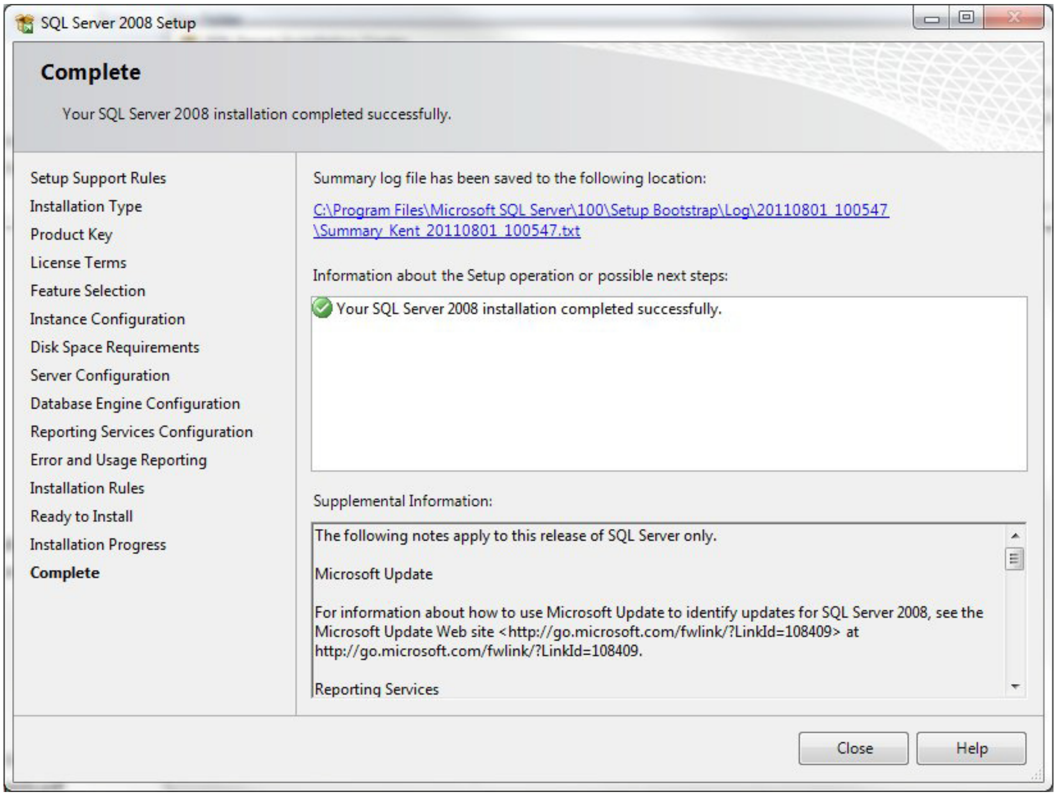
Installation Progress

Running validation configuration timing.

SQLDumper_install_getdefaultconfig_Cpu64

51. Click **Next** to continue.

52. The **Complete** screen should appear.



Your SQL Server 2008 installation completed successfully.

Summary log file has been saved to the following location:

C:\Program Files\Microsoft SQL Server\100\Setup Bootstrap\Log\20110801_100547
\Summary_Kent_20110801_100547.txt

Information about the Setup operation or possible next steps:

Your SQL Server 2008 installation completed successfully.

Supplemental Information:

The following notes apply to this release of SQL Server only.

Microsoft Update

For information about how to use Microsoft Update to identify updates for SQL Server 2008, see the Microsoft Update Web site <http://go.microsoft.com/fwlink/?LinkId=108409> at http://go.microsoft.com/fwlink/?LinkId=108409.

Reporting Services

Verify the installation was completed successfully and review the supplemental information. **Close** to finish the installation. You can now install TRAVERSE 11 using this instance.

NOTE: If after installation you have trouble connecting to the SQL Express Instance check these things:

On the server:

- Make sure ***SQL Service*** and ***SQL Server Browser Service*** are running in ***SQL Server Configuration Manager - SQL Server 2008 Services.***
- Make sure TCP\IP is enabled in ***SQL Server Configuration Manager - SQL 2008 Network Configuration - Protocols.***

- Make sure the server is allowed to accept remote connections. This can be checked in ***Management Studio*** by right clicking on the server name, choosing properties and going to ***Connections***.
- Verify Login information is correct. If trying a SQL login, verify that the server is using ***SQL Server and Windows Authentication mode***. This can be checked by going to ***Management Studio*** and right clicking on the server name, choosing properties and going to ***Connections***.

On Vista clients unable to connect:

- Make sure TCP\IP is enabled and there are no aliases setup in ***SQL Server Configuration Manager -SQL Native Client Configuration - Protocols.***
- If this option is unavailable, go to Start -Run and type in ***cliconfg*** (SQL Server Client Network Utility) to verify these settings.

On XP clients unable to connect:

- Make sure TCP\IP is enabled and there are no aliases setup by going to Start -Run and typing in ***cliconfg*** (SQL Server Client Network Utility)
- For SQL installed on Vista\XP sp2 with Firewall enabled (SQL 2008 SP2 must be installed in this setup)
- Add firewall exceptions for **ALL *sqlsrv.exe*** programs (located in Program Files\MS SQL Server\MSSQL.x\MSSQL\Binn).
- Add firewall exception for ***sqlbrowser.exe*** (located in Program Files\MS SQL Server\90\Shared).

If still having problems, then there may be other network issues. You can test by trying to use ODBC to connect to the SQL 2005 instance, if you still can't connect, then there are probably more advanced network or client OS problems.

THE REPORT SERVER CONFIGURATION

The TRAVERSE 10.5 installation CD includes an optional Report Viewer that can be used to manage and view reports created via SQL Reporting Services and/or Crystal Reports. To use this viewer, a workstation must have IIS (Internet Information Services) and Microsoft .Net Framework installed. You must also have SQL Server Reporting Services installed and configured correctly on the server if using reports generated via SQL Reporting Services. The following steps will go over some basic configuration information for setting up the report manager. For IIS configuration, SQL Server Reporting Services configuration (and report creation) as well as Crystal Report creation, extensive OS\Network\Crystal knowledge may be required.

For this class we will not be covering the Report Viewer setup and use. For more information refer to Chapter 5.

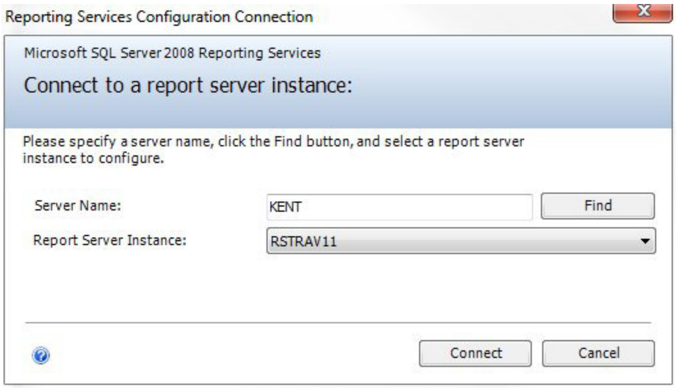
NOTE: The following section is an example of settings that may be used to get SQL Reporting Services configured correctly to allow use of the Viewer with SQL Reporting services reports. More extensive setup or configuration may be needed and additional Networking\Consulting resources may be required to configure this service correctly. Consult your network administrator, SQL Books Online or www.microsoft.com for more detailed information on SQL Reporting Services configuration.

SQL Reporting services is an optional component that can be installed when installing SQL 2005/2008 or SQL 2005/2008 Express with Advanced Services. This is a required component if you plan on using the TRAVERSE Report Viewer with SQL Reporting service. If using SQL 2000 or MSDE an additional download from Microsoft is required to install SQL Server Reporting Services.

NOTE: The following screen shots are from an existing test implementation of SQL Reporting Services on a server using SQL Server 2005/2008 Express. These configuration screenshots are meant to be used only as a reference, as this document is not a comprehensive configuration guide. Consult your network administrator, SQL Books Online or www.microsoft.com for more detailed information on SQL Reporting Services installation and configuration.

1. To start the configuration of the SQL Server Reporting services go to **Start** – **Programs** – **Microsoft SQL Server 2005/2008** – **Configuration Tools** – **Reporting Services Configuration**.

2. When the login screen comes up, enter the login information for the SQL Server and instance you installed Reporting Services to, and click **Connect**.


Reporting Services Configuration Connection
Microsoft SQL Server 2008 Reporting Services
Find
Connect
Cancel

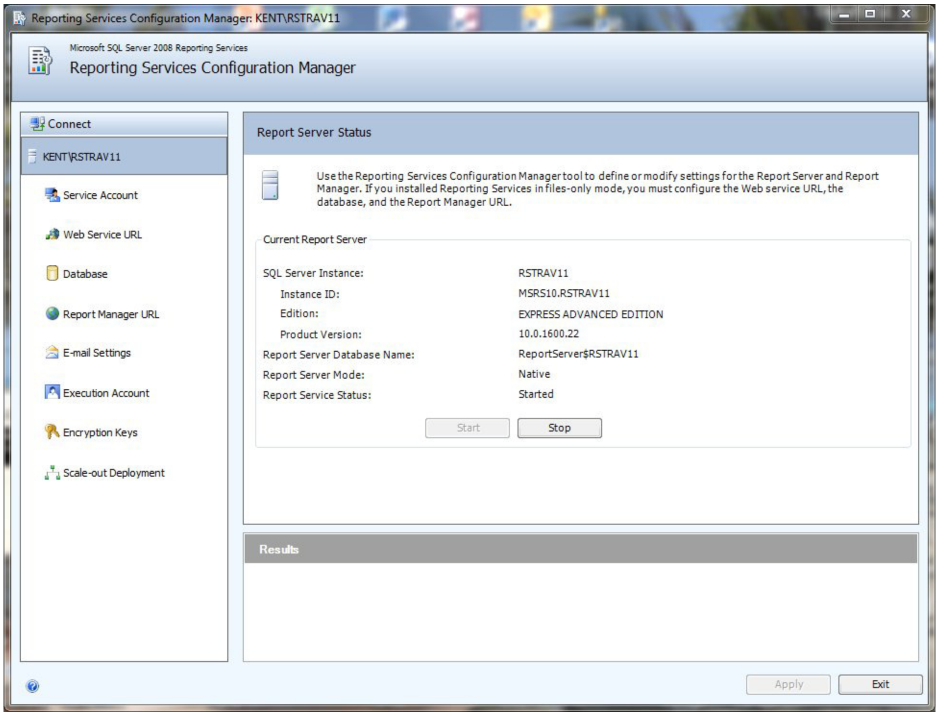
Connect to a report server instance:

Please specify a server name, click the Find button, and select a report server instance to configure.

Server Name: KENT

Report Server Instance: RSTRAV11

3. The **Configure Report Server** screen appears.



Report Server Status

Use the Reporting Services Configuration Manager tool to define or modify settings for the Report Server and Report Manager. If you installed Reporting Services in files-only mode, you must configure the Web service URL, the database, and the Report Manager URL.

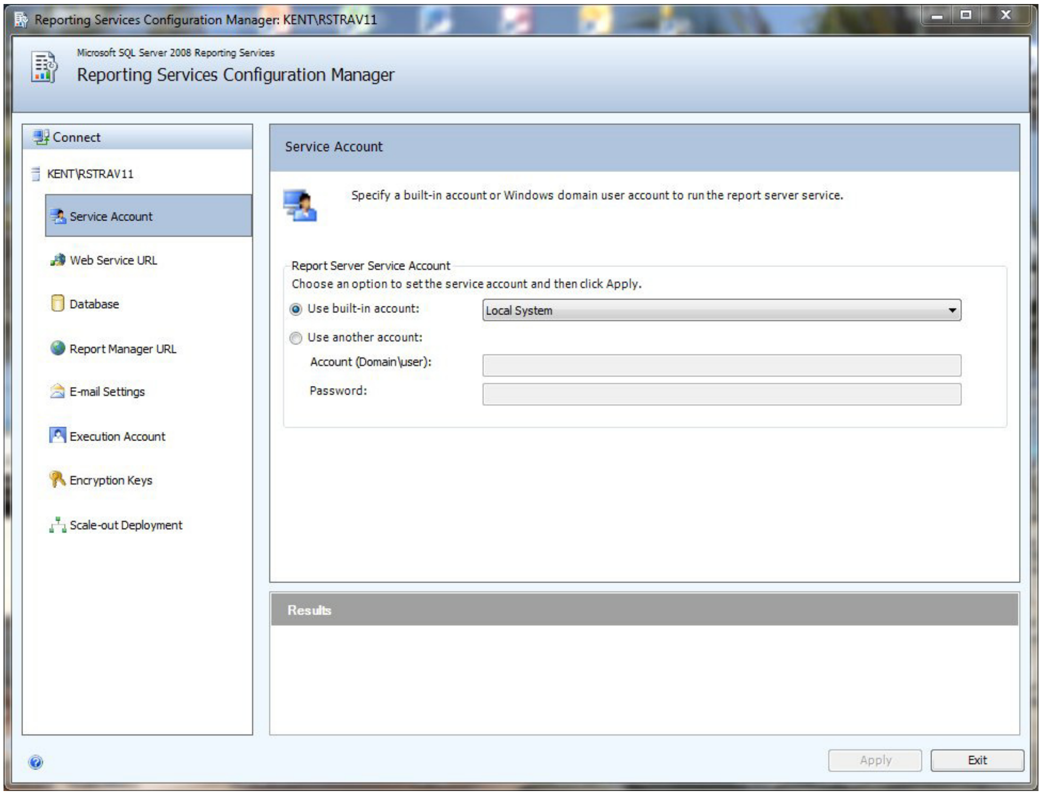
Current Report Server

SQL Server Instance: RSTRAV11
Instance ID: MSRS10.RSTRAV11
Edition: EXPRESS ADVANCED EDITION
Product Version: 10.0.1600.22
Report Server Database Name: ReportServer$RSTRAV11
Report Server Mode: Native
Report Service Status: Started

4. On the **Server Status** screen Start the Service if currently stopped. The service must be started for configuration to be set correctly.

5. If the service is not started click the **Start** button to start the reporting services service.

6. Click **Service Account** and select **Local System** for the **Use built-in account** selection.



Specify a built-in account or Windows domain user account to run the report server service.

Report Server Service Account

Choose an option to set the service account and then click Apply.

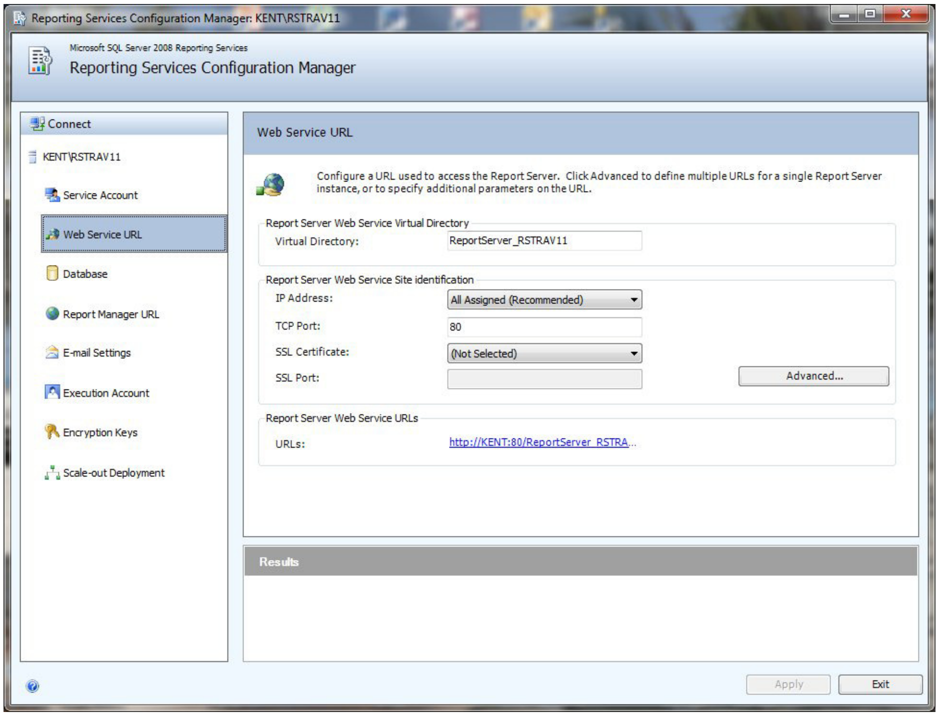
● Use built-in account: Local System

○ Use another account:

Account (Domain\user):

Password:

7. Click on the **Web Service URL**.



Web Service URL

Configure a URL used to access the Report Server. Click Advanced to define multiple URLs for a single Report Server instance, or to specify additional parameters on the URL.

Report Server Web Service Virtual Directory

Virtual Directory: ReportServer_RSTRAV11

Report Server Web Service Site identification

IP Address: All Assigned (Recommended)

TCP Port: 80

SSL Certificate: (Not Selected)

SSL Port:

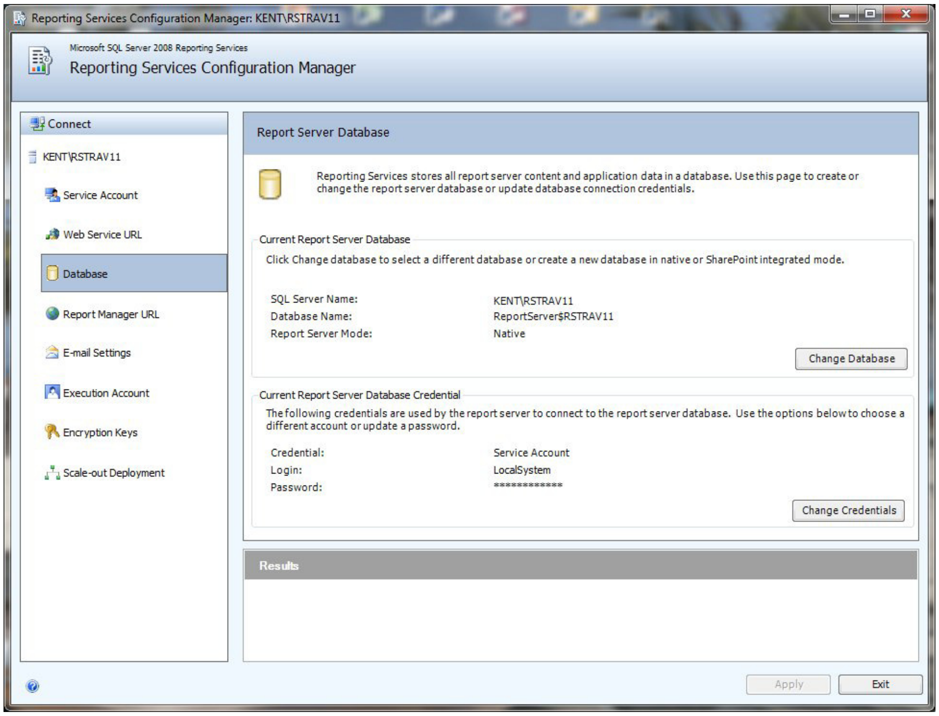
Report Server Web Service URLs

URLs: http://KENT:80/ReportServer_RSTRA...

8. Set the **Virtual Directory.** Leave the Report Server Web Service Site identification as the defaults and click **Apply**.

The above screen shot shows the default settings, which should be fine in most TRAVERSE installations. **Note the name for future use.**

9. Click on **Database Setup**.



Reporting Services stores all report server content and application data in a database. Use this page to create or change the report server database or update database connection credentials.

Current Report Server Database

Click Change database to select a different database or create a new database in native or SharePoint integrated mode.

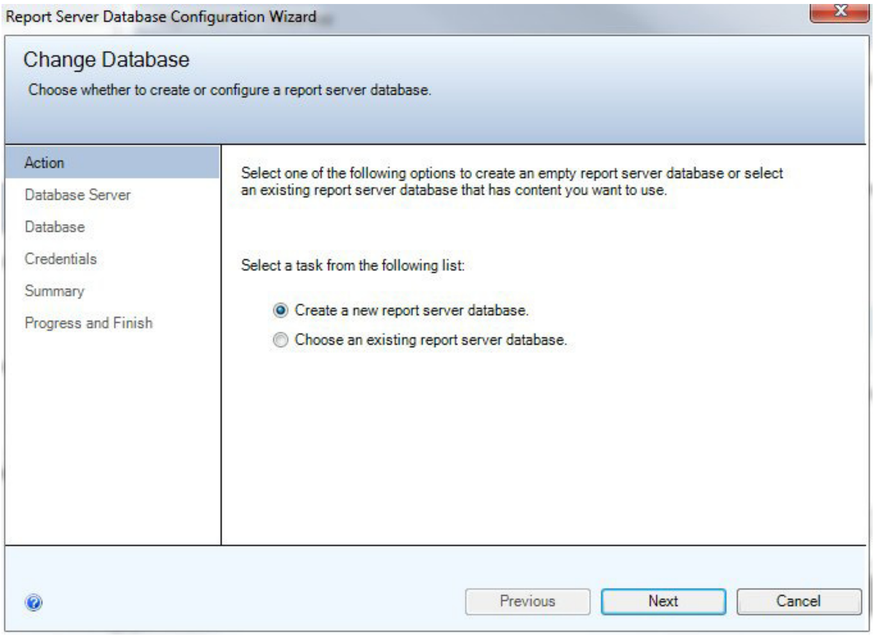
SQL Server Name: KENT\RSTRAV11
Database Name: ReportServer$RSTRAV11
Report Server Mode: Native

Current Report Server Database Credential

The following credentials are used by the report server to connect to the report server database. Use the options below to choose a different account or update a password.

Credential: Service Account
Login: LocalSystem
Password: ************

10. Click on **Change Database** and the following screen appears.



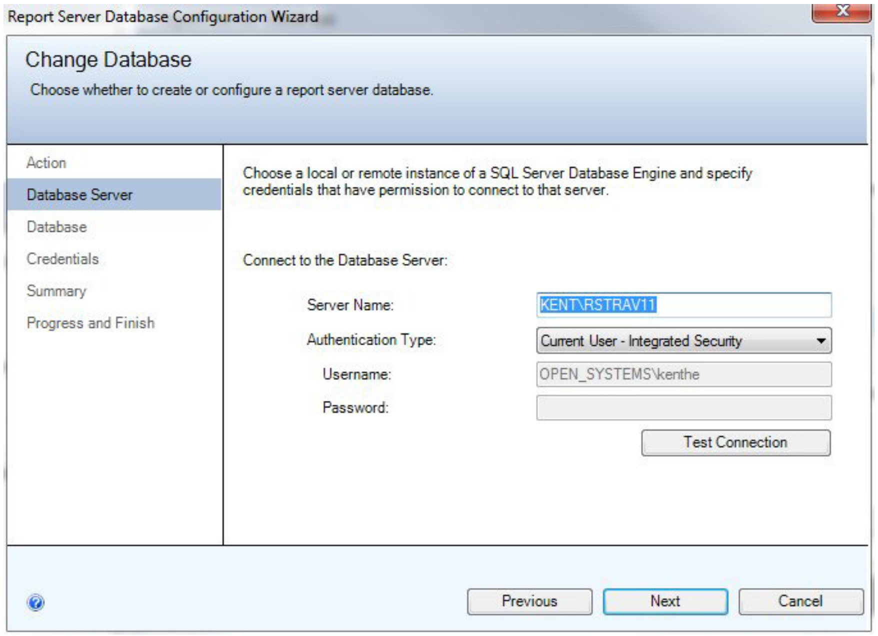
Choose whether to create or configure a report server database.

Select one of the following options to create an empty report server database or select an existing report server database that has content you want to use.

Select a task from the following list:

● Create a new report server database.
○ Choose an existing report server database.

11. Select “Create an new report server database” and click **Next**.



Report Server Database Configuration Wizard

Change Database

Choose whether to create or configure a report server database.

Choose a local or remote instance of a SQL Server Database Engine and specify credentials that have permission to connect to that server.

Connect to the Database Server:

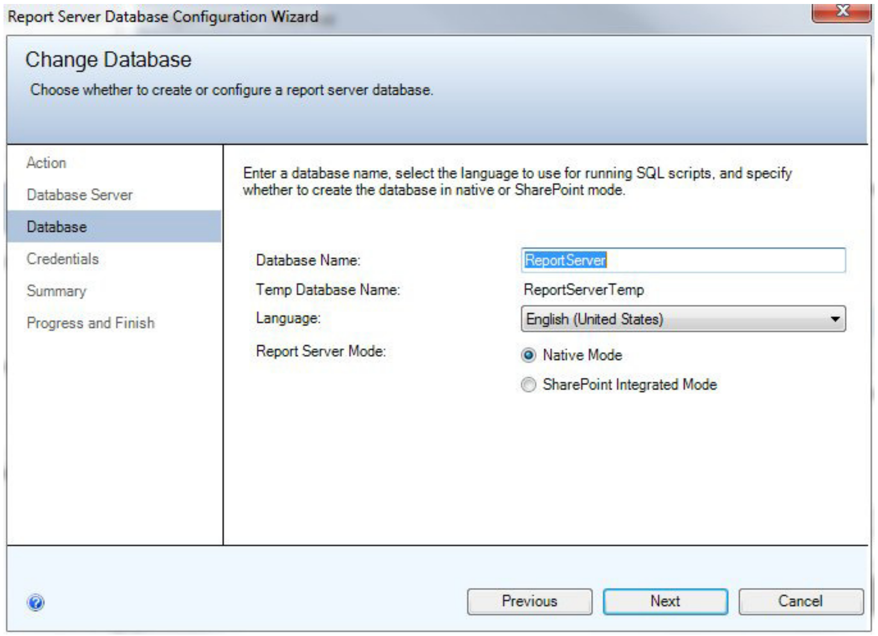
Server Name: KENT\RSTRAV11

Authentication Type: Current User - Integrated Security

Username: OPEN_SYSTEMS\kenthe

Password:

12. Enter the **SQL Server Name** and **instance** Reporting Services in installed on, **Authentication Type** (SQL Server Account works best), **Username** (sa works best), **Password** (sa password you entered when SQL was installed) and a **Database Name** (ReportServer is a good database name). This can be an existing blank database or if you type in a name that does not exist, it will be created for you. Click **Next** to go to the Database setup screen.



Report Server Database Configuration Wizard

Change Database

Choose whether to create or configure a report server database.

Enter a database name, select the language to use for running SQL scripts, and specify whether to create the database in native or SharePoint mode.

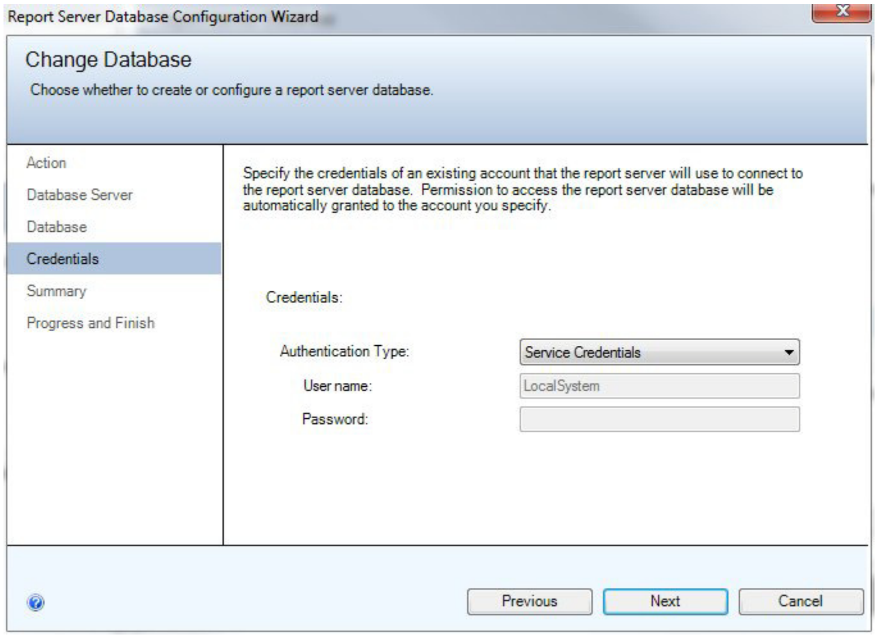
Database Name: ReportServer

Temp Database Name: ReportServerTemp

Language: English (United States)

Report Server Mode: ● Native Mode ○ SharePoint Integrated Mode

13. Click on **Next** to configure the connection credentials.



Report Server Database Configuration Wizard

Change Database

Choose whether to create or configure a report server database.

Specify the credentials of an existing account that the report server will use to connect to the report server database. Permission to access the report server database will be automatically granted to the account you specify.

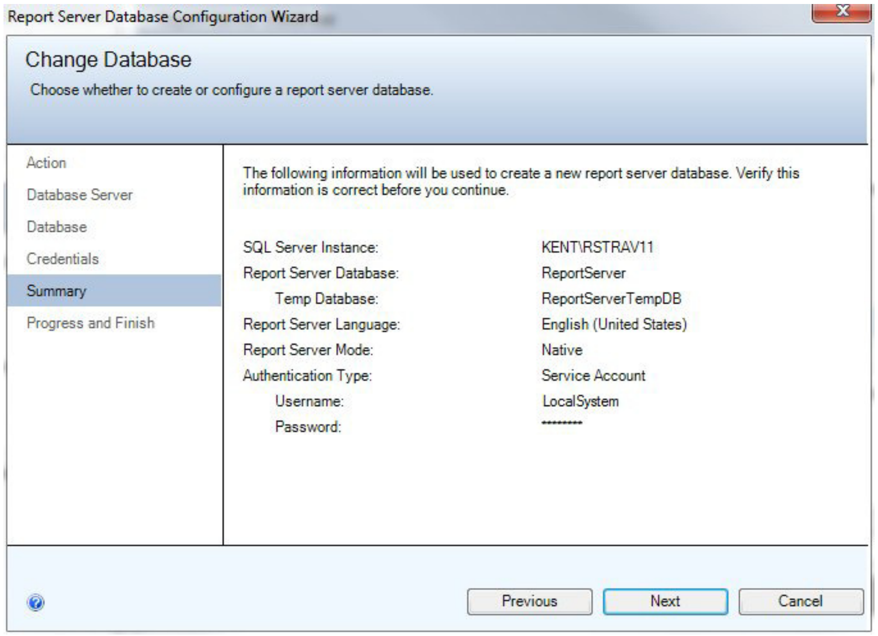
Credentials:

Authentication Type: Service Credentials

User name: LocalSystem

Password:

14. Select **Service Credentials** for the **Authentication Type**. Click Next to get to the Summary screen.



Report Server Database Configuration Wizard

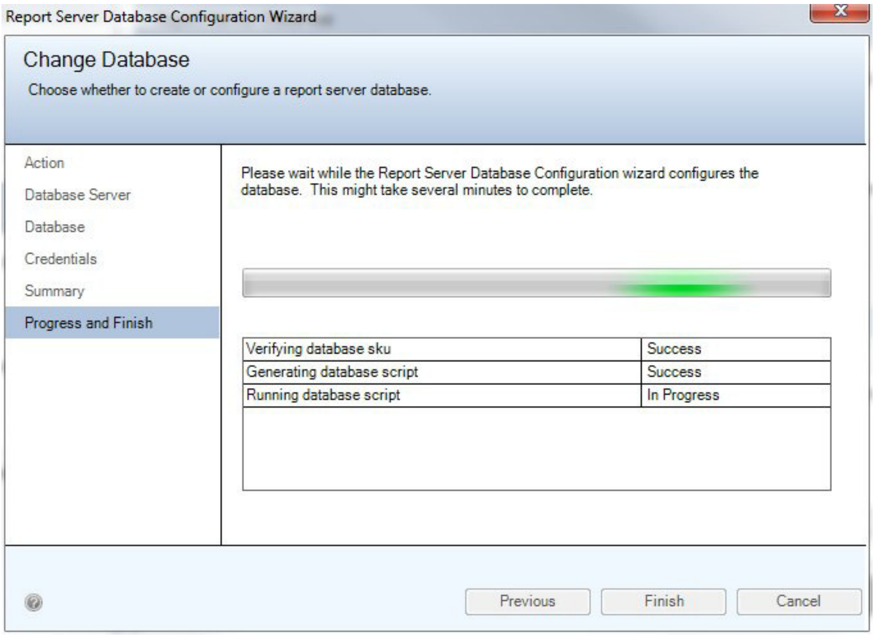
Change Database

Choose whether to create or configure a report server database.

The following information will be used to create a new report server database. Verify this information is correct before you continue.

SQL Server Instance:	KENT\RSTRAV11
Report Server Database:	ReportServer
Temp Database:	ReportServerTempDB
Report Server Language:	English (United States)
Report Server Mode:	Native
Authentication Type:	Service Account
Username:	LocalSystem
Password:	********

15. Click **Next** to go to the **Progress and Finish** screen which sets up the database.



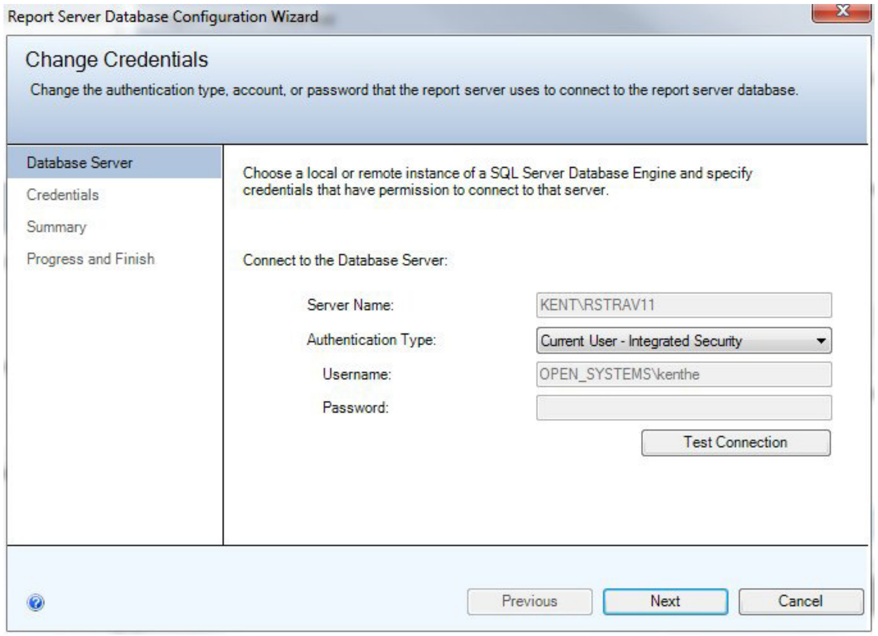
Report Server Database Configuration Wizard
Change Database
Choose whether to create or configure a report server database.
Action
Database Server
Database
Credentials
Summary
Progress and Finish
Previous
Finish
Cancel

Please wait while the Report Server Database Configuration wizard configures the database. This might take several minutes to complete.

Verifying database sku	Success
Generating database script	Success
Running database script	In Progress

16. Verify that there are no errors in the task status section. If you receive errors during any point of the configuration they will need to be addressed before you can move on to the report designing section. Once everything is configured correctly you can close Configure Report Server.

17. If you wish to change credentials for the current database, click **Change Credentials**.



Report Server Database Configuration Wizard

Change Credentials

Change the authentication type, account, or password that the report server uses to connect to the report server database.

Choose a local or remote instance of a SQL Server Database Engine and specify credentials that have permission to connect to that server.

Connect to the Database Server:

Server Name: KENT\RSTRAV11

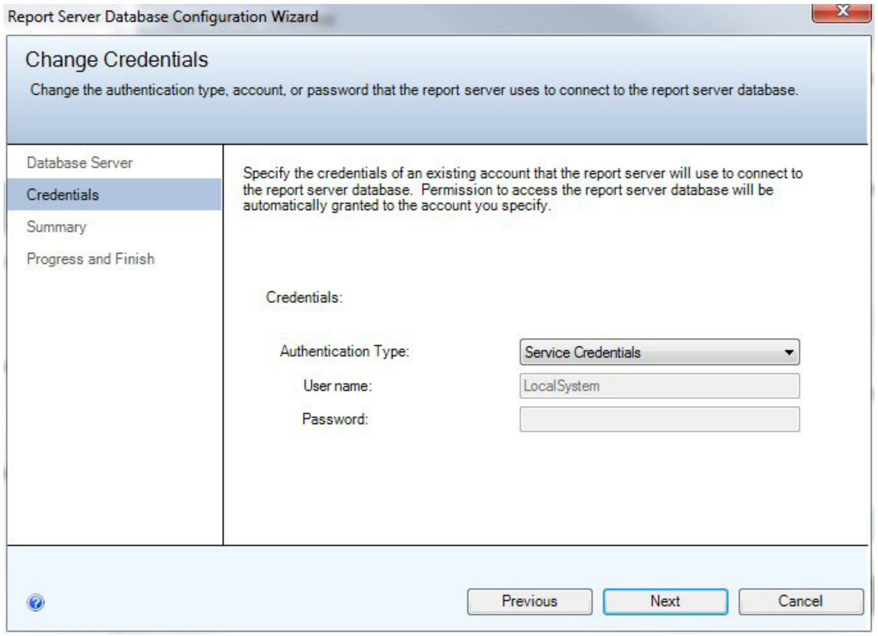
Authentication Type: Current User - Integrated Security

Username: OPEN_SYSTEMS\kenthe

Password:

18. The **SQL Server Name** and **instance** Reporting Services in installed on are displayed, **Authentication Type** (SQL Server Account works best), **Username** (sa works best), **Password** (sa password you entered when SQL was installed) and a **Database Name** (ReportServer is a

good database name). This can be an existing blank database or if you type in a name that does not exist, it will be created for you. Click **Next** to go to the Summary screen.



Report Server Database Configuration Wizard

Change Credentials

Change the authentication type, account, or password that the report server uses to connect to the report server database.

Specify the credentials of an existing account that the report server will use to connect to the report server database. Permission to access the report server database will be automatically granted to the account you specify.

Credentials:

Authentication Type: Service Credentials

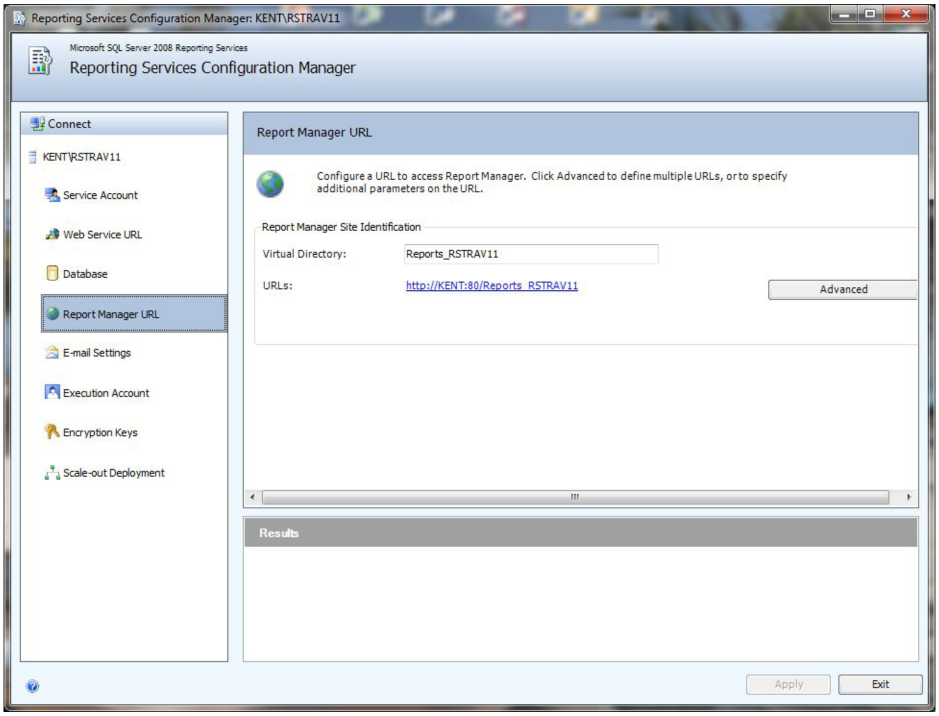
User name: LocalSystem

Password:

19. The **Summary** screen will be displayed to show the information you changed. Click **Next** to get to the **Progress and Finish** screen.

20. Once the **Progress and Finish** has completed click **Finish** to change the credentials.

21. Click on **Report Manager URL**.



Reporting Services Configuration Manager: KENT\RSTRAV11

Microsoft SQL Server 2008 Reporting Services
Reporting Services Configuration Manager

Report Manager URL

Configure a URL to access Report Manager. Click Advanced to define multiple URLs, or to specify additional parameters on the URL.

Report Manager Site Identification

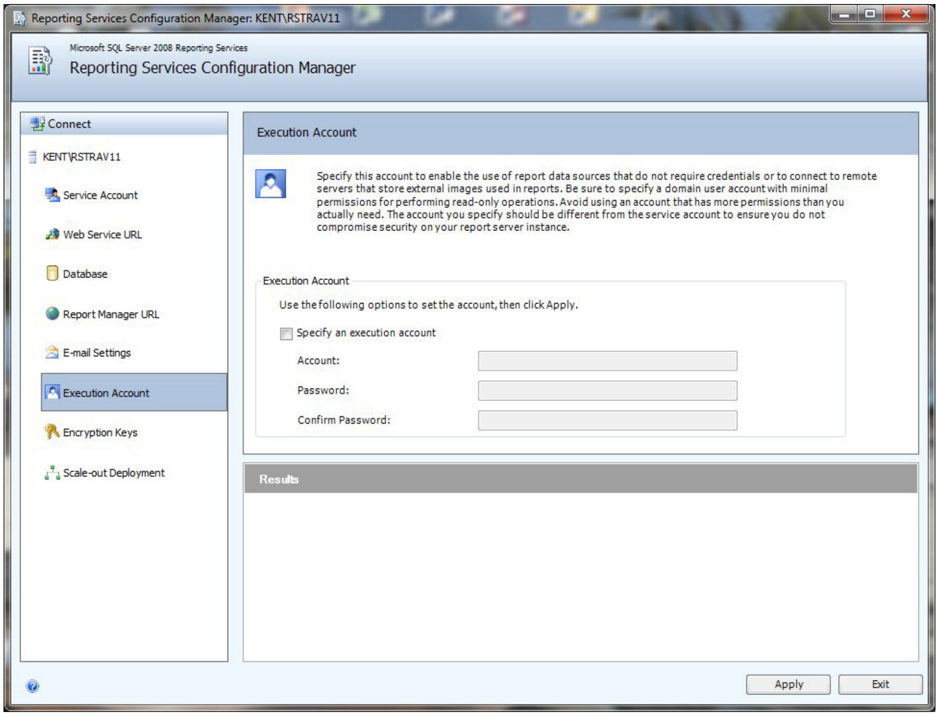
Virtual Directory: Reports_RSTRAV11

URLs: http://KENT:80/Reports_RSTRAV11

22. Set the **Virtual Directory** and click **Apply**.

The above screen shot shows the default settings, which should be fine in most TRAVERSE installations. **Note the name for future use.**

23. Click on **Execution Account**.



Reporting Services Configuration Manager: KENT\RSTRAV11

Microsoft SQL Server 2008 Reporting Services
Reporting Services Configuration Manager

Execution Account

Specify this account to enable the use of report data sources that do not require credentials or to connect to remote servers that store external images used in reports. Be sure to specify a domain user account with minimal permissions for performing read-only operations. Avoid using an account that has more permissions than you actually need. The account you specify should be different from the service account to ensure you do not compromise security on your report server instance.

Execution Account

Use the following options to set the account, then click Apply.

□ Specify an execution account

Account:

Password:

Confirm Password:

24. Set the permissions accordingly and click **Apply** to set the permissions.

    Local System is the default and should work fine for most TRAVERSE installations. Do not specify an execution account, accept the default.

3

Designing Reports

Overview

We are now ready to start designing reports using Reporting Services. Of course, few people build reports just for the fun of it. Usually there is some business reason for making the reports. For our report designing needs, we will use the Continental Products Unlimited (CPU) company that is supplied as the sample company with TRAVERSE, as the company needing reports.

Each of the sample reports used in this chapter is presented in a manner similar to what you will see in this section. The report is introduced with a list of the Report Services features highlighted. This is followed by the business need of our sample company, Continental Products Unlimited (CPU), which this report is meant to fill. Next is an overview of the tasks that must be accomplished to create the report.

Finally there are the steps to walk through for each task, step by step. In addition to the step-by-step descriptions, each task includes a few notes to provide additional information on the steps you just completed. Follow the step-by-step instructions to complete the task, and then read through the task notes to gain additional understanding of the process you have just completed. You can complete the step-by-step instructions using either the Business Intelligence Development Studio or Visual Studio.

THE CUSTOMER LIST REPORT

Features Highlighted

- Creating a data source
- Using Query Designer to create a dataset
- Using the Report Wizard to create a table report

Business Need

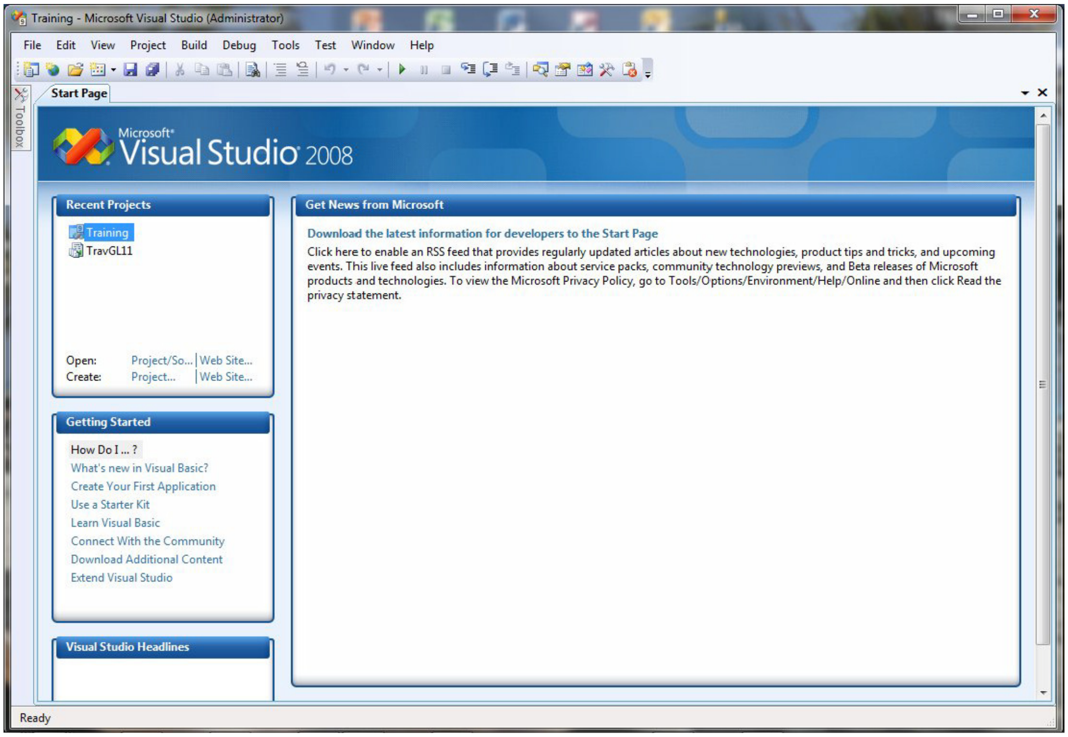
The accounting department at Continental Products Unlimited (CPU) would like an e-mail and address directory containing all contacts for its customers. The directory should be an alphabetical list of all CPU customers. It must include the customer name, along with the contact, e-mail address and mailing address for each customer.

Task Overview

1. Begin a New Project in the Business Intelligence Development Studio or Visual Studio.
2. Create a Data Source.
3. Create a Dataset.
4. Choose the Report Layout.
5. Edit the layout of the report to better match our needs

Customer List Report, Task 1: Begin a New Project in the Business Intelligence Development Studio or Visual Studio

1. To start the Business Intelligence Development Studio go to **Start** - **All Programs** - **Microsoft SQL Server 2008** - **SQL Server Business Intelligence Development Studio**. The Start Page appears as shown below.

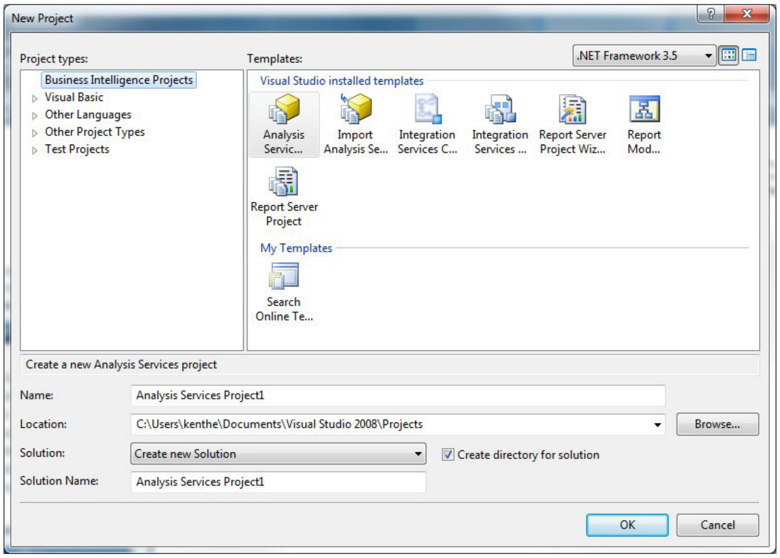

Training - Microsoft Visual Studio (Administrator)
File
Edit
View
Project
Build
Debug
Tools
Test
Window
Help
Toolbox
Start Page
Microsoft® Visual Studio 2008
Recent Projects
Training
TravGL11
Open: Project/So... | Web Site...
Create: Project... | Web Site...
Getting Started
How Do I ... ?
What's new in Visual Basic?
Create Your First Application
Use a Starter Kit
Learn Visual Basic
Connect With the Community
Download Additional Content
Extend Visual Studio
Visual Studio Headlines
Get News from Microsoft
Ready

Download the latest information for developers to the Start Page

Click here to enable an RSS feed that provides regularly updated articles about new technologies, product tips and tricks, and upcoming events. This live feed also includes information about service packs, community technology previews, and Beta releases of Microsoft products and technologies. To view the Microsoft Privacy Policy, go to Tools/Options/Environment/Help/Online and then click Read the privacy statement.

2. Click the **New Project** button . This displays the **New Project** dialog box, as shown below. (You can create a new project in three different ways: Select **File** - **New** - **Project** from

the Main menu, click the **New Project** toolbar button , or click the **Create**: **Project** link on the Start page.)



Create a new Analysis Services project

Name: Analysis Services Project1

Location: C:\Users\kenthe\Documents\Visual Studio 2008\Projects

Solution: Create new Solution

☑ Create directory for solution

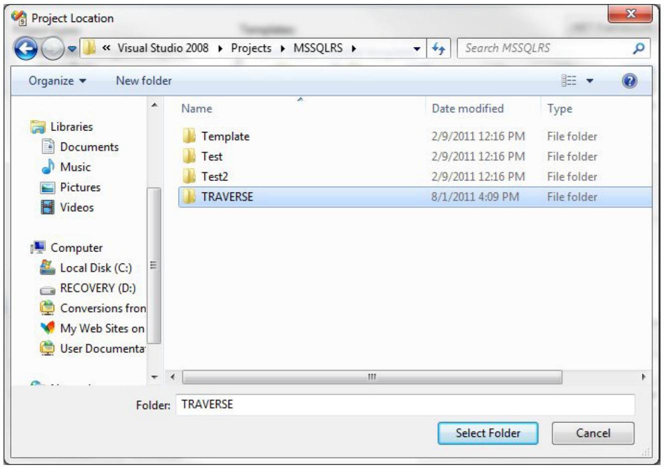
Solution Name: Analysis Services Project1

3. Select **Business Intelligence Projects** in the **Project Types** area of the dialog box.

4. Select **Report Server Project Wizard** in the **Templates** area of the dialog box.

5. Type **TRAVERSE** for the project name. This project will contain all the reports you create for this book.

6. Click **Browse** to open the **Project Location** dialog box.



Name	Date modified	Type
Template	2/9/2011 12:16 PM	File folder
Test	2/9/2011 12:16 PM	File folder
Test2	2/9/2011 12:16 PM	File folder
TRAVERSE	8/1/2011 4:09 PM	File folder

Folder: TRAVERSE

7. Browse to **My Documents\Visual Studio 2008\Projects**.

8. Click the **New Folder** button in the toolbar at the top of the Project Location dialog box.

9. Enter **MSSQLRS** for the name of the new folder. This folder will contain all the projects you create for this class.

10. Click **OK** in the New Folder dialog box.

11. Click **Open** in the lower-right corner of the Project Location dialog box. The New Project dialog box should now look like the picture above.

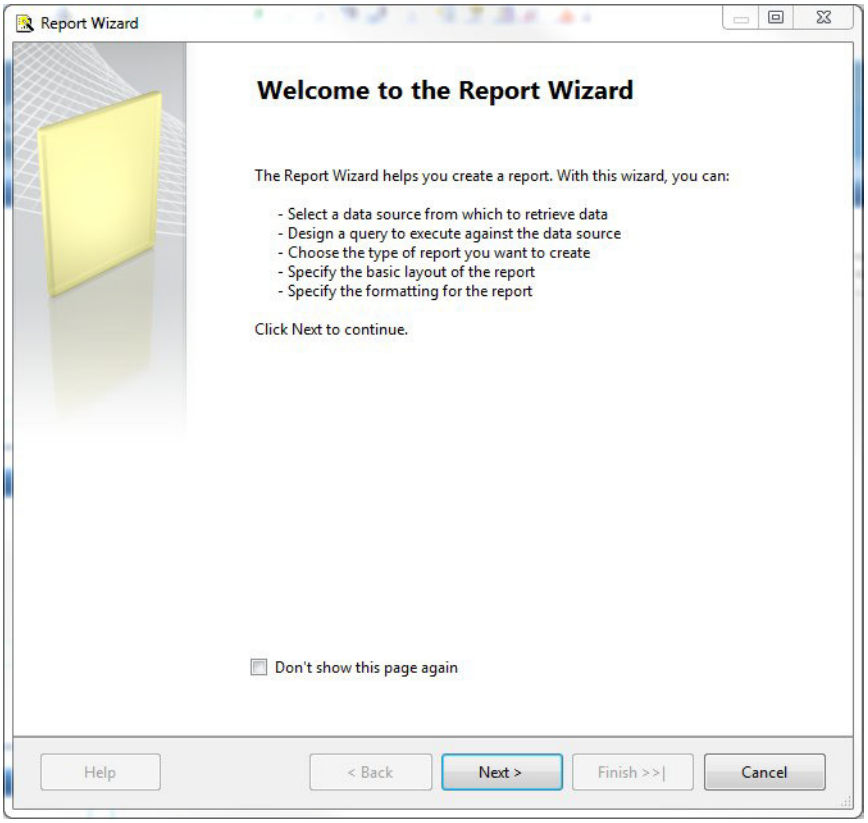
NOTE: We have now established a name and location for this project. This must be done for every project you create. Because the Business Intelligence Development Studio and Visual Studio 2005 use the project name to create a folder for all the project files, the project name should be a valid Windows folder name. You can use the Browse button to browse to the appropriate location, as we did here, or you can type the path in the Location text box.

Task Notes:

The project name is appended to the end of the location path to create the full path for the folder that will contain the new project. In our example, a folder called TRAVERSE will be created inside the folder MSSQLRS. All the files created as part of the TRAVERSE project will be placed in this folder.

Customer List, Task 2: Create a Data Source

1. Click **OK** in the New Project dialog box to start the Report Wizard. The **Welcome to the Report Wizard** page appears as shown here.



Welcome to the Report Wizard

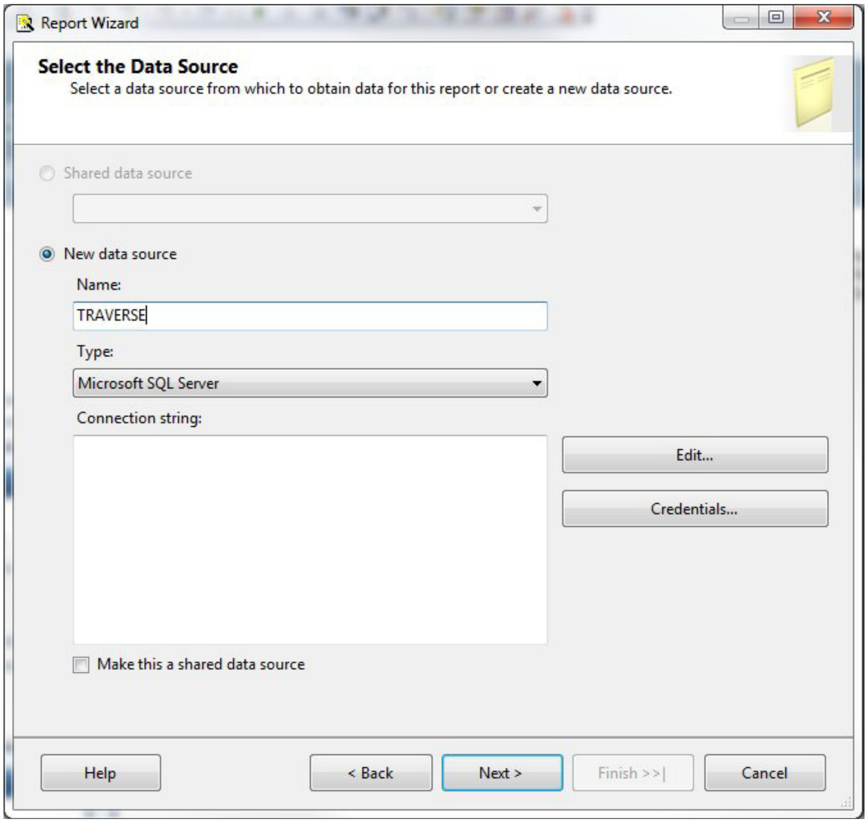
The Report Wizard helps you create a report. With this wizard, you can:

- Select a data source from which to retrieve data
- Design a query to execute against the data source
- Choose the type of report you want to create
- Specify the basic layout of the report
- Specify the formatting for the report

Click Next to continue.

□ Don't show this page again

2. Click **Next**. The **Select the Data Source** page appears.



Select the Data Source

Select a data source from which to obtain data for this report or create a new data source.

○ Shared data source

● New data source

Name: TRAVERSE

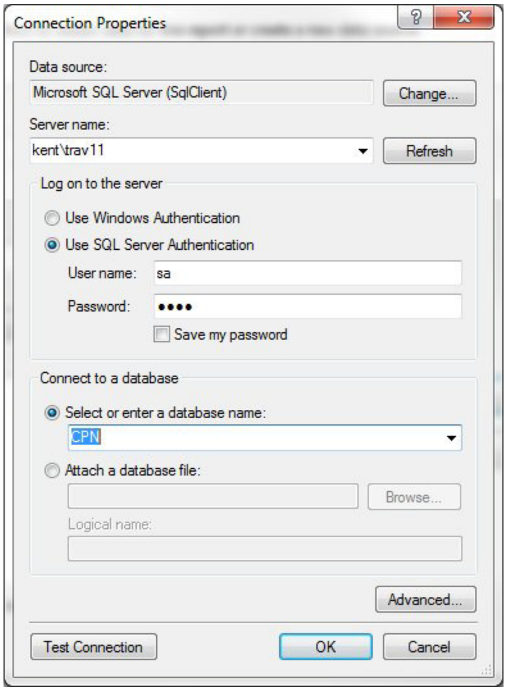
Type: Microsoft SQL Server

Connection string:

3. Type **TRAVERSE** for the New data source **Name**.

4. Select **Microsoft SQL Server** from the Type drop-down list, if it is not already selected.

5. Click **Edit**. The Connection Properties dialog box appears.



Connection Properties

Data source: Microsoft SQL Server (SqlClient)

Server name: kent\trav11

Log on to the server

○ Use Windows Authentication

● Use SQL Server Authentication

User name: sa

Password: ••••

□ Save my password

Connect to a database

● Select or enter a database name: CPN

○ Attach a database file:

Logical name:

6. Type the name and instance of the SQL Server that has your TRAVERSE databases.

   If you have a server with an instance name the syntax for that should be similar to this: servername\instancename. Notice the Server Name field in the picture above.

7. Select the **Use SQL Server Authentication** radio button.

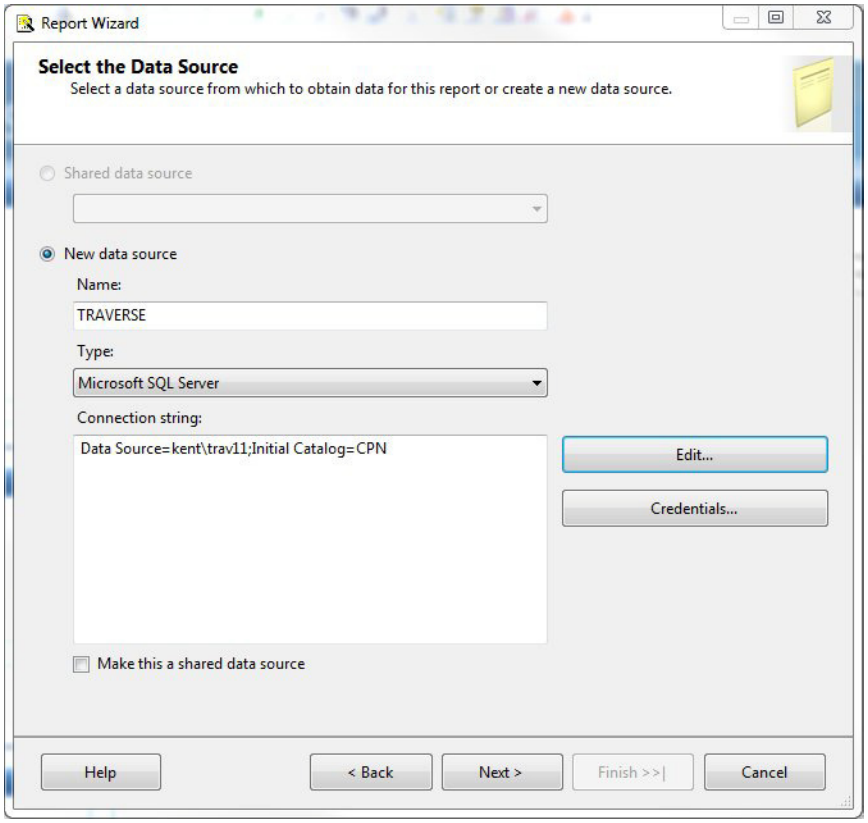
8. Type **sa** as the **Username** and your sa password as the **Password**.

9. Check the **Save my password** box.

10. Select the TRAVERSE company ID as the database to connect to. For this exercise I am connecting to CPN.

11. Click the **Test Connection** button. If the message “Test connection succeeded” appears, click **OK**. If an error appears make sure you typed the information correctly into all the fields.

12. Click **OK** to return to the Select the Data Source page. The Select Data Source page should now look similar to this.



Select the Data Source

Select a data source from which to obtain data for this report or create a new data source.

○ Shared data source

● New data source

Name: TRAVERSE

Type: Microsoft SQL Server

Connection string:

```
Data Source=kent\trav11;Initial Catalog=CPN
```

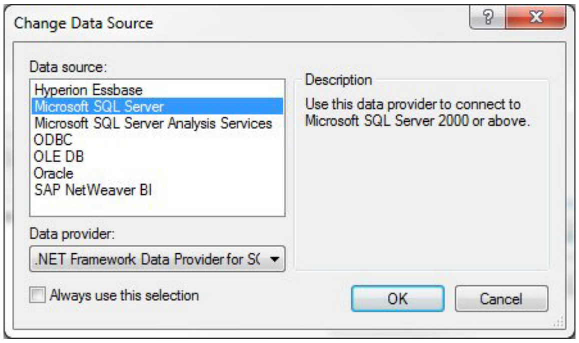
□ Make this a shared data source

NOTE: The data source is a set of instructions for connecting the database server or the data file that will provide the information for your report. This set of instructions is also known as a connection string. In this sample report we used the Connection Properties dialog box to build the connection string.

Task Notes:

Reporting Services can utilize data from a number of different databases and data files, but you need to tell the wizard what type of database or data file the report will be using. You did this using the Type drop-down list in Step 4 or the previous task. This selection tells Reporting Services which data provider to use when accessing the database or data file. When you select Microsoft SQL Server, Reporting Services uses the .NET Framework Data Provider for SQL Server. The data provider knows how to retrieve information from a SQL Server database.

The Type drop-down list on the Select the Data Source page includes only a few of the possible types of data sources. If you are using data from a data source other than a Microsoft SQL Server database, you need to click the Change button on the Connection Properties dialog box. This displays the Change Data Source dialog box.



Change Data Source

Data source:

- Hyperion Essbase
- Microsoft SQL Server
- Microsoft SQL Server Analysis Services
- ODBC
- OLE DB
- Oracle
- SAP NetWeaver BI

Description

Use this data provider to connect to Microsoft SQL Server 2000 or above.

Data provider: .NET Framework Data Provider for S(

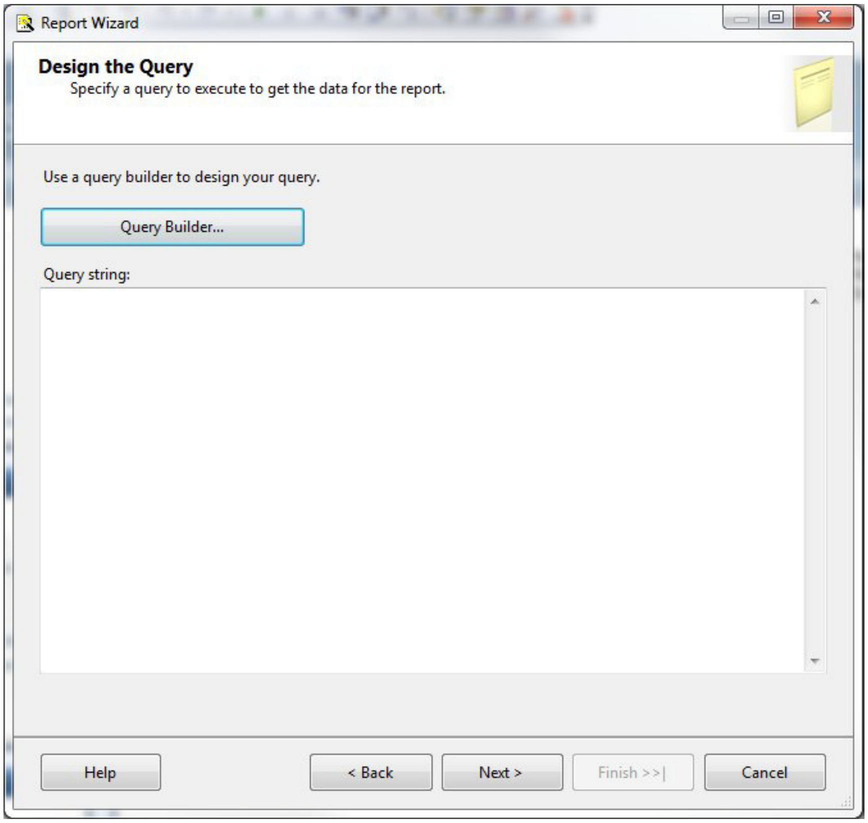
□ Always use this selection

Use this dialog box to select the appropriate data source type.

A data source can be used by a single report, or it can ba shared by several reports in the same project. Checking the Make This a Shared Data Source check box allows this data source to be used by many reports. Shared data sources are stored separately from the reports that use them. Nonshared data sources are stored right in the report definition. If you have a number of reports in the same project that utilize data from the same database or the same data files, you will save time by using a shared data source.

Customer List Report, Task 3: Create a Dataset

1. Click **Next**. The Design the Query page of the Report Wizard appears.



Report Wizard

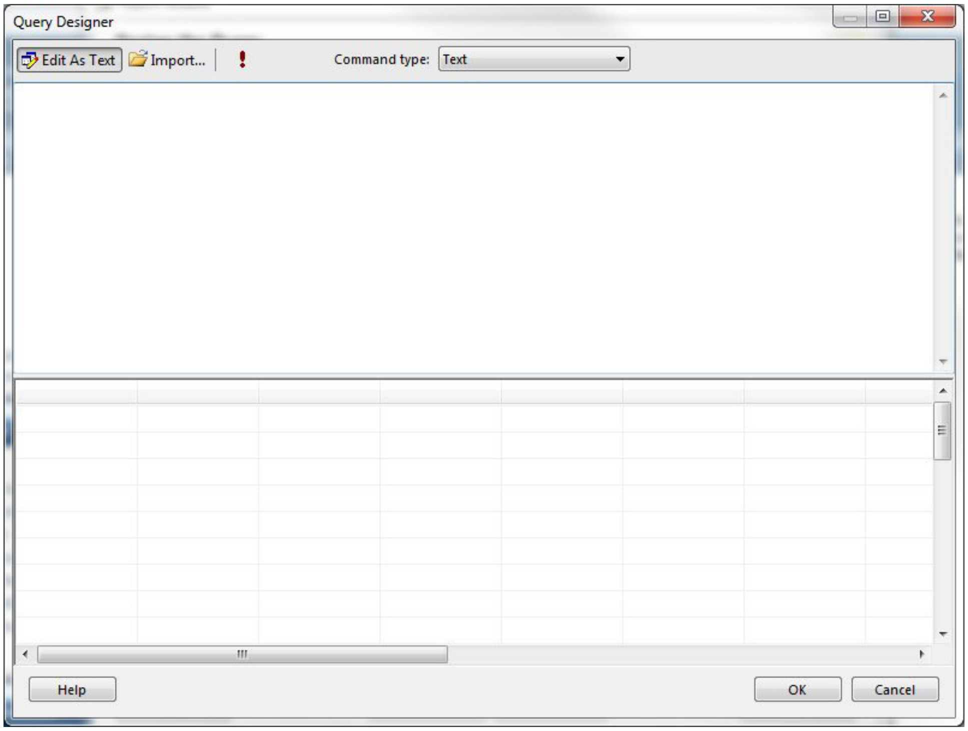
Design the Query

Specify a query to execute to get the data for the report.

Use a query builder to design your query.

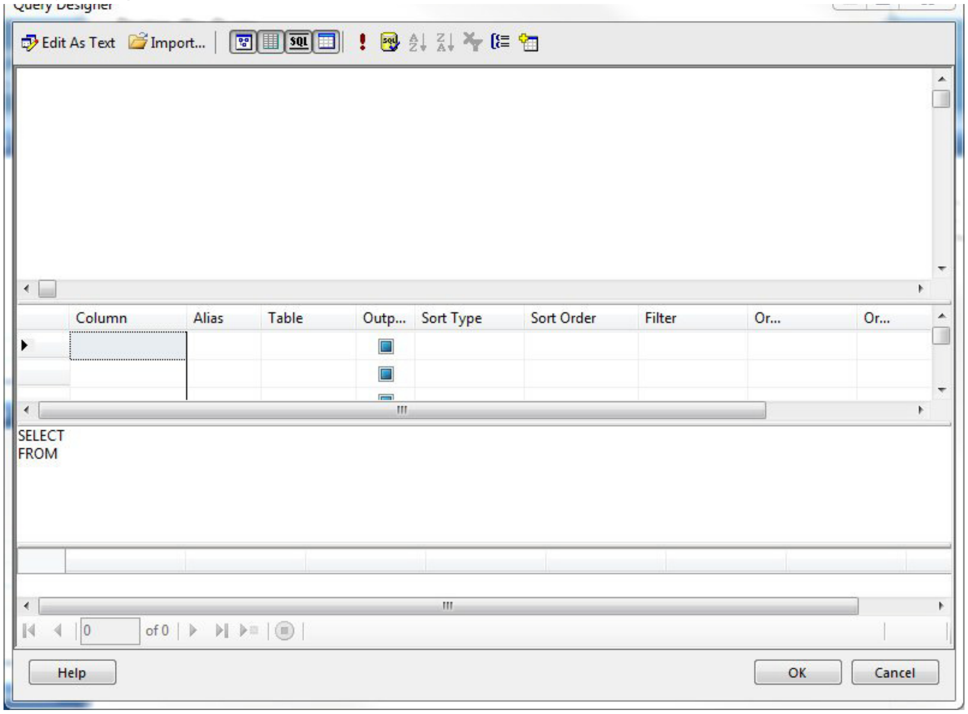
Query string:

2. Click **Query Builder**. The Query Design window appears with the Generic Query Designer active.



Command type: Text

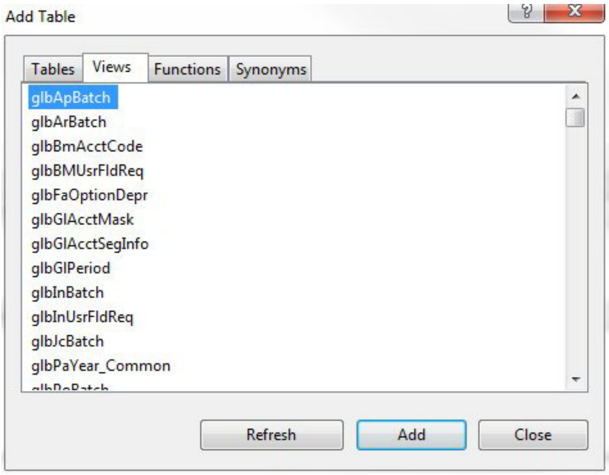
3. Click the **Edit as Text** toggle button . The Query Designer will switch to the Graphical Query Designer.



```
SELECT
FROM
```

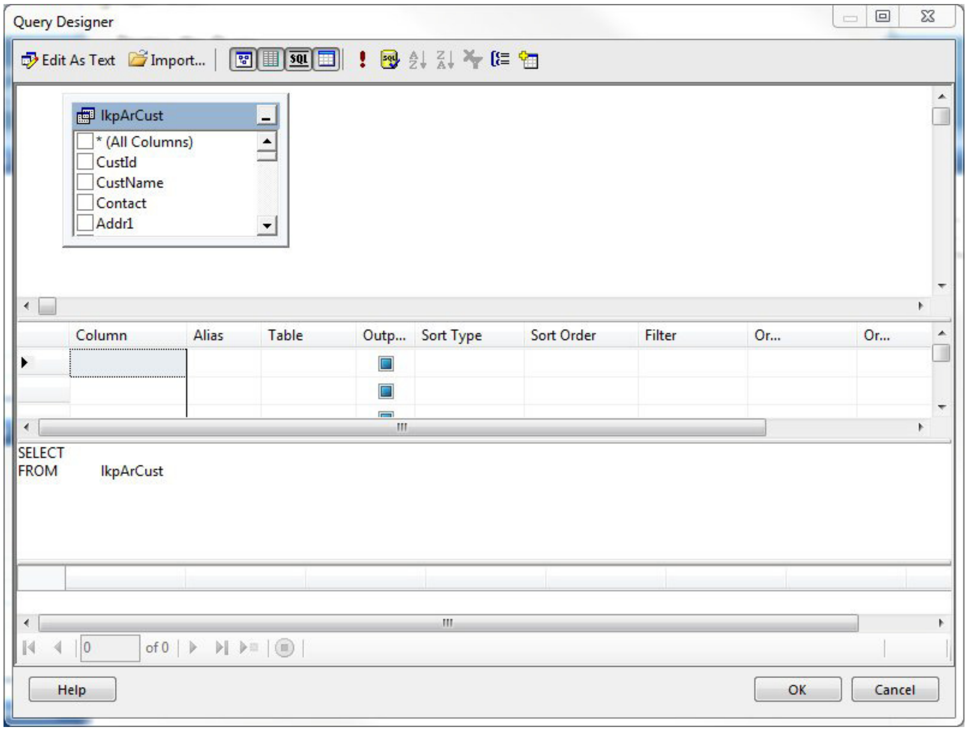
4. The Graphical Query Designer is divided into four horizontal sections. The top section is called the *diagram pane*. Right-click the diagram pane. You see the Context menu. Select **Add Table** or click on the **Add Table** button in the toolbar .

5. The **Add Table** dialog box appears. The dialog box contains a list of the tables, views, and functions that return datasets, which are found in the data source.



6. Click the Views tab and scroll down to find the view **lkpArCust** and double click on it, or highlight it and click **Add**.

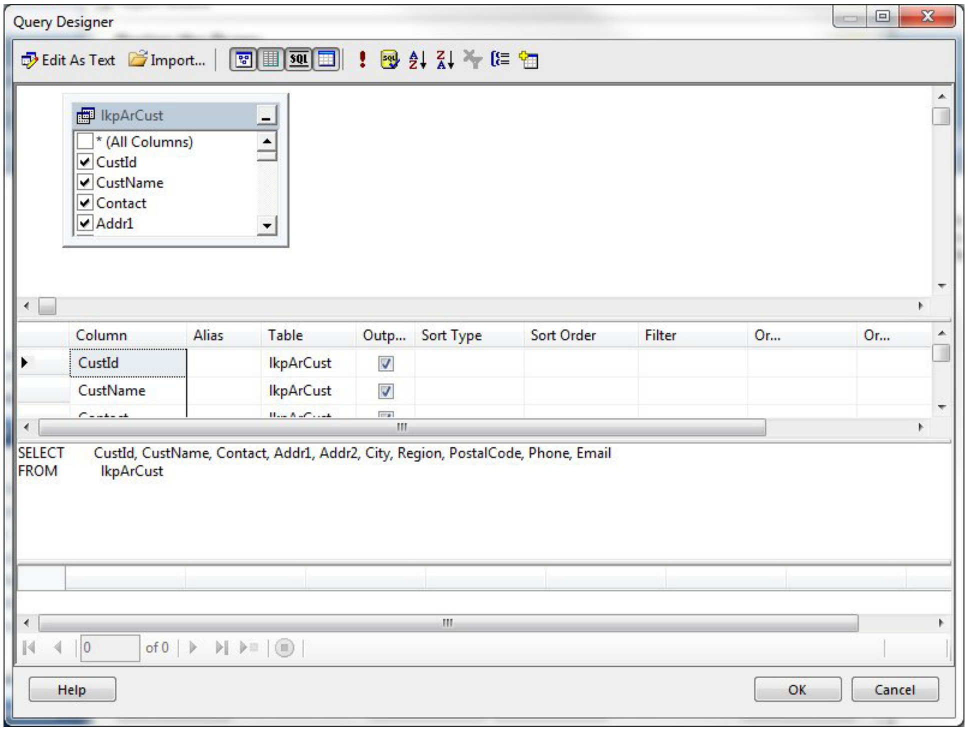
7. Click **Close** to exit the Add Table dialog box.



```
SELECT
FROM      lkpArCust
```

8. A list of fields in the Customer view (lkpArCust) is displayed. Click the check box next to the **CustID** field.

9. Scroll down and check the boxes in front of the following fields **CustName**, **Contact**, **Addr1**, **Addr2**, **City**, **Region**, **PostalCode**, **Phone**, **Email**.



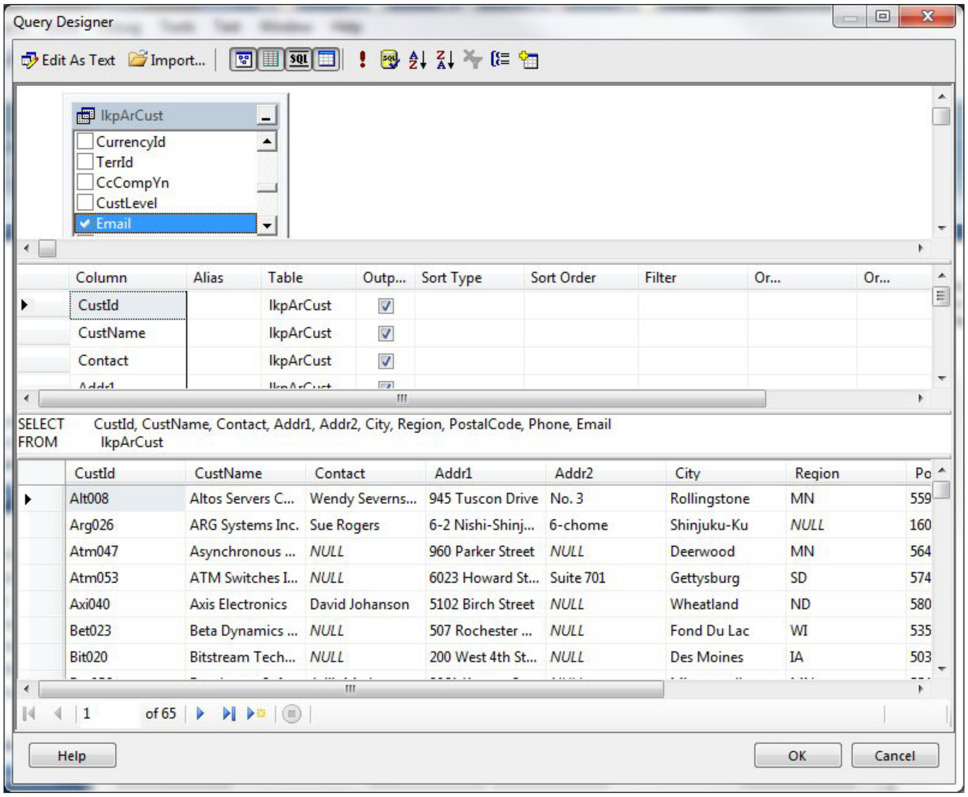
	Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...
▶	CustId		lkpArCust	☑					
	CustName		lkpArCust	☑					

```
SELECT    CustId, CustName, Contact, Addr1, Addr2, City, Region, PostalCode, Phone, Email
FROM      lkpArCust
```

10. The section of the Query Designer directly below the diagram pane is called the *criteria pane.* This area displays the fields we selected from the table fields displayed in the table list area. We can do filtering and sorting functions in this area of the Query Designer. For this report we do not need to sort or filter any of the fields we added. We will do that later in the report design.

11. The section of the Query Designer directly below the criteria pane is the *SQL pane.*

12. Right click in the SQL Pane and select **Execute SQL**, or click the Execute button ! in the toolbar. This executes the query and displays the results in the bottom section of the Query Designer. This section is called the *results pane*.



Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...
CustId		IkpArCust	☑					
CustName		IkpArCust	☑					
Contact		IkpArCust	☑					

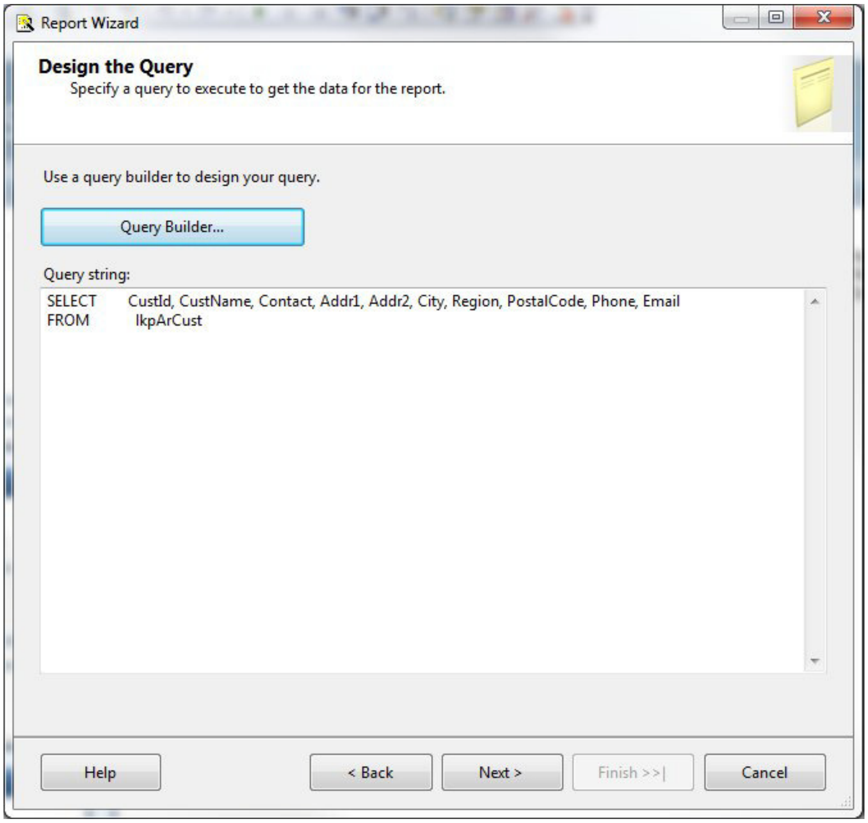
```
SELECT     CustId, CustName, Contact, Addr1, Addr2, City, Region, PostalCode, Phone, Email
FROM       IkpArCust
```

CustId	CustName	Contact	Addr1	Addr2	City	Region	Po
Alt008	Altos Servers C...	Wendy Severns...	945 Tuscon Drive	No. 3	Rollingstone	MN	559
Arg026	ARG Systems Inc.	Sue Rogers	6-2 Nishi-Shinj...	6-chome	Shinjuku-Ku	*NULL*	160
Atm047	Asynchronous ...	*NULL*	960 Parker Street	*NULL*	Deerwood	MN	564
Atm053	ATM Switches I...	*NULL*	6023 Howard St...	Suite 701	Gettysburg	SD	574
Axi040	Axis Electronics	David Johanson	5102 Birch Street	*NULL*	Wheatland	ND	580
Bet023	Beta Dynamics ...	*NULL*	507 Rochester ...	*NULL*	Fond Du Lac	WI	535
Bit020	Bitstream Tech...	*NULL*	200 West 4th St...	*NULL*	Des Moines	IA	503

1 of 65

13. Right click the results pane and select **Clear Results** from the right click menu.

14. Click **OK** to return to the Design the Query page of the Report Wizard. This page should now look like the following picture.


Report Wizard
Design the Query
Specify a query to execute to get the data for the report.
Use a query builder to design your query.
Query Builder...
Query string:
Help
< Back
Next >
Finish >>|
Cancel

```
SELECT     CustId, CustName, Contact, Addr1, Addr2, City, Region, PostalCode, Phone, Email
FROM        lkpArCust
```

NOTE: The dataset represents the information to be retrieved from the data source and used in your report. The dataset consists of two parts. The first part is the database command used to retrieve data. This is the SELECT statement you created using the Query Designer. This database command is called the *query string*. The second part is the list of columns in the result set created by executing the query string. This list of columns is called the *structure* or *schema* of the result set.

NOTE: If you are familiar with the data source and familiar with the SELECT statement, you can type your SELECT statement directly in the Query String text box on the Design the Query page.

NOTE: We selected the view for the customer information because permissions are set to the views in the TRAVERSE database, when menu permissions are set in Server

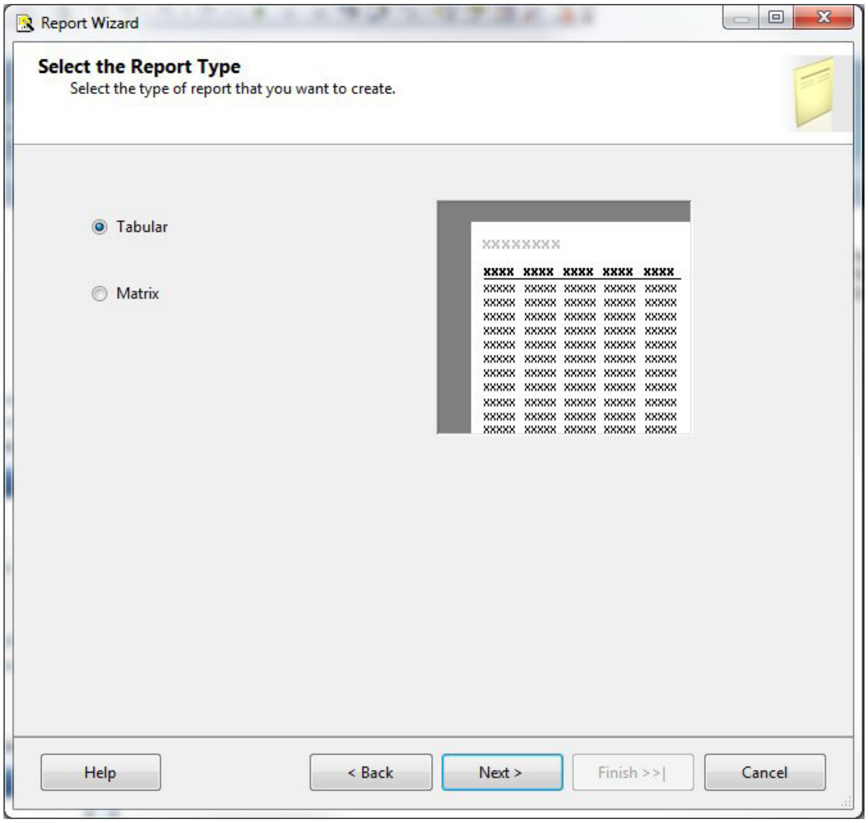
Manager. Permissions are not set to the tables so users cannot open the tables and edit the data. Views allow us to see the data, but not edit the data.

Task Notes:

It is a good idea to run the query yourself before exiting the Query Designer. We did this in Step 12 of this task. This ensures no errors exist in the SQL statement the Query Designer created for you. It also lets you look at the results set in the results pane, so you can make sure you are getting the information you expected.

Customer List Report, Task 4: Choose the Report Layout

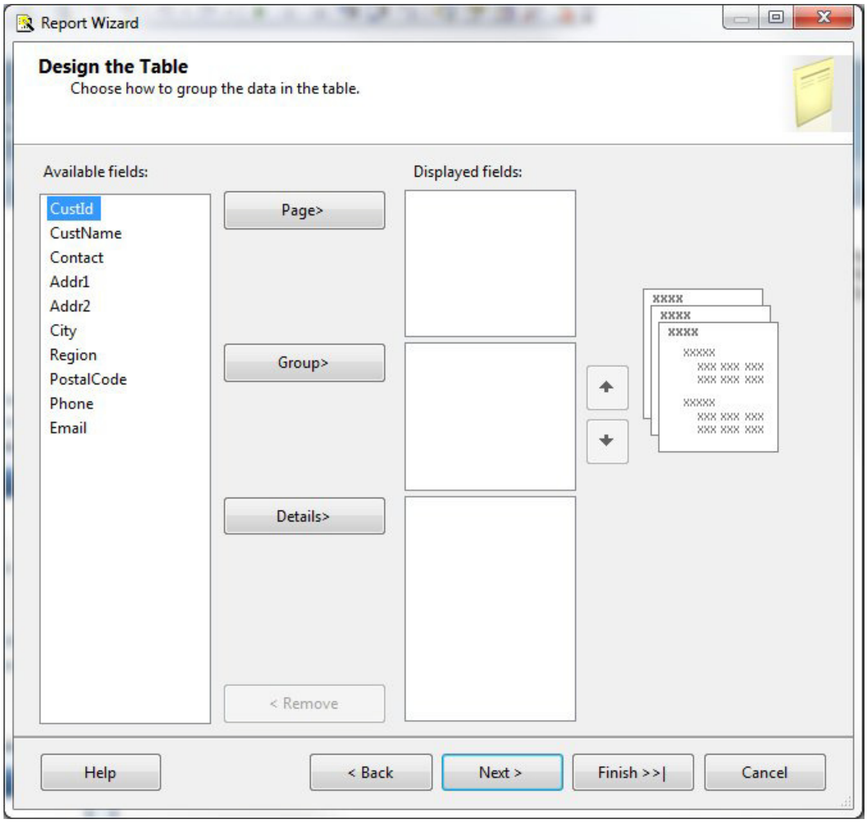
1. Click **Next**. The **Select the Report Type** page of the Report Wizard appears.


Report Wizard

Select the Report Type
Select the type of report that you want to create.

● Tabular
○ Matrix

2. Make sure the **Tabular** radio button is selected and click **Next**. The **Design the Table** page of the Report Wizard appears.



Report Wizard

Design the Table

Choose how to group the data in the table.

Available fields:

CustId
CustName
Contact
Addr1
Addr2
City
Region
PostalCode
Phone
Email

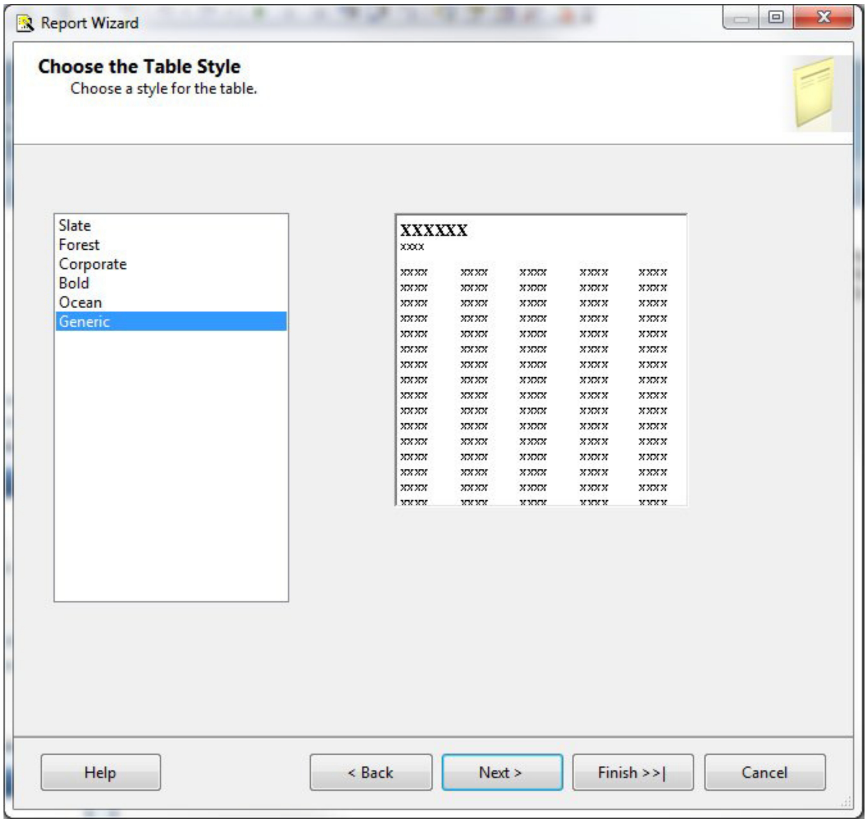
Displayed fields:

3. With the **Cust ID** highlighted, click the **Group** button to move the Cust ID into the Group **Displayed fields** box.

4. With the **CustName** field highlighted, click the **Details** button to move the CustName into the Detail **Displayed fields** box.

5. Do the same thing with the **Addr1**, **City**, **Region**, **PostalCode**, **Phone** and **Email** fields.

6. Click **Next**. The **Choose the Table Style** page of the Report Wizard appears.



Report Wizard

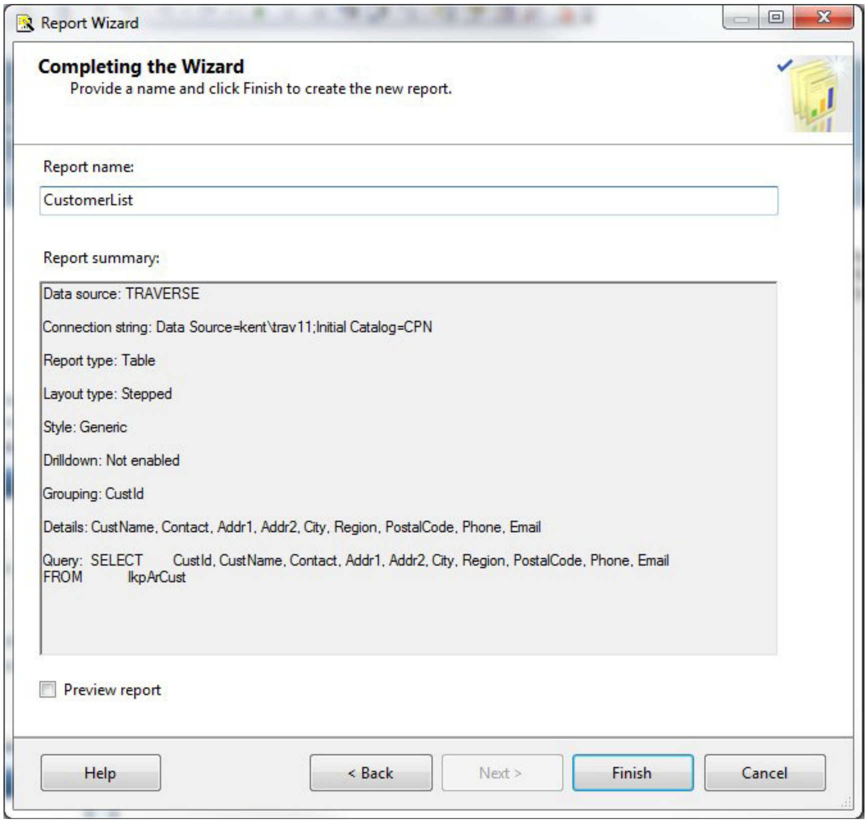
Choose the Table Style

Choose a style for the table.

Slate
Forest
Corporate
Bold
Ocean
Generic

7. Select the **Generic** style in the style list and click **Next**.

8. The **Completing the Wizard** page appears.

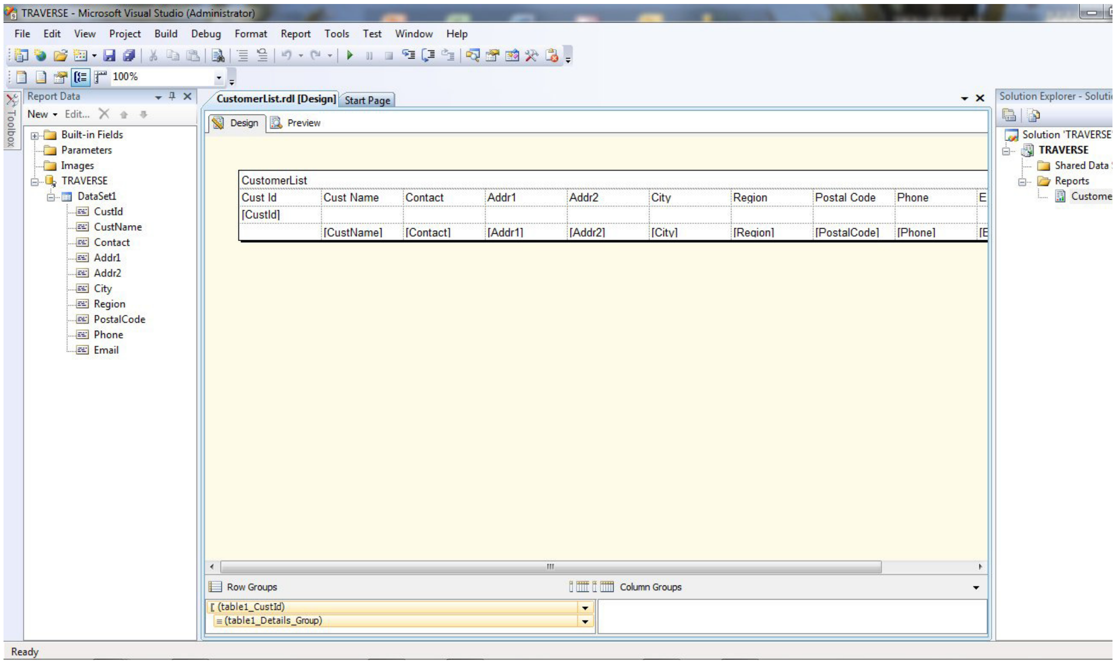


Report name: CustomerList

```
Data source: TRAVERSE
Connection string: Data Source=kent\trav11;Initial Catalog=CPN
Report type: Table
Layout type: Stepped
Style: Generic
Drilldown: Not enabled
Grouping: CustId
Details: CustName, Contact, Addr1, Addr2, City, Region, PostalCode, Phone, Email
Query:  SELECT      CustId, CustName, Contact, Addr1, Addr2, City, Region, PostalCode, Phone, Email
FROM         lkpArCust
```

9. Type **CustomerList** for the report name.

10. Click **Finish**. The Business Intelligence Development Studio or Visual Studio window appears with the **Report Designer** active.



CustomerList

Cust Id	Cust Name	Contact	Addr1	Addr2	City	Region	Postal Code	Phone	E
[CustId]									
	[CustName]	[Contact]	[Addr1]	[Addr2]	[City]	[Region]	[PostalCode]	[Phone]	[E

11. If we click on the **Preview** tab the report is generated and displays all the field headings on one line and all the detail data on another line.

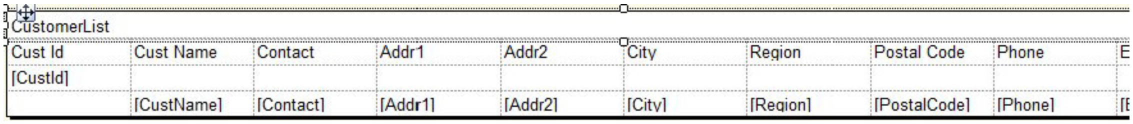
Task Notes:

We would like our Customer List to be formatted a little differently. Mainly so the fields can be fully displayed and on multiple lines.

We also want our report heading to be centered, bolded and larger.

Customer List Report, Task 5: Final Formatting

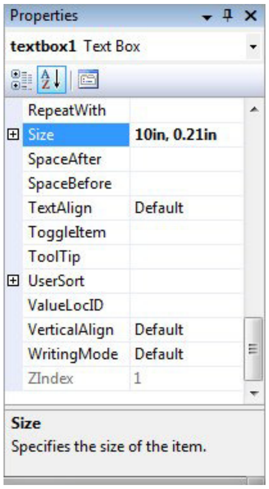
1. The first thing we want to do is format the report heading. Click in the top field of the table to select the **CustomerList** text box.C



CustomerList

Cust Id	Cust Name	Contact	Addr1	Addr2	City	Region	Postal Code	Phone	E
[CustId]									
	[CustName]	[Contact]	[Addr1]	[Addr2]	[City]	[Region]	[PostalCode]	[Phone]	[E

2. In the lower right corner of the screen is the **Properties** area.



Properties

textbox1 Text Box

RepeatWith	
Size	**10in, 0.21in**
SpaceAfter	
SpaceBefore	
TextAlign	Default
ToggleItem	
ToolTip	
UserSort	
ValueLocID	
VerticalAlign	Default
WritingMode	Default
ZIndex	1

Size
Specifies the size of the item.

3. Change the following properties to match the information in the table below.

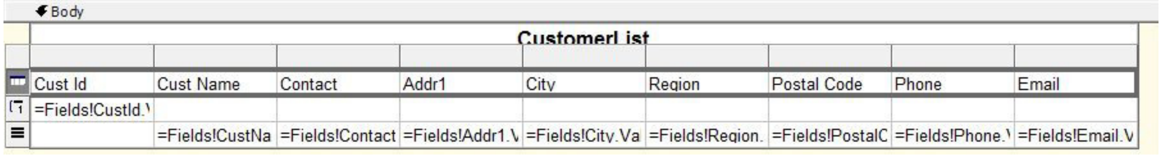
Table 1: Heading Text Box Properties

NameProperty	NameNew Value
Font Size	14
Font Weight	Bold
Text Align	Center
Value	Customer List
Size Width	7.5 In
Size Height	.33 In

4. The next thing we want to do is to add another detail row for our Addr1, City, Region and PostalCode fields to be printed on a second line.

5. Make sure the report is displayed in the **Layout** tab. If you are still in the Preview, click the Layout tab to view the design of the report.

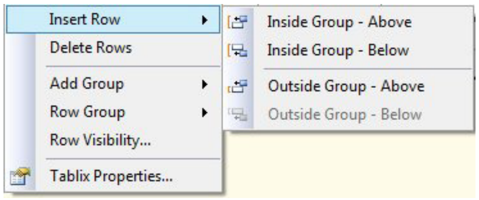
6. Click in the cell in the center of the table that has the data for the **CustName**. This will select the table and display grey boxes on the left side and top of the table.



CustomerList								
Cust Id	Cust Name	Contact	Addr1	City	Region	Postal Code	Phone	Email
=Fields!CustId.\								
	=Fields!CustNa	=Fields!Contact	=Fields!Addr1.\	=Fields!City.Va	=Fields!Region.	=Fields!PostalC	=Fields!Phone.\	=Fields!Email.V

7. Click the grey box on the left side of the table in the row that contains the **=Fields!CustName.Value** text. This is a detail row.

8. Right click on the grey box and select **Insert Row, Inside Group Below**.



9. We now see a new, empty detail row below our current row.

10. Now right click on the grey box next to the row with the field headings in it and select **Insert Row Below**.

11. We now have a new empty header row.

12. Click in the detail row that contains **Addr1** and press **Ctrl X** on your keyboard. This will cut the contents of the field.

13. Click in the empty field directly **below** the **CustName** data field and press **Crtl V**. This will paste the cut data into this field.

14. Do the same thing with the **City, Region** and **PostalCode** data fields. Moving them sequentially in the second detail row.

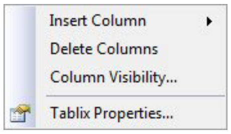
15. Follow the steps to cut and paste the **Phone** and **Email** data fields to move them after the Contact data field. They should be replacing the now empty fields where the Addr1 and City fields were.

16. Click the **Addr1** heading field and press **Ctrl X** to cut this field and paste it (**Ctrl V**) into the empty heading field under the CustName heading.

17. Follow these same steps to move the other headings to match the data fields below them.

18. You will have two empty columns at the end of the table. Click the grey box above the first empty column. Hold you shift key and click the next empty column.

19. Right click on one of the selected column grey boxes at the top to see the right click menu.



20. Select **Delete Columns** to delete the selected columns.

21. We do not need the Table Footer rows for this table. Right click on one of the grey boxes next to a row and select **Table Footer** from the menu. This will take the footer row off the table.

22. Next we want to format our table headings to make them stand out as headings for the data.

23. Click in the heading field for **CustID**. Hold your shift key and click in the heading field for **Postal Code**. This will highlight all the table heading fields.

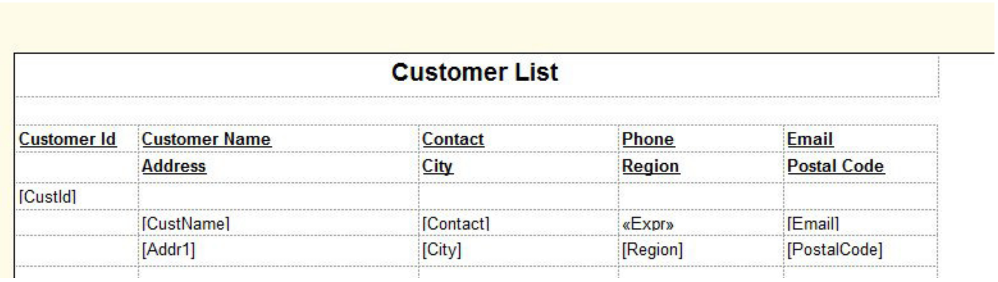
24. In the properties area in the bottom right corner of the screen change the properties to match those in the table below.

Table 2: Table Heading Properties

NameProperty	NameNew Value
Font Size	10 pt
Font Weight	Bold
Text Decoration	Underline

25. We will probably need to resize the fields so we can see all the data in the columns. To resize a column click on any cell in the table so you see the grey boxes at the top and left side. If you place your mouse on the line between the column headings your mouse pointer should change to a bar with arrows on each side. Change the size of the **Cust Name** column to make it larger.

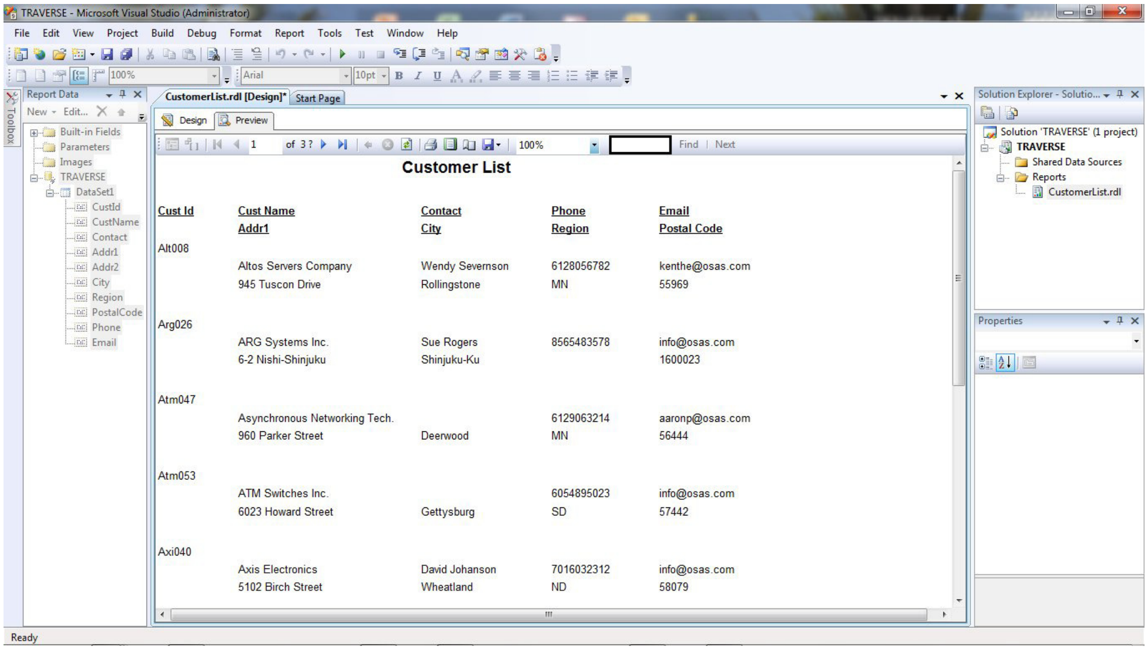
26. The table should look similar to this now.



Customer List

Customer Id	Customer Name	Contact	Phone	Email
	Address	City	Region	Postal Code
[CustId]				
	[CustName]	[Contact]	«Expr»	[Email]
	[Addr1]	[City]	[Region]	[PostalCode]

27. Click on the **Preview** tab to see what your report looks like.

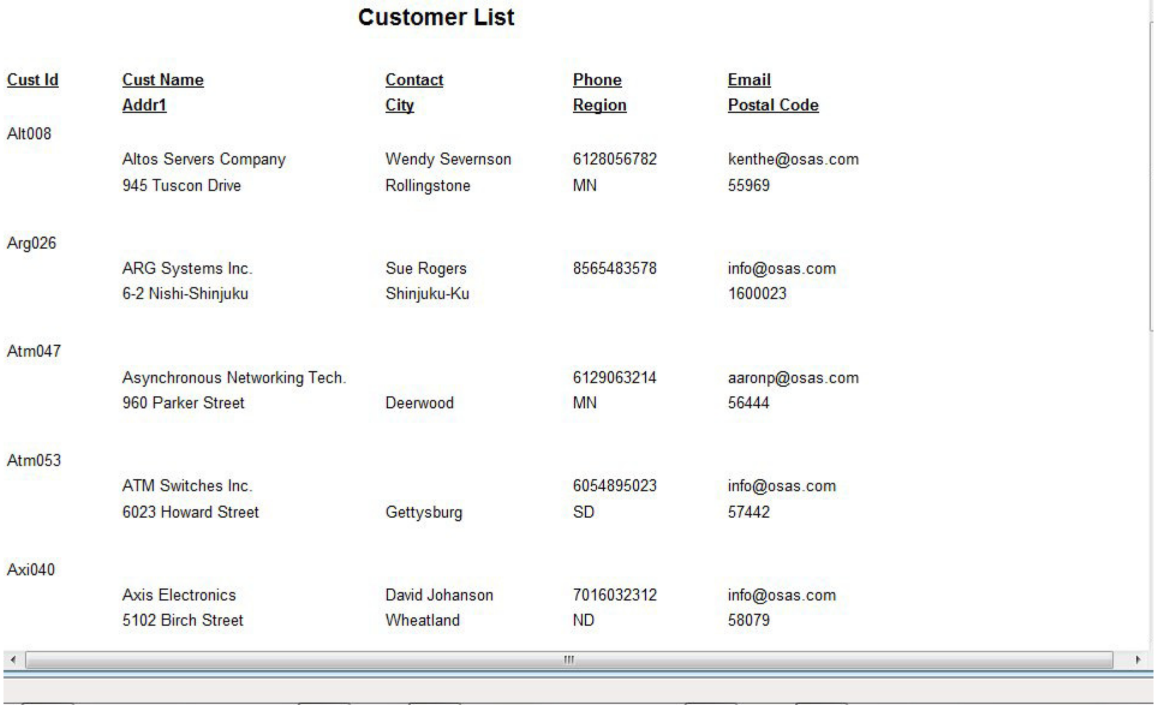


Customer List

Cust Id	Cust Name Addr1	Contact City	Phone Region	Email Postal Code
Alt008	Altos Servers Company 945 Tuscon Drive	Wendy Severnson Rollingstone	6128056782 MN	kenthe@osas.com 55969
Arg026	ARG Systems Inc. 6-2 Nishi-Shinjuku	Sue Rogers Shinjuku-Ku	8565483578	info@osas.com 1600023
Atm047	Asynchronous Networking Tech. 960 Parker Street	Deerwood	6129063214 MN	aaronp@osas.com 56444
Atm053	ATM Switches Inc. 6023 Howard Street	Gettysburg	6054895023 SD	info@osas.com 57442
Axi040	Axis Electronics 5102 Birch Street	David Johanson Wheatland	7016032312 ND	info@osas.com 58079

28. You may not have a space between the customer records. If you do not have a space between customer records, return to the **Layout** tab.

29. Click in the table to select it and right click on the last detail row and select **Add Row Below**. This will add a blank detail row below the last row with data in it. You should now have a blank row between the customer records when you preview it.



Customer List

Cust Id	Cust Name	Contact	Phone	Email
	Addr1	City	Region	Postal Code
Alt008				
	Altos Servers Company	Wendy Severnson	6128056782	kenthe@osas.com
	945 Tuscon Drive	Rollingstone	MN	55969
Arg026				
	ARG Systems Inc.	Sue Rogers	8565483578	info@osas.com
	6-2 Nishi-Shinjuku	Shinjuku-Ku		1600023
Atm047				
	Asynchronous Networking Tech.		6129063214	aaronp@osas.com
	960 Parker Street	Deerwood	MN	56444
Atm053				
	ATM Switches Inc.		6054895023	info@osas.com
	6023 Howard Street	Gettysburg	SD	57442
Axi040				
	Axis Electronics	David Johanson	7016032312	info@osas.com
	5102 Birch Street	Wheatland	ND	58079

30. Click the Save All button in the toolbar.

If you want your phone number to be masked to match the US phone format, right click on the phone data cell and select **Expression**. In the top box in the **Expression** dialog box type the following:

```
= string.Format(“({0}) {1}-{2}”, Fields!Phone.Value.Substring(0,
3), Fields!Phone.Value.Substring(3, 3), Fields!Phone.Value.Sub-
string(6))
```

You have now completed designing your first report!

Task Notes:

This report looks OK, but it would be nice if the reports had a standard layout. We will design a template report to use for the remaining reports so they will have a common look and feel.

Steps for Making Reporting Services Report

1. Install and set up Reporting Services and Report Manager
2. Start SQL Server Business Intelligence Development Studio (Visual Studio 2005)
3. Create a Project
4. Create a new Report
5. Establish data connection
6. Select information to include on report
7. Determine grouping and sorting
8. Choose report style
9. Test and preview
10. Modify design if needed
11. Deploy

DESIGNING A REPORT TEMPLATE

Features Highlighted

- Creating a reusable template for reports
- Using values from the Globals collection

Business Need

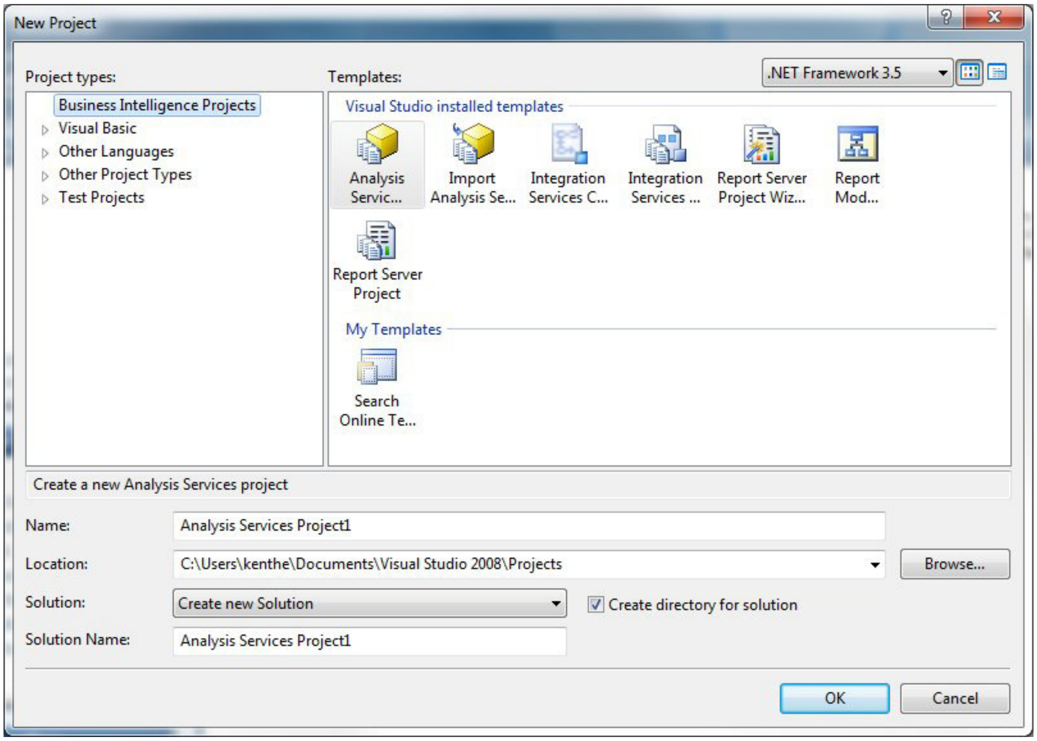
Continental Products Unlimited would like a template that can be used for each new report created. The report will include the company name and report name in a header across the top of each page. The template is also going to include a Page x of x area, the date the report was run and the time the report was run. It will also include a footer that will let the users know which page is the last of the report.

Task Overview

1. Create the Template Project and the Template Report with a Page Header.
2. Create the Page Footer on the Template Report.
3. Copy the Template to the Report Project Directory.

Report Template, Task 1: Create the Template Project and the Template Report with a Page Header

1. Create a new Reporting Services project called **Template** in the MSSQLRS folder. Click **OK** to create the new project.



New Project

Project types: Business Intelligence Projects

Templates: Visual Studio installed templates — Analysis Servic...

.NET Framework 3.5

Create a new Analysis Services project

Name: Analysis Services Project1

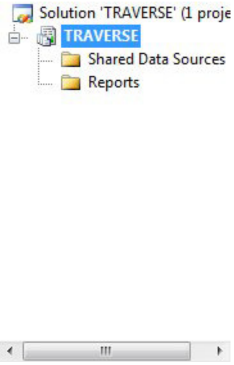
Location: C:\Users\kenthe\Documents\Visual Studio 2008\Projects

Solution: Create new Solution

☑ Create directory for solution

Solution Name: Analysis Services Project1

2. Right click on the **Reports** under the Template project in the Solution area and select **Add** then **New Item**.

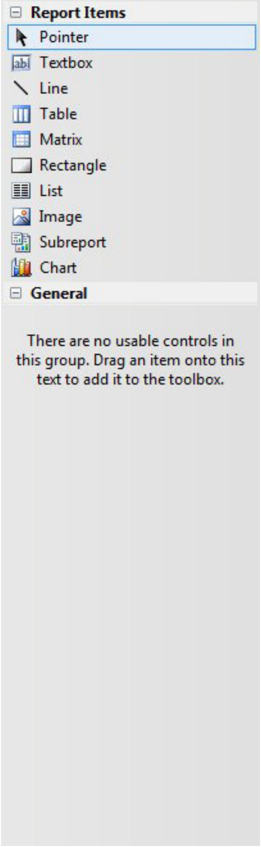


3. The Add New Item window appears. Select **Report** from the available choices in the top window. Enter **TRAVERSETemplate** as the report name and click **OK**. This will make a blank report and we now can set up our template for reports.

4. From the menu bar select **Report - Report Properties** and select **Landscape** for the Orientation.

5. From the Main menu, select **Report** - **Add Page Header**. A space for the page header layout appears above the layout area for the body of the report. (If Report is not showing on the Main menu, click anywhere on the report layout. the Format and Report menu choices appear.) Drag the grey bar separating the page header and the body down, so the page header area is larger.

6. From the **toolbox** tab, on the far left side of the screen, place a textbox in the layout area in the page header.



There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.

7. Edit the properties of this textbox using the properties area in the lower right corner of the screen to edit the following properties.

Table 1: Template Heading 1 Textbox

NameProperty	NameNew Value
Font size	12 pt
Font weight	Bold
Text Align	Center
Value	Continental Products Unlimited
Location Left	0 in
Location Top	0 in
Size Width	7.5 in
Size Height	.25 in

8. Place another textbox directly under the first textbox and set the properties as follows.

Table 2: Template Heading 2 Textbox

NameProperty	NameNew Value
Font Size	12pt
Font Weight	Bold
Text Align	Center
Value	Report Name
Location Left	0 in
Location Top	.25 in
Size Width	7.5 in
Size Height	.25 in

9. Add a third textbox directly under the last one added and set the properties as follows.

Table 3: Template Heading 3 Textbox

NameProperty	NameNew Value
Location Left	0 in
Location Top	.5 in
Size Width	7.5 in
Size Height	.25 in

10. Add 5 more textboxes in the upper right corner of the report and edit the following properties for these textboxes.

Table 4: Template Heading Page #

NameProperty	NameNew Value
Font Size	8 pt
Text Align	Right
Name	PageXofN
Location Left	7.5 in
Location Top	0 in
Size Width	1.35 in
Size Height	.25 in

Table 5: Template Heading Report Date Label

NameProperty	NameNew Value
Font Size	8 pt
Text Align	Left
Value	Report Date
Name	ReportDateTimeLabel
Location Left	7.5 in
Location Top	.25 in
Size Width	.7 in
Size Height	.25 in

Table 6: Template Heading Report Date

NameProperty	NameNew Value
Font Size	8 pt
Text Align	Right
Value	= Today
Location Left	8.2 in
Location Top	.25 in
Size Width	.65 in
Size Height	.25 in
Format	d

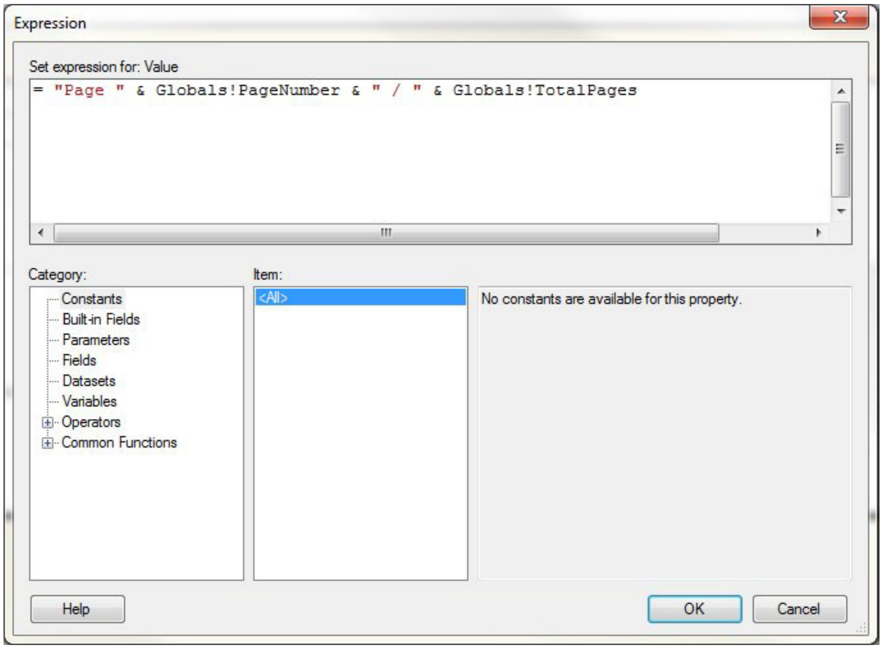
Table 7: Template Heading Report Time Label

NameProperty	NameNew Value
Font Size	8 pt
Text Align	Left
Value	Report Time
Location Left	8.2 in
Location Top	.5 in
Size Width	.7 in
Size Height	.25 in

Table 8: Heading Report Report Time

NameProperty	NameNew Value
Font Size	8 pt
Text Align	Right
Value	= Now
Location Left	8.2 in
Location Top	.5 in
Size Width	.65 in
Size Height	.25 in
Format	t

11. Right click the top right text box to highlight it and select **Expression** from the menu. The Expression dialog box appears.



Set expression for: Value

```
= "Page " & Globals!PageNumber & " / " & Globals!TotalPages
```

No constants are available for this property.

12. Enter the following into the box at the top of the dialog box:

```
= “Page “ & Globals!PageNumber & “/ “& Globals!TotalPages
```

If you don’t want to type the whole expression you can use the selections at the bottom for parts of the expression.

After the **= type “Page “**then at the bottom select **Global** in the left window and double click on **PageNumber**.

Type **&** with a space in front and behind and then **“/” &** with a space in front and behind again.

Select **Global** again and then double click on **TotalPages**.

Once you have done all this your expression should be the same as above.

13. Click **OK** once you have finished.

Report Template, Task 2: Create the Page Footer on the Template Report

1. Click in the report layout area.

2. From the Main menu, select **Report** - **Add Page Footer**. A space for the page footer layout appears below the layout area for the body of the report.

3. Page Footer is selected in the Properties window. Modify the following property for the page footer.

Table 9: Template Footer Properties

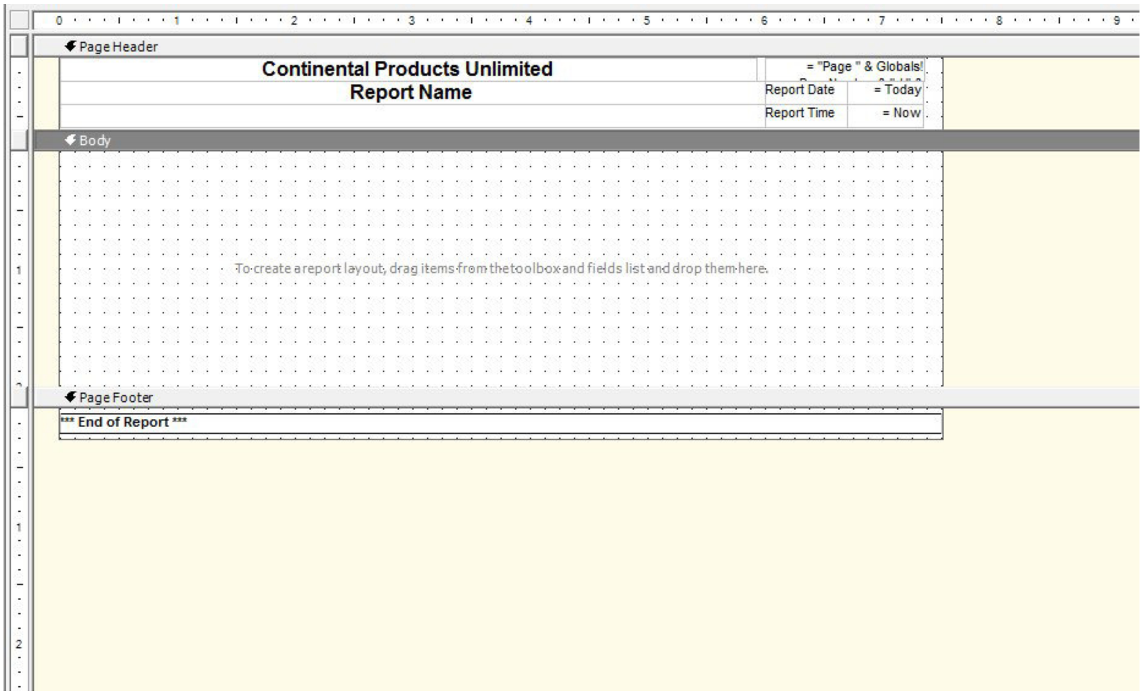
NameProperty	NameNew Value
Size Height	.25 in
Print on First Page	False
Print on Last Page	True

4. Place a **textbox** from the **toolbox** in the layout area for the page footer. Modify the following properties of the text box.

Table 10: Template Footer Textbox

NameProperty	NameNew Value
Border Style Top	Solid
Border Style Bottom	Solid
Font Size	8 pt
Font Weight	Bold
Text Align	Left
Value	*** End of Report ***
Size Width	9 in
Size Height	.1667 in

5. Your template should look similar to this.



To create a report layout, drag items from the toolbox and fields list and drop them here.

6. Click the Save All button in the toolbar.

Report Template, Task 3: Copy the Template to the Report Project Directory

1. From the Main menu, select **File** - **Close Solution** to close the solution.
2. Open **Windows Explorer (My Computer)** and navigate to the folder you created for the Template project. From the My Documents folder the path should be the following:

   Visual Studio 2008\Projects\MSSQLRS\Template

3. In the **Template** folder, highlight the file **TRAVERSETemplate.rdl**. This is the template report we just created.
4. Press **Ctrl C** to copy this file.
5. Navigate to the directory where the Report Designer stores its templates. In a default installation this is

   C:\Program Files\Microsoft Visual Studio 9\Common7\IDE\PrivateAssemblies\ProjectItems\

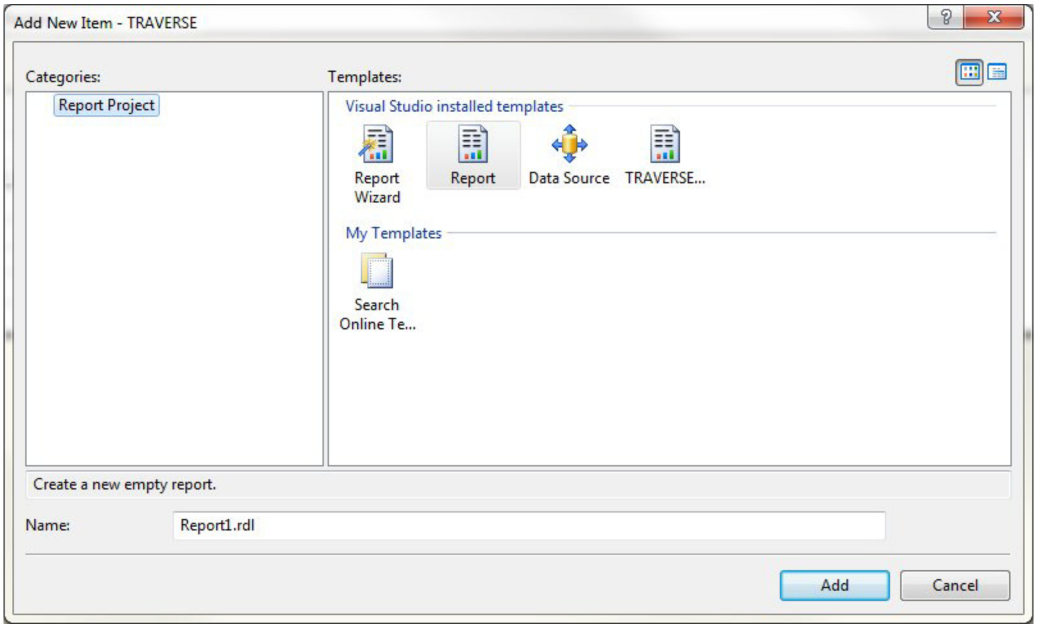
6. Select the **ReportProject** folder.

7. Press **Ctrl V** to paste the copied file in this directory.

8. **Close** Windows Explorer (My Computer).

NOTE: When we add a new item to a report project, the Report Designer looks in the ProjectItems\ReportProject folder. Any report files (.rdl) it finds in this folder are included in the Templates area of the Add New Items Dialog box. This is shown in the picture below.

In the remainder of this lesson, we use our new template to create reports.


Add New Item - TRAVERSE

Categories:

Templates:

Create a new empty report.

Name: Report1.rdl

SALES JOURNAL SUMMARY REPORT

Now that you have a taste of how the Report Wizard works and what it can do, let's try something a bit more complex. Let's create a table report that implements an interactive feature called drilldown. With the *drilldown* type of report, only the high-level summary information is initially presented to the viewers. They can then click a special area of the report (in our case, that area is designated by a plus (+) sign) to reveal part of the lower-level, detail information. The viewers drill down through the summary to get to the detail.

Features Highlighted

- Linking views in the Graphical Query Designer
- Assigning columns for page breaks and grouping
- Enabling subtotals and drilldown
- Copying our Template header and footer into our report

Business Need

The accounting department would like a report listing the Accounts Receivable invoices that will be posted when running the post. The invoices need to be grouped by Batch, with each batch beginning a new page. The report allows a viewer to drill down from the customer level to view the invoices for that customer.

Task Overview

1. Reopen the TRAVERSE Project
2. Create a New Report in the TRAVERSE Project, make a data source and Create a Dataset
3. Choose the Report Layout
4. Copy and paste the Template into the report
5. Final formatting of the report

Sales Journal Summary Report, Task 1: Reopen the TRAVERSE Project

If you have not closed the TRAVERSE project since working on the previous section of this lesson, skip to Step 8. Otherwise, follow these steps, starting with Step 1:

1. Open the **Business Intelligence Development Studio** or **Visual Studio 2005**.

2. If a link to the **TRAVERSE** project is visible on the Start Page, click this link and the TRAVERSE project opens. Proceed to Step 8. If a link to the TRAVERSE project is not visible on the Start Page continue with Step 3.
3. **Select File - Open - Project/Solution**.
4. Click **My Projects**.
5. Double-click **MSSQLRS**.
6. Double-click **TRAVERSE**.
7. Double-click **TRAVERSE.sln**. (This is the file that contains the solution for TRAVERSE.)
8. If the **CustomerList** report is displayed in the center of the screen, click the X button in the upper-right corner or the center section of the screen to close this report.

Task Notes:

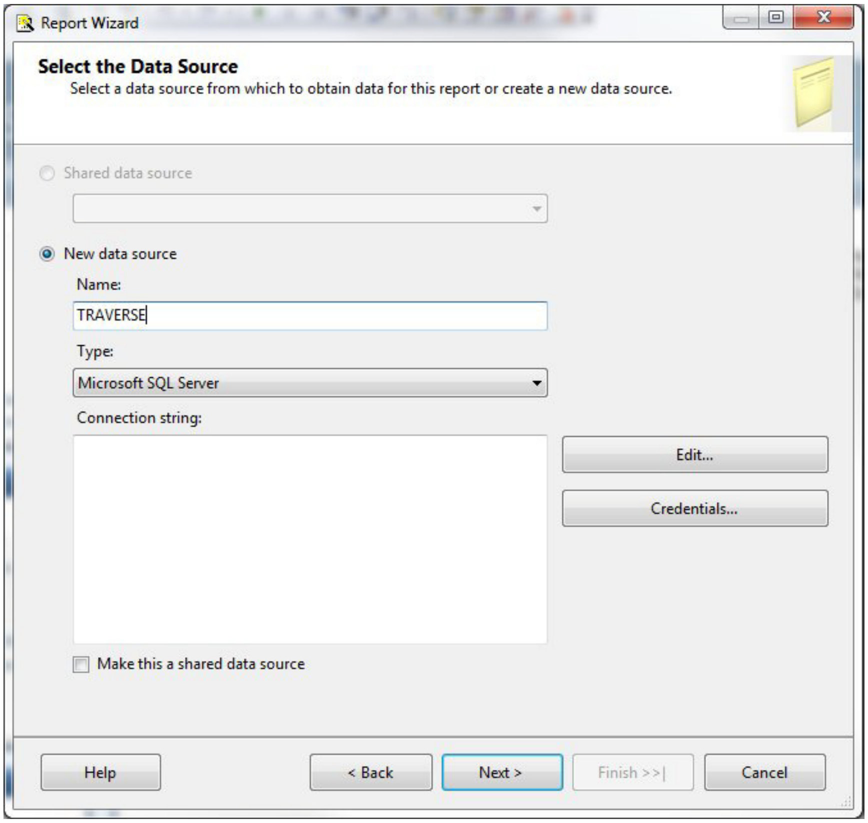
Opening the TRAVERSE solution (TRAVERSE.sln) and opening the TRAVERSE project (TRAVERSE.rptproj) produced the same end result, so you can do either. Only one project is in the TRAVERSE solution, so that project is automatically opened when the solution is opened. When the TRAVERSE project is opened, the last report you worked on is displayed in the center of the screen. In this case, it is probably the Customer List Report.

You do not need to close one report before working on another report. In fact, you can have multiple reports open at one time and use the tables containing the report names to move among them. In most cases, however, a philosophy of “the less clutter, the better” works well when creating reports. For this reason, it is recommended you close all unneeded reports as you move from one report to the next.

Sales Journal Summary Report, Task 2: Create a New Report in the TRAVERSE Project, make a data source and Create a Dataset

1. In the Solution Explorer on the right side of the screen, right-click the Reports folder. You see the right-click menu.
2. Select the **Add New Report** command from the right-click menu. This starts the Report Wizard, enabling you to create another report in the current project.

3. Click **Next**. The **Select the Data Source** page appears.



Report Wizard

Select the Data Source

Select a data source from which to obtain data for this report or create a new data source.

○ Shared data source

● New data source

Name: TRAVERSE

Type: Microsoft SQL Server

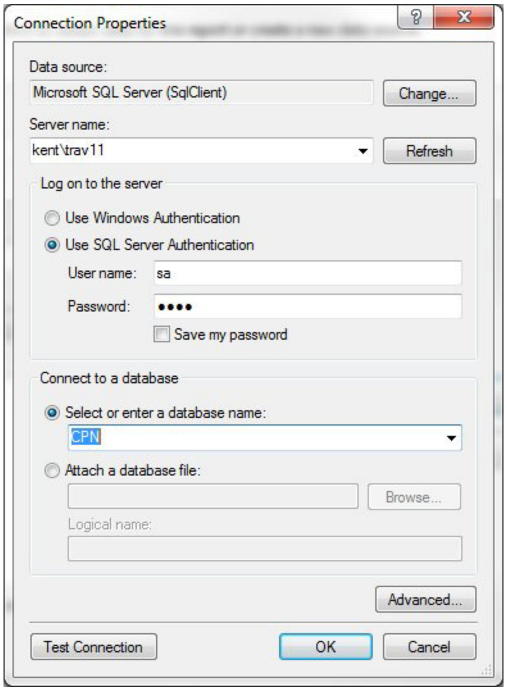
Connection string:

□ Make this a shared data source

4. Select **New Data Source** and type **TRAVERSE** for the New data source **Name**.

5. Select **Microsoft SQL Server** from the Type drop-down list, if it is not already selected.

6. Click **Edit**. The **Connection Properties** dialog box appears.



Connection Properties

Data source: Microsoft SQL Server (SqlClient)

Server name: kent\trav11

Log on to the server

○ Use Windows Authentication

● Use SQL Server Authentication

User name: sa

Password: ••••

□ Save my password

Connect to a database

● Select or enter a database name: CPN

○ Attach a database file:

Logical name:

7. Type the name and instance of the SQL Server that has your TRAVERSE databases.

   If you have a server with an instance name the syntax for that should be similar to this: servername\instancename. Notice the Server Name field in the picture above.

8. Select the **Use SQL Server Authentication** radio button.

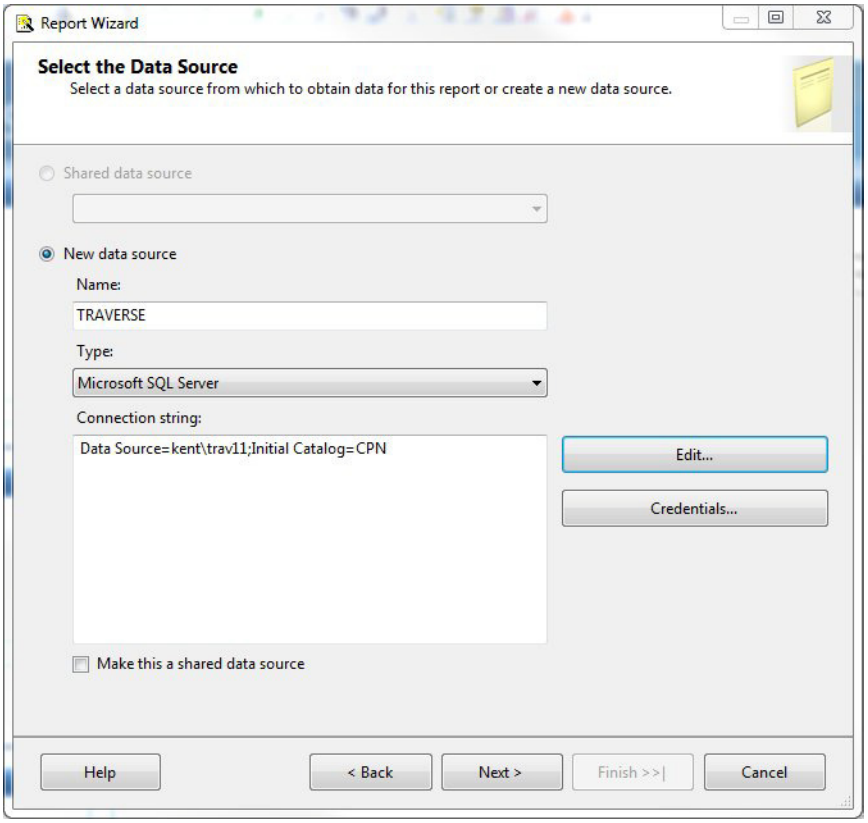
9. Type **sa** as the **Username** and your sa password as the **Password**.

10. Check the **Save my password** box.

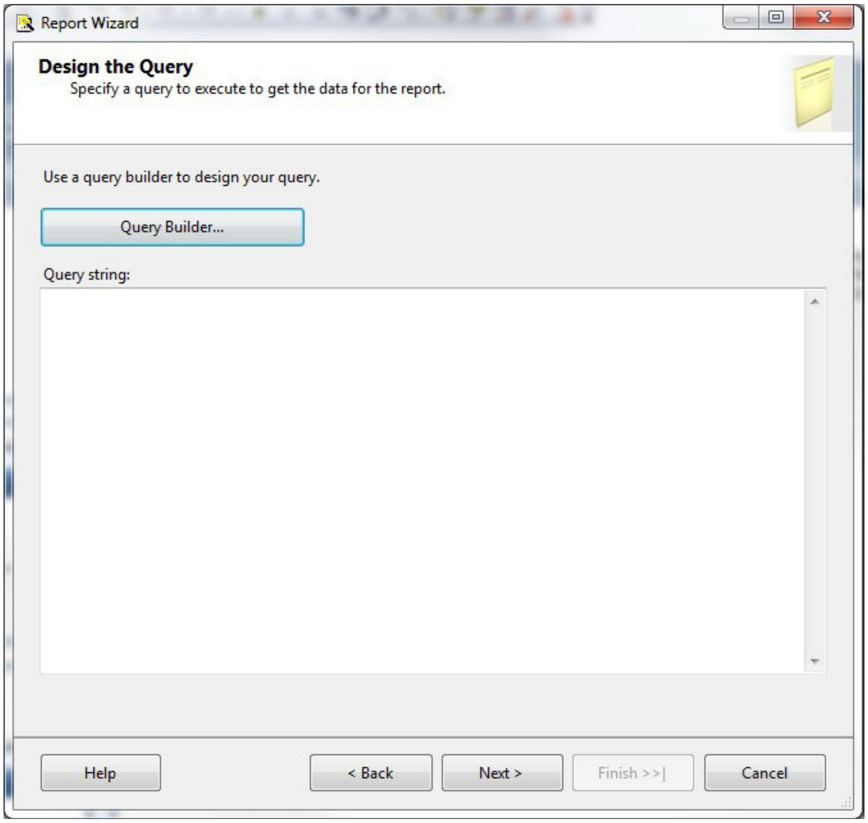
11. Select the TRAVERSE company ID as the database to connect to. For this exercise I am connecting to CPN.

12. Click the **Test Connection** button. If the message “Test connection succeeded” appears, click **OK**. If an error appears make sure you typed the information correctly into all the fields.

13. Click **OK** to return to the **Select the Data Source** page. The Select Data Source page should now look similar to this.



14. Click **Next**. The **Design the Query** page of the Report Wizard appears.



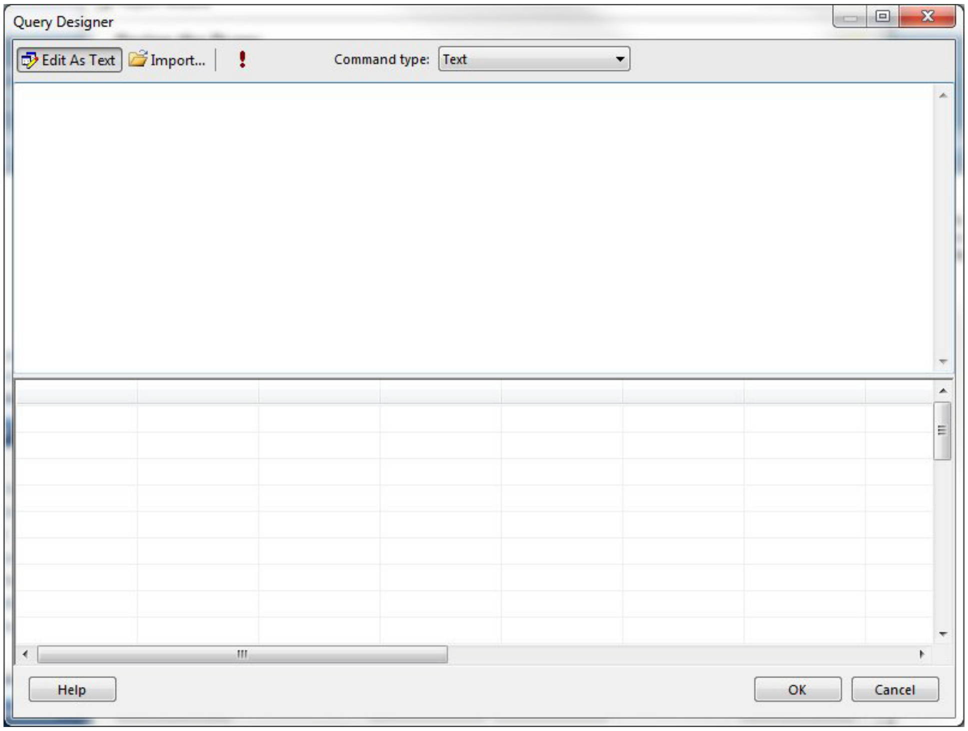
Design the Query

Specify a query to execute to get the data for the report.

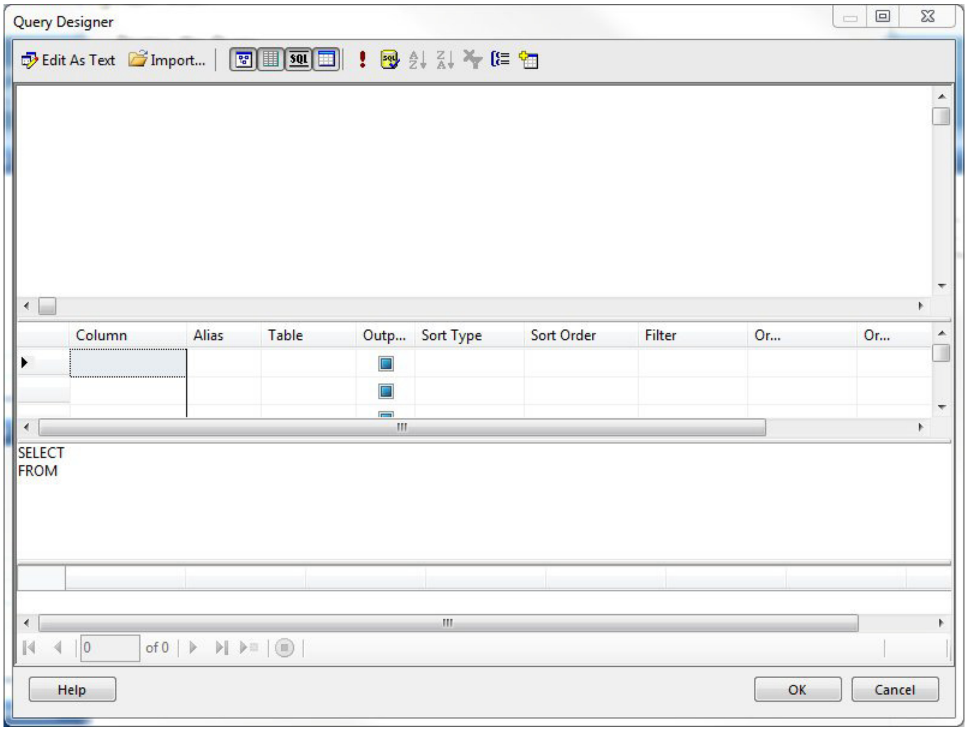
Use a query builder to design your query.

Query string:

15. Click **Query Builder**. The **Query Builder** window appears with the Generic Query Designer active.



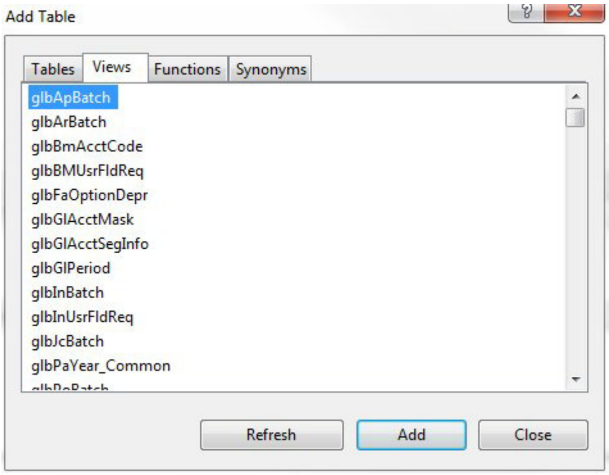
16. Click the **Query Designer** toggle button . The Query Designer will switch to the **Graphical Query Designer**.



```
SELECT
FROM
```

17. Select **Add Table** from the right-click menu or click on the **Add Table** button in the toolbar .

18. The **Add Table** dialog box appears. The dialog box contains a list of the tables, views, and functions that return datasets, which are found in the data source.

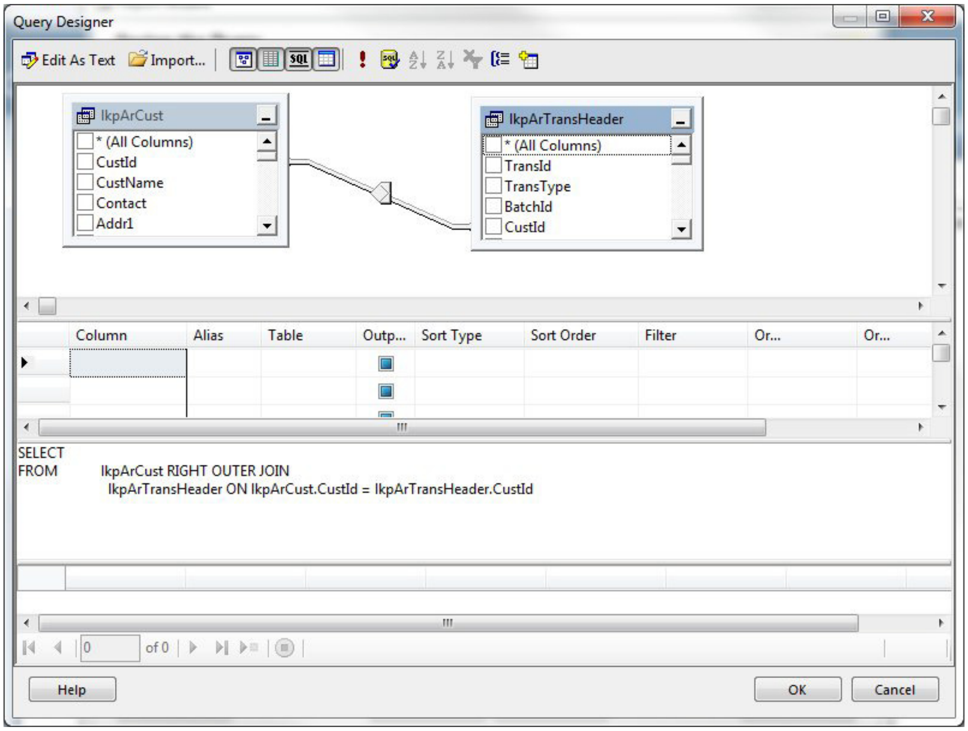


glbApBatch
glbArBatch
glbBmAcctCode
glbBMUsrFldReq
glbFaOptionDepr
glbGlAcctMask
glbGlAcctSegInfo
glbGlPeriod
glbInBatch
glbInUsrFldReq
glbJcBatch
glbPaYear_Common

19. Click the **Views** tab and scroll down to find the table **lkpArCust** and double click on it, or highlight it and click **Add**.

20. Scroll down a little further and find the view **lkpArTransHeader** and double click on it or highlight it and click **Add**. Make sure you select the **lkpArTransHeader** and **NOT** the **lkpArTransDetail** view.

21. Click **Close** to exit the Add Table dialog box. We need to create the INNER JOIN between the **lkpArCust** and the **lkpArTransHeader** tables. Click on the **CustId** in the **lkpArCust** and drag it over on top of the **CustId** in the **lkpArTransHeader** to make the join. Your query designer should look as shown below.



```
SELECT
FROM        lkpArCust RIGHT OUTER JOIN
              lkpArTransHeader ON lkpArCust.CustId = lkpArTransHeader.CustId
```

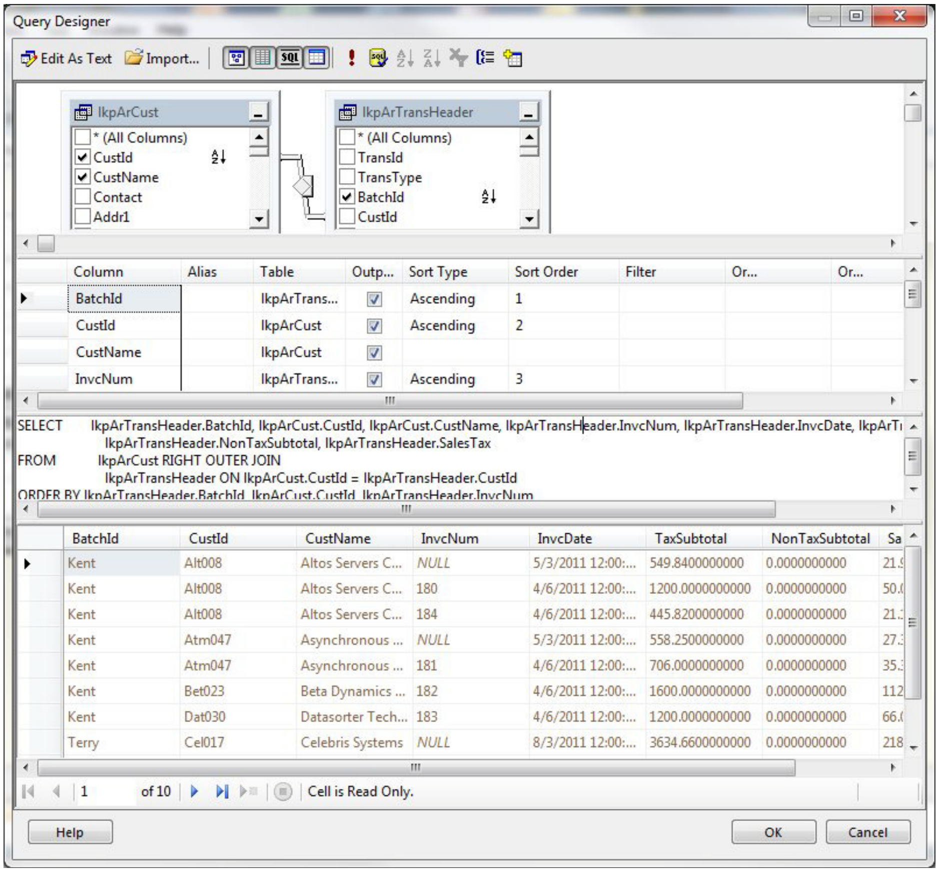
22. Right click the grey diamond in the middle of the link joining the **lkpArCust** and the **lkpArTransHeader** views. The Join right-click menu is displayed.

23. Choose the **Select All Rows from lkpArTransHeader** option in the right-click menu. The diamond symbol changes. It now has a half square on the right side.

24. Select the **BatchId** field in the **lkpArTransHeader** view list of fields. This places the field in the resulting SQL query and Column list.

25. Scroll down the list of columns for the **lkpArCust** view and check the box next to the **CustID** and **CustName** fields. This places the fields in the resulting SQL query and Column list.

26. Select the following fields in the **IkpArTransHeader** view list of fields: **InvcNum**, **InvcDate**, **TaxSubtotal**, **NonTaxSubtotal** and **SalesTax**.



	IkpArCust
□	* (All Columns)
☑	CustId
☑	CustName
□	Contact
□	Addr1

	IkpArTransHeader
□	* (All Columns)
□	TransId
□	TransType
☑	BatchId
□	CustId

Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...
BatchId		IkpArTrans...	☑	Ascending	1			
CustId		IkpArCust	☑	Ascending	2			
CustName		IkpArCust	☑					
InvcNum		IkpArTrans...	☑	Ascending	3			

```
SELECT     IkpArTransHeader.BatchId, IkpArCust.CustId, IkpArCust.CustName, IkpArTransHeader.InvcNum, IkpArTransHeader.InvcDate, IkpArTı
              IkpArTransHeader.NonTaxSubtotal, IkpArTransHeader.SalesTax
FROM        IkpArCust RIGHT OUTER JOIN
              IkpArTransHeader ON IkpArCust.CustId = IkpArTransHeader.CustId
ORDER BY IkpArTransHeader.BatchId, IkpArCust.CustId, IkpArTransHeader.InvcNum
```

BatchId	CustId	CustName	InvcNum	InvcDate	TaxSubtotal	NonTaxSubtotal	Sa
Kent	Alt008	Altos Servers C...	*NULL*	5/3/2011 12:00:...	549.8400000000	0.0000000000	21.
Kent	Alt008	Altos Servers C...	180	4/6/2011 12:00:...	1200.0000000000	0.0000000000	50.
Kent	Alt008	Altos Servers C...	184	4/6/2011 12:00:...	445.8200000000	0.0000000000	21.
Kent	Atm047	Asynchronous ...	*NULL*	5/3/2011 12:00:...	558.2500000000	0.0000000000	27.
Kent	Atm047	Asynchronous ...	181	4/6/2011 12:00:...	706.0000000000	0.0000000000	35.
Kent	Bet023	Beta Dynamics ...	182	4/6/2011 12:00:...	1600.0000000000	0.0000000000	112
Kent	Dat030	Datasorter Tech...	183	4/6/2011 12:00:...	1200.0000000000	0.0000000000	66.
Terry	Cel017	Celebris Systems	*NULL*	8/3/2011 12:00:...	3634.6600000000	0.0000000000	218

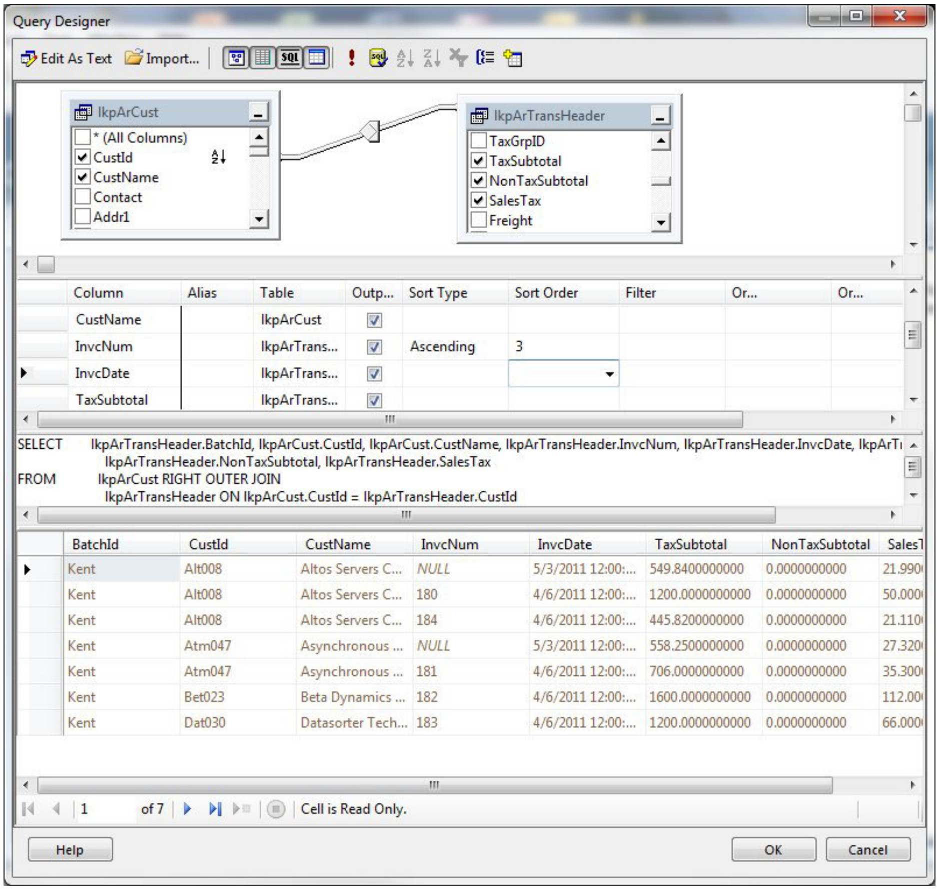
1 of 10 Cell is Read Only.

27. Place a **1** in the **Sort Order** column for the **BatchId** field either by typing in the cell or by using the drop down list.

28. Place a **2** in the **Sort Order** column for the **CustId** field.

29. Place a **3**in the **Sort Order** column for the **InvoiceNum** field.

30. Right-click in the SQL pane and select **Execute SQL** or click on the execute button ! . The query executes and the result set is displayed in the results pane. The Query Designer should appear similar to the following illustration.



Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...
CustName		IkpArCust	☑					
InvcNum		IkpArTrans...	☑	Ascending	3			
InvcDate		IkpArTrans...	☑					
TaxSubtotal		IkpArTrans...	☑					

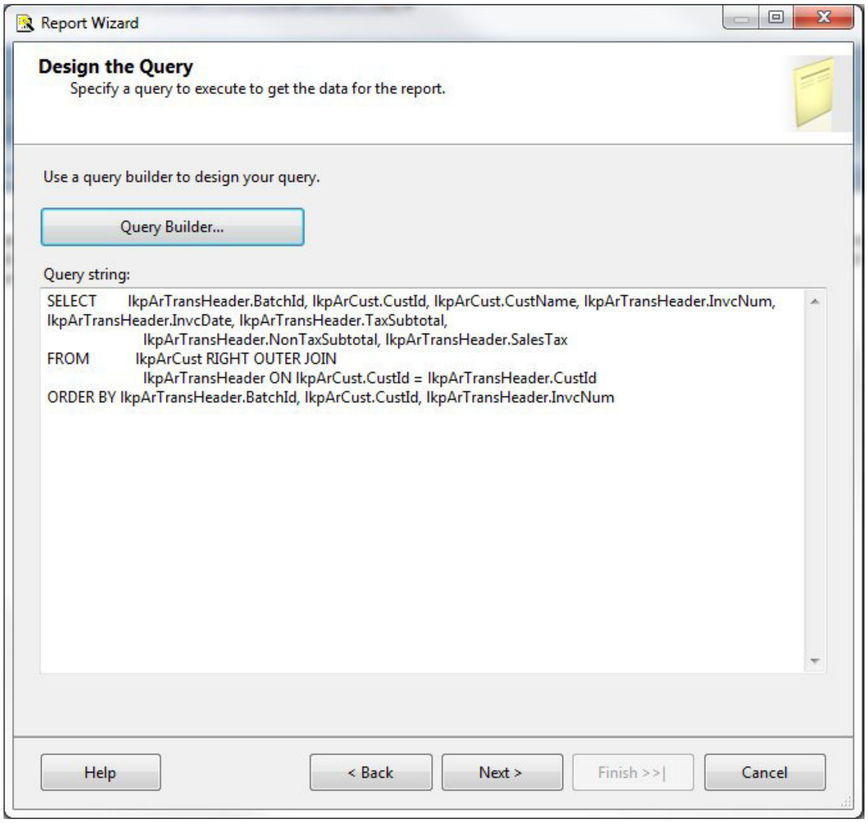
```
SELECT     IkpArTransHeader.BatchId, IkpArCust.CustId, IkpArCust.CustName, IkpArTransHeader.InvcNum, IkpArTransHeader.InvcDate, IkpArTı
             IkpArTransHeader.NonTaxSubtotal, IkpArTransHeader.SalesTax
FROM       IkpArCust RIGHT OUTER JOIN
             IkpArTransHeader ON IkpArCust.CustId = IkpArTransHeader.CustId
```

BatchId	CustId	CustName	InvcNum	InvcDate	TaxSubtotal	NonTaxSubtotal	SalesT
Kent	Alt008	Altos Servers C...	*NULL*	5/3/2011 12:00:...	549.8400000000	0.0000000000	21.990
Kent	Alt008	Altos Servers C...	180	4/6/2011 12:00:...	1200.0000000000	0.0000000000	50.000
Kent	Alt008	Altos Servers C...	184	4/6/2011 12:00:...	445.8200000000	0.0000000000	21.110
Kent	Atm047	Asynchronous ...	*NULL*	5/3/2011 12:00:...	558.2500000000	0.0000000000	27.320
Kent	Atm047	Asynchronous ...	181	4/6/2011 12:00:...	706.0000000000	0.0000000000	35.300
Kent	Bet023	Beta Dynamics ...	182	4/6/2011 12:00:...	1600.0000000000	0.0000000000	112.00
Kent	Dat030	Datasorter Tech...	183	4/6/2011 12:00:...	1200.0000000000	0.0000000000	66.000

1 of 7 Cell is Read Only.

31. Right-click the results pane. Select **Clear Results** from the right-click menu.

32. Click **OK**. This returns you to the **Design the Query** page.



Design the Query

Specify a query to execute to get the data for the report.

Use a query builder to design your query.

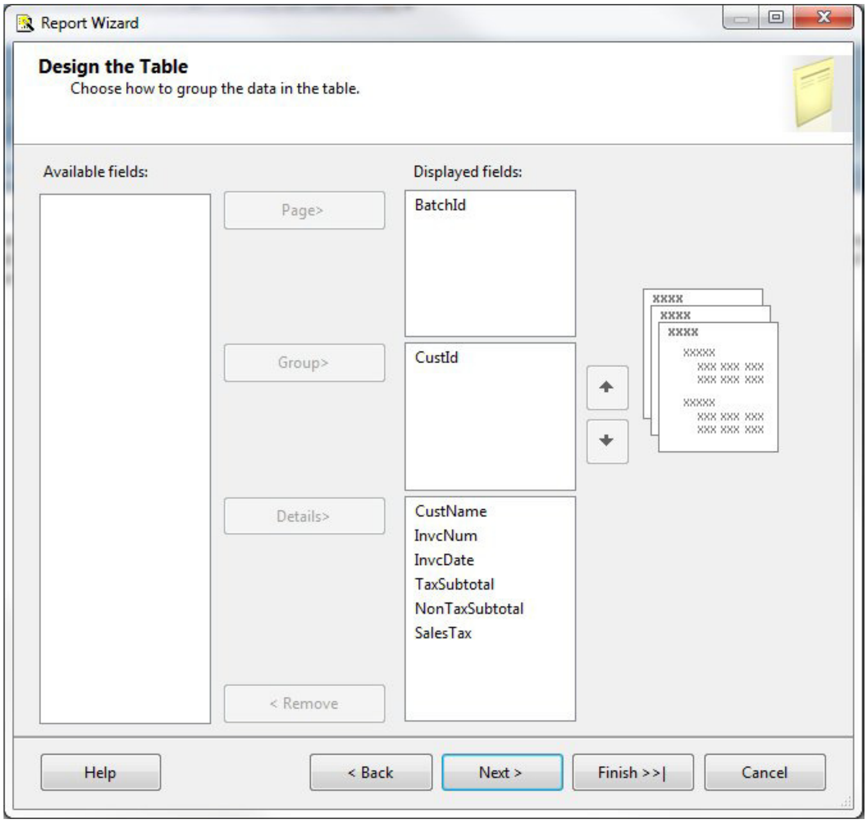
Query string:

```
SELECT     IkpArTransHeader.BatchId, IkpArCust.CustId, IkpArCust.CustName, IkpArTransHeader.InvcNum,
IkpArTransHeader.InvcDate, IkpArTransHeader.TaxSubtotal,
                 IkpArTransHeader.NonTaxSubtotal, IkpArTransHeader.SalesTax
FROM        IkpArCust RIGHT OUTER JOIN
                 IkpArTransHeader ON IkpArCust.CustId = IkpArTransHeader.CustId
ORDER BY IkpArTransHeader.BatchId, IkpArCust.CustId, IkpArTransHeader.InvcNum
```

Sales Journal Summary Report, Task 3: Choose the Report Layout

1. Click **Next**. The **Select the Report Type** page of the Report Wizard appears.
2. Make sure the **Tabular** radio button is selected and click **Next**. The **Design the Table** page of the Report Wizard appears.
3. With the **BatchId** field highlighted in the Available Fields list click **Page**. The BatchId field is moved to the Displayed Fields list.
4. With the **CustId** field highlighted in the Available Fields list click **Group**. The CustId field is moved to the Displayed Fields list.
5. With the **CustName** field highlighted in the Available Fields list click **Detail**. The CustName field is moved to the Displayed Fields list.

6. Move the remaining fields to the **Detail** Displayed Fields list. Your Design the Table screen should appear like the picture below.



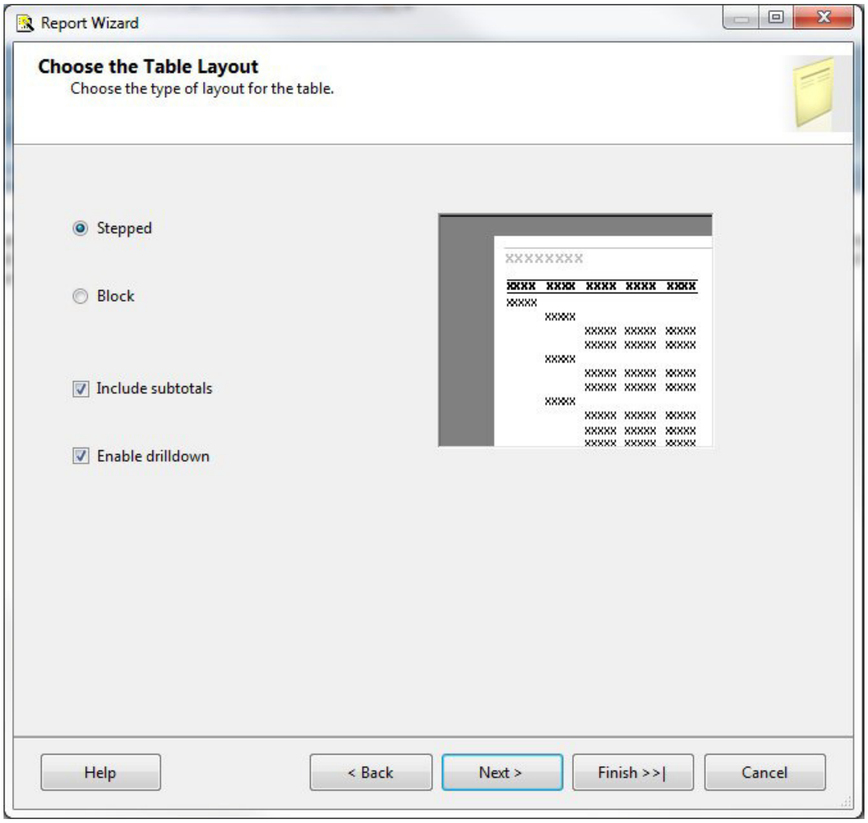
Displayed fields:

BatchId

CustId

CustName
InvcNum
InvcDate
TaxSubtotal
NonTaxSubtotal
SalesTax

7. Click **Next**. The **Choose the Table Layout** page of the Report Wizard appears. This page appears in the Report Wizard because we put fields in the Group area on the Design the Table page.



Report Wizard

Choose the Table Layout

Choose the type of layout for the table.

● Stepped

○ Block

☑ Include subtotals

☑ Enable drilldown

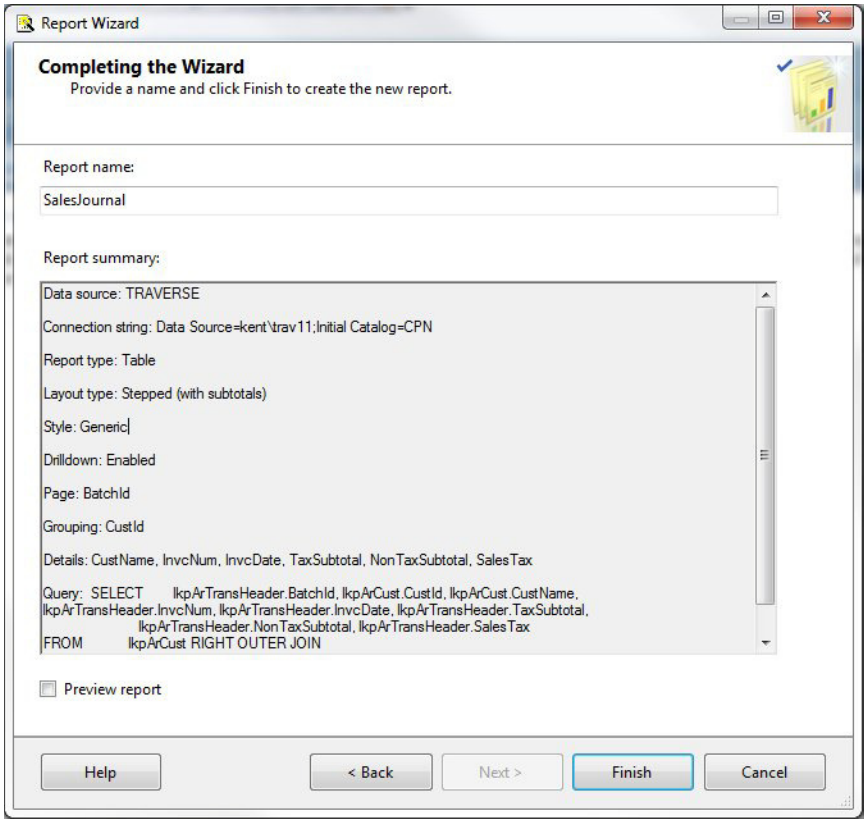
8. Make sure the **Stepped** radio button is selected if it already isn't selected.

9. Check the **Include Subtotals** check box.

10. Check the **Enable Drilldown** check box. The **Choose the Table Layout** page should look like the picture above.

11. Click **Next**. The **Choose the Table Style** page of the Report Wizard appears.

12. Select **Generic** in the style list, and then click **Next**. The **Completing the Wizard** page appears.



Completing the Wizard

Provide a name and click Finish to create the new report.

Report name:

SalesJournal

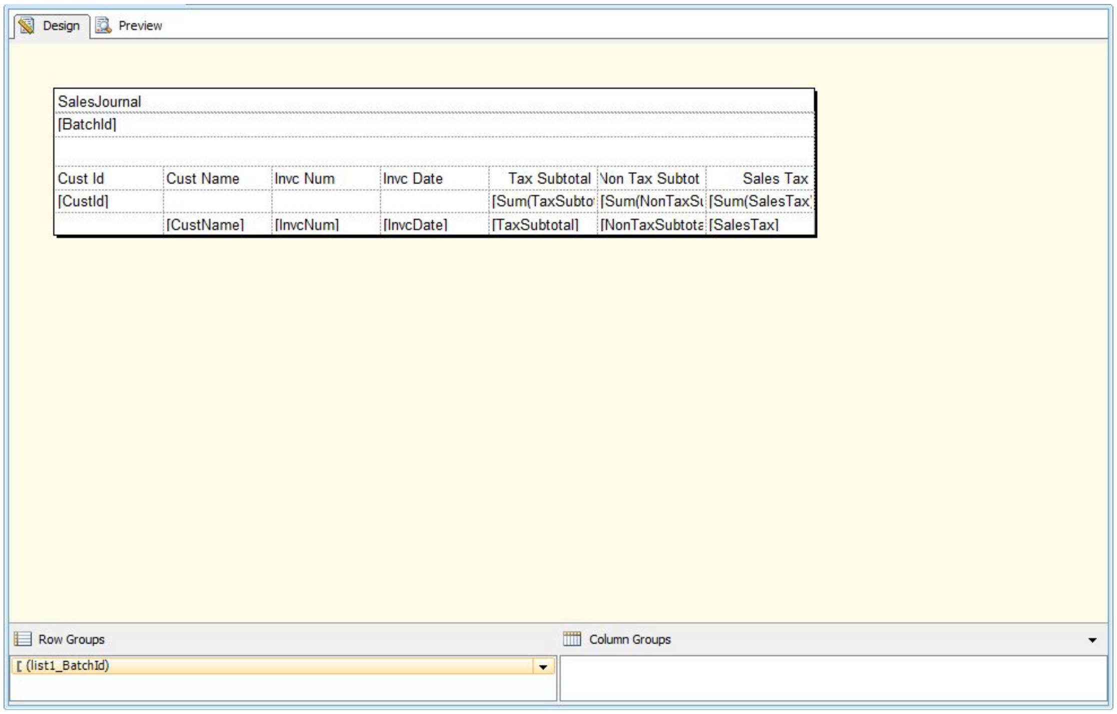
Report summary:

```
Data source: TRAVERSE

Connection string: Data Source=kent\trav11;Initial Catalog=CPN

Report type: Table

Layout type: Stepped (with subtotals)

Style: Generic

Drilldown: Enabled

Page: BatchId

Grouping: CustId

Details: CustName, InvcNum, InvcDate, TaxSubtotal, NonTaxSubtotal, SalesTax

Query:  SELECT      tkpArTransHeader.BatchId, tkpArCust.CustId, tkpArCust.CustName,
tkpArTransHeader.InvcNum, tkpArTransHeader.InvcDate, tkpArTransHeader.TaxSubtotal,
                    tkpArTransHeader.NonTaxSubtotal, tkpArTransHeader.SalesTax
FROM          tkpArCust RIGHT OUTER JOIN
```

□ Preview report

13. Type **SalesJournal** for the **Report Name**.

14. Click **Finish**. The **Report Designer** window appears with the **Layout** tab as shown below.



SalesJournal						
[BatchId]						
Cust Id	Cust Name	Invc Num	Invc Date	Tax Subtotal	Non Tax Subtot	Sales Tax
[CustId]				[Sum(TaxSubto	[Sum(NonTaxSu	[Sum(SalesTax
	[CustName]	[InvcNum]	[InvcDate]	[TaxSubtotal]	[NonTaxSubtota	[SalesTax]

This is a very basic setup of a drill down report. The title is very generic and the page designation field (BatchId) has no label on it, so we will not know what that field is.

We would want the title to be larger and in the center of the page. We also would want a label for our BatchId field so we know what each page break contains.

15. With the **SalesJournal** textbox selected change the following properties.

Table 11: Sales Journal Title

NameProperty	NameNew Value
Font Size	12 pt
Font Weight	Bold
Text Align	Center
Value	Sales Journal
Size Height	.2 in

16. Select the **BatchId** text box and change the following properties.

Table 12: Batch ID Value

NameProperty	NameNew Value
Location Left	1.125 in
Location Top	.25 in
Size Width	1.875 in
Size Height	.25

17. From the **Toolbox** place a textbox next to the BatchId data field and edit the following properties.

Table 13: Batch ID Label

NameProperty	NameNew Value
Font Weight	Bold
Text Decoration	Underline
Value	Batch ID:
Location Left	.125 in
Location Top	.25 in
Size Width	.75 in
Size Height	.25 in

18. Now let's make the table headings look better.

19. Click in a cell in the table and then click the grey box to the left of the headings row and change the following properties.

Table 14: Table Headings Properties

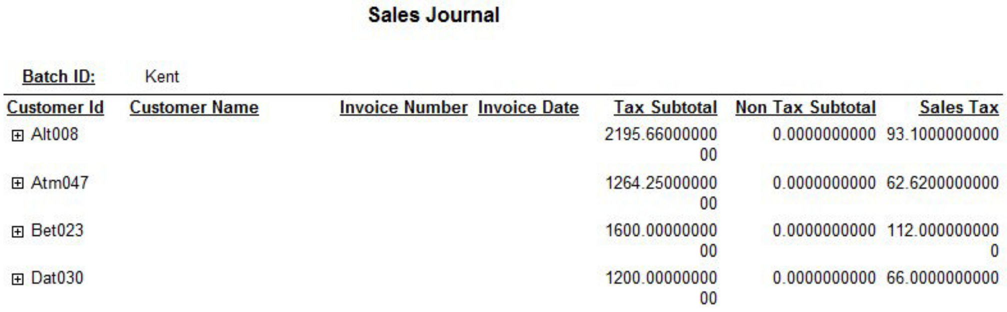
NameProperty	NameNew Value
Border Style Top	Solid
Font Size	10 pt
Font Weight	Bold

Table 14: Table Headings Properties

NameProperty	NameNew Value
Text Decoration	Underline
Layout Height	.21 in

20. Click in each of the heading cells and change the text in the cells to have better titles. For example Customer ID, Customer Name, Tax Subtotal etc.....

21. Click on the **Preview** tab and your report should look similar to this.



Sales Journal

Batch ID: Kent

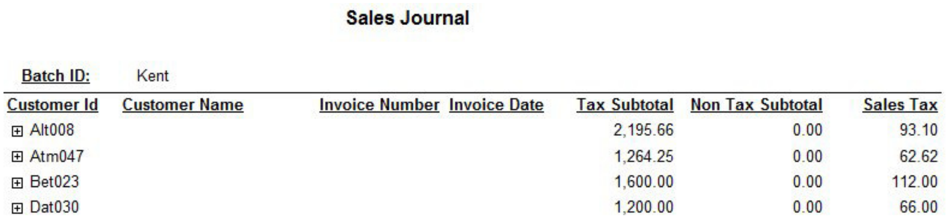
Customer Id	Customer Name	Invoice Number	Invoice Date	Tax Subtotal	Non Tax Subtotal	Sales Tax
⊞ Alt008				2195.6600000000	0.0000000000	93.1000000000
⊞ Atm047				1264.2500000000	0.0000000000	62.6200000000
⊞ Bet023				1600.0000000000	0.0000000000	112.0000000000
⊞ Dat030				1200.0000000000	0.0000000000	66.0000000000

22. The numbers in the **Tax Subtotal**, **Non Tax Subtotal** and **Sales Tax** fields are not formatted very well.

23. Go back to the **Design** tab and click in the cell containing **=Sum(TaxSubtotal.Value)**, hold your shift key down and click in the field containing **[SalesTax]**.

24. In the properties area find the **Format** field and enter **n2** into this field.

25. Click the **Preview** tab and your report should now look like this.

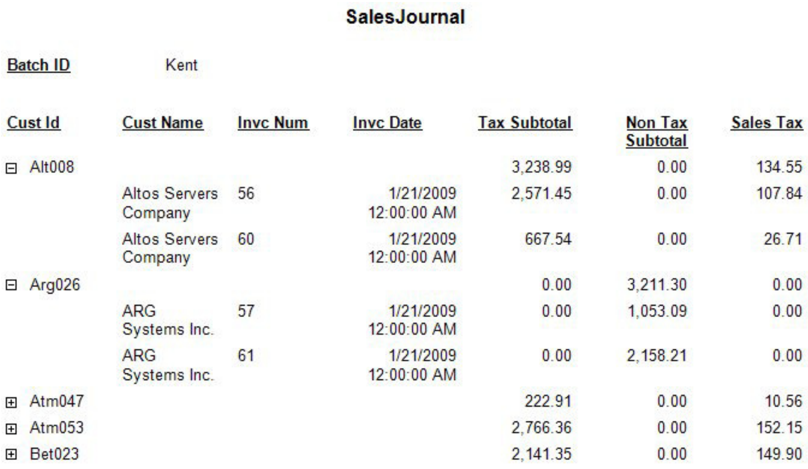


Sales Journal

Batch ID: Kent

Customer Id	Customer Name	Invoice Number	Invoice Date	Tax Subtotal	Non Tax Subtotal	Sales Tax
⊞ Alt008				2,195.66	0.00	93.10
⊞ Atm047				1,264.25	0.00	62.62
⊞ Bet023				1,600.00	0.00	112.00
⊞ Dat030				1,200.00	0.00	66.00

26. Click the plus (**+**) next to the first record to expand the customer record to see the invoices for that customer. Your report should look similar to this.



SalesJournal

Batch ID Kent

Cust Id	Cust Name	Invc Num	Invc Date	Tax Subtotal	Non Tax Subtotal	Sales Tax
⊟ Alt008				3,238.99	0.00	134.55
	Altos Servers Company	56	1/21/2009 12:00:00 AM	2,571.45	0.00	107.84
	Altos Servers Company	60	1/21/2009 12:00:00 AM	667.54	0.00	26.71
⊟ Arg026				0.00	3,211.30	0.00
	ARG Systems Inc.	57	1/21/2009 12:00:00 AM	0.00	1,053.09	0.00
	ARG Systems Inc.	61	1/21/2009 12:00:00 AM	0.00	2,158.21	0.00
⊞ Atm047				222.91	0.00	10.56
⊞ Atm053				2,766.36	0.00	152.15
⊞ Bet023				2,141.35	0.00	149.90

27. Notice the **Invc Date** field is not formatted very well either. Go back to the **Layout** tab and click in the cell containing **[InvcDate]**.

28. In the Properties area find the **Format** field and enter **d** for the format. This will give us just the date and not the time.

29. We want to incorporate our template into this report so we have our standardized report header and report footer.

30. In the Solution area of the screen double click the **TRAVERSETemplate** report, if you do not have it open in a different tab.

31. This will open our template.

32. Click the first text box in the **Page Header** to select it. Hold your shift key and select each of the other text boxes and press **Ctrl C** on your keyboard. This will copy the page header.

33. Click back to the tab containing our **Sales Journal** report and select **Report** from the main menu and select **Report Header** and **Report Footer.** This will add the report header and footer fields to the report.

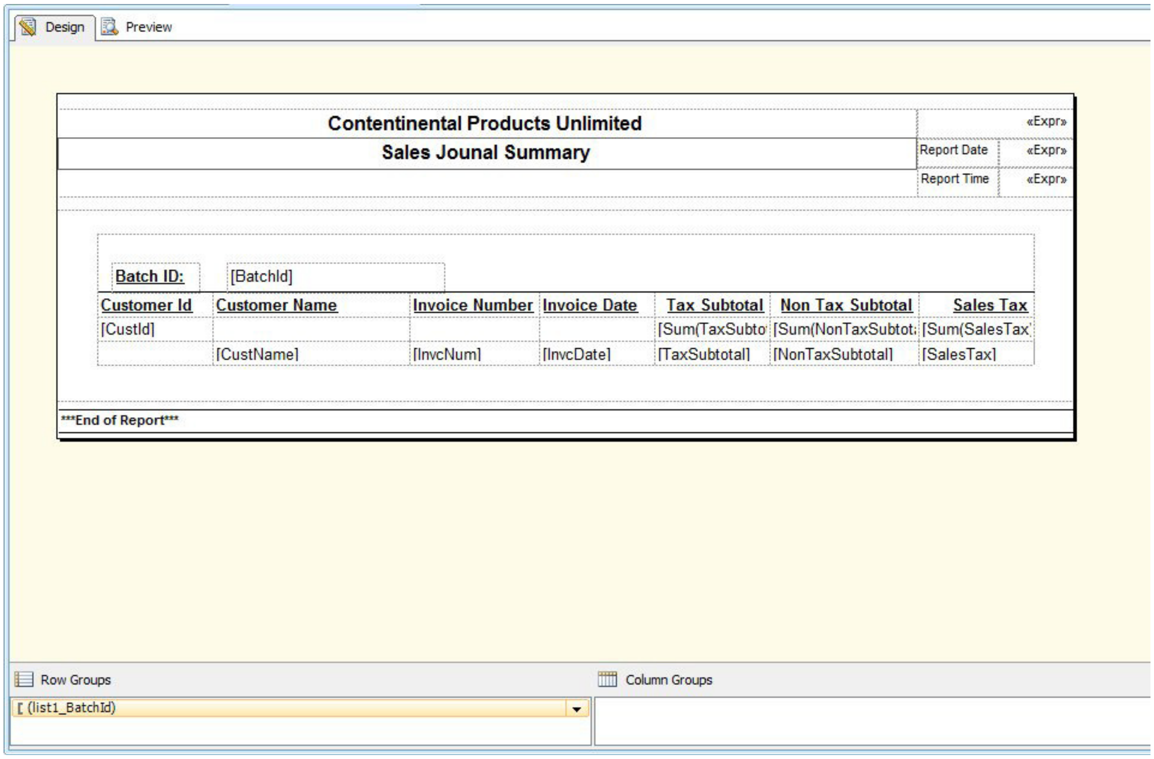
34. Click the **Page Header** bar and press **Ctrl V** to paste our template report header into our report.

35. Go back to the **TRAVERSETemplate** tab and click the text box in the **Report Footer** bar and press **Ctrl C** to copy our report footer.

36. Go to the **Sales Journal** tab and select the **Report Footer** and press **Ctrl V** to paste our template report footer.

37. Highlight the **Sales Journal** text box at the top of the **Body** section of the report and press **Delete** on your keyboard.

38. Your report layout should now look similar to this.



Contentinental Products Unlimited «Expr»

Sales Jounal Summary Report Date «Expr»

Report Time «Expr»

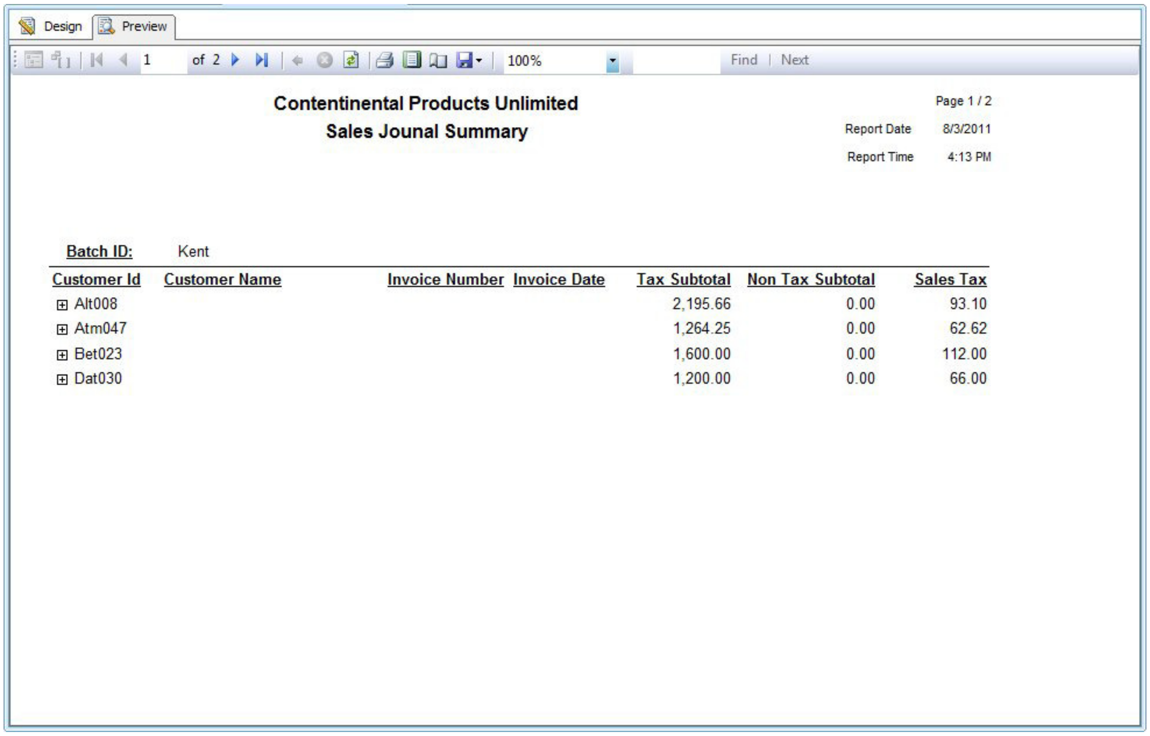
Batch ID: [BatchId]

Customer Id	**Customer Name**	**Invoice Number**	**Invoice Date**	**Tax Subtotal**	**Non Tax Subtotal**	**Sales Tax**
[CustId]				[Sum(TaxSubto	[Sum(NonTaxSubtot:	[Sum(SalesTax
	[CustName]	[InvcNum]	[InvcDate]	[TaxSubtotal]	[NonTaxSubtotal]	[SalesTax]

End of Report

[(list1_BatchId)

39. Click the **Preview** tab and your report should look similar to this.



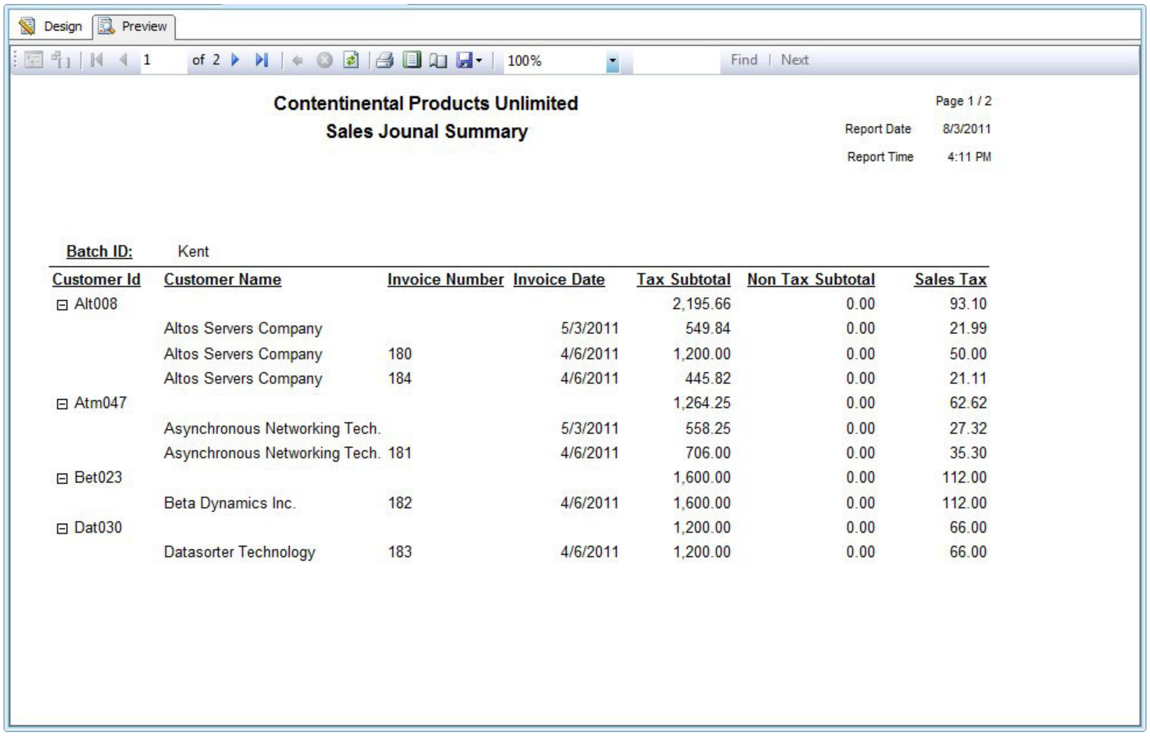
Contentinental Products Unlimited
Sales Jounal Summary

Page 1 / 2
Report Date 8/3/2011
Report Time 4:13 PM

Batch ID: Kent

Customer Id	Customer Name	Invoice Number	Invoice Date	Tax Subtotal	Non Tax Subtotal	Sales Tax
⊞ Alt008				2,195.66	0.00	93.10
⊞ Atm047				1,264.25	0.00	62.62
⊞ Bet023				1,600.00	0.00	112.00
⊞ Dat030				1,200.00	0.00	66.00

40. Click the plus (+) next a couple of the customers and your report should now look similar to this.



Contentinental Products Unlimited
Sales Jounal Summary

Page 1 / 2
Report Date 8/3/2011
Report Time 4:11 PM

Batch ID: Kent

Customer Id	Customer Name	Invoice Number	Invoice Date	Tax Subtotal	Non Tax Subtotal	Sales Tax
⊟ Alt008				2,195.66	0.00	93.10
	Altos Servers Company		5/3/2011	549.84	0.00	21.99
	Altos Servers Company	180	4/6/2011	1,200.00	0.00	50.00
	Altos Servers Company	184	4/6/2011	445.82	0.00	21.11
⊟ Atm047				1,264.25	0.00	62.62
	Asynchronous Networking Tech.		5/3/2011	558.25	0.00	27.32
	Asynchronous Networking Tech.	181	4/6/2011	706.00	0.00	35.30
⊟ Bet023				1,600.00	0.00	112.00
	Beta Dynamics Inc.	182	4/6/2011	1,600.00	0.00	112.00
⊟ Dat030				1,200.00	0.00	66.00
	Datasorter Technology	183	4/6/2011	1,200.00	0.00	66.00

Task Notes:

When we created the Customer List Report, we put all but one of the columns from the dataset into the detail line of the report. This time we put the BatchID column in the page area of the table layout. Because of this, the Report Wizard created a report that begins a new page every time there is a new value in the BatchID column. In addition, the value of the Batch ID column appears at the top of each report page.

Using the Report Wizard, we put the CustID column in the Group area of the table layout. This means the report will create a new group each time the value of the CustID column changes.

By checking the Enable Drilldown checkbox, you told the Report Wizard to create a report where the detail lines for a group become visible when the plus sign for that group is clicked. By checking the Include Subtotals checkbox, you told the Report Wizard to total any numeric columns in the detail and to show those totals in the group header for each group.

We had to reformat the number and date fields because the data coming from the SQL table were not formatted very well coming into the report. We also included the template information in our report.

SALES HISTORY REPORT

In our previous reports we have used the Report Wizard. This is like learning to ride your first two-wheeler with the training wheels on. Not it is time for the training wheels to come off, so we can see what this application can really do! We are going to start by building a report from scratch. We hope these next sections provide the hand holding you need to generate some basic reports without too much programming knowledge.

First we will work with building a report using a table with a couple drill downs without using the Report Wizard. From there we will use this same report to add selection criteria at the top of the report so you can filter the report to get just the information you want.

Features Highlighted

- Build a Dataset using the Graphical Query Designer to bring in data from 3 different views.
- Create a table report from scratch.
- Add grouping and drill down for multiple levels.
- Add subtotals by invoice number for items in the invoice.

Business Need

The Accounts Receivable department at Continental Products Unlimited needs to be able to print a sales history report sorted and grouped by Customer ID. They also need to be able to see the invoices for each customer and then if needed see the items on those invoices. They want to have the Customer ID, Customer Name, Invoice Number, Invoice Date, Item ID and Description, Item Unit Cost and Price, Extended Cost and Price, Sales Tax and Invoice totals.

Task Overview

1. Create a New Report from our Template in the TRAVERSE project.
2. Create a Dataset.
3. Place a Table Item on the report, Populate It and set up table groupings.
4. Final Report Formatting.

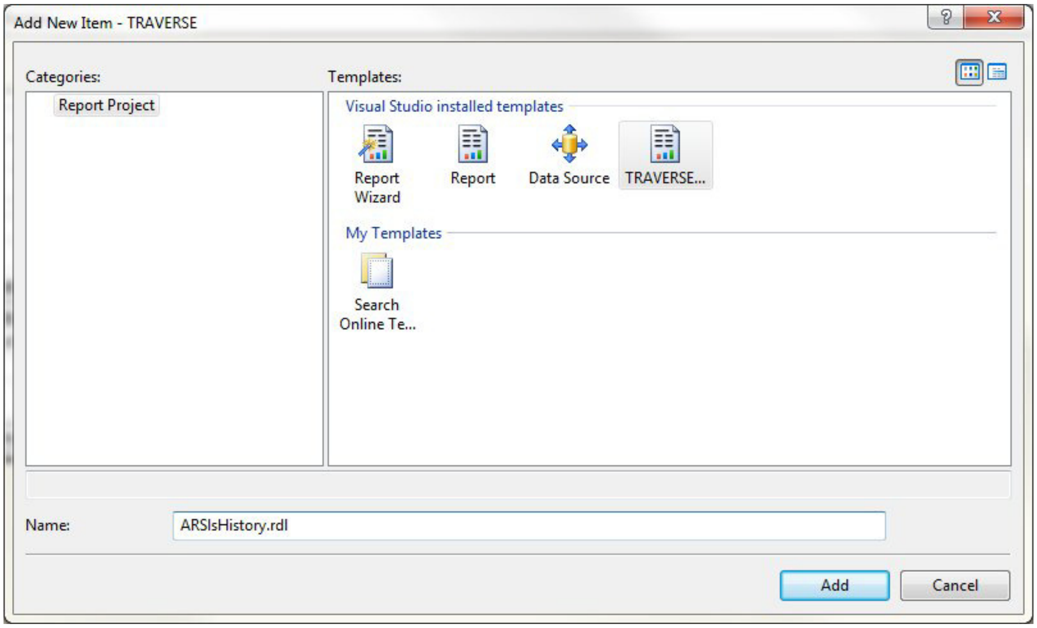
Sales History Report, Task 1: Create a New Report from our Template in the TRAVERSE Project.

1. If you closed the TRAVERSE project, reopen it.

2. In the Solution Explorer area, right-click the Reports folder.

3. Put your mouse pointer over **Add** in the right-click menu and wait for a submenu to appear. Select the **New Item** command from the menu.

4. The **Add New Item - TRAVERSE** dialog box appears. Make sure the TRAVERSE Template icon is selected in the Templates area. Enter **ARSlsHistory** for the name.

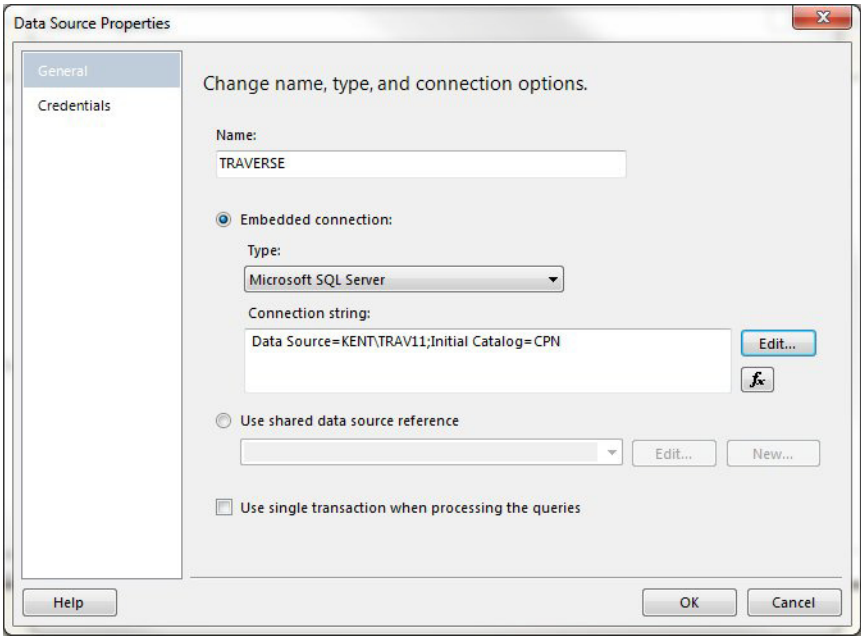


Name: ARSlsHistory.rdl

5. Click **Add**. A new report called **ARSlsHistory.rdl** is created in the TRAVERSE project. You are taken to the **Data** tab of this new report.

Sales History Report, Task 2: Create a Dataset

1. Select **<New Dataset...>** from the **Dataset** drop-down list. The Data Source dialog box appears.



Data Source Properties

Change name, type, and connection options.

Name: TRAVERSE

● Embedded connection:

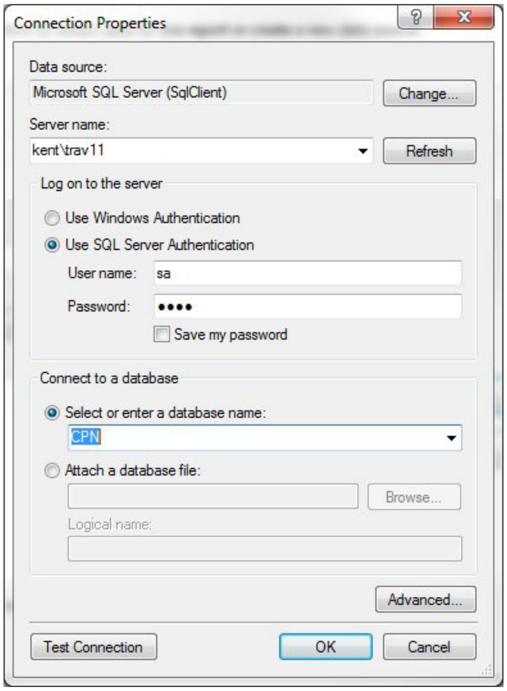
Type: Microsoft SQL Server

Connection string: Data Source=KENT\TRAV11;Initial Catalog=CPN

○ Use shared data source reference

□ Use single transaction when processing the queries

2. Enter **TRAVERSE** as the name and select **Microsoft SQL Server** from the Data source drop down list. Click the **Edit** button to bring up the **Connection Properties** dialog box.



Data source: Microsoft SQL Server (SqlClient)

Server name: kent\trav11

Log on to the server

○ Use Windows Authentication

● Use SQL Server Authentication

User name: sa

Password: ••••

□ Save my password

Connect to a database

● Select or enter a database name: CPN

○ Attach a database file:

Logical name:

3. Enter the name of the SQL 2005 server that stores your TRAVERSE databases.

4. Select the **Use SQL Server Authentication** radio button and enter **sa** as the user name and the password.

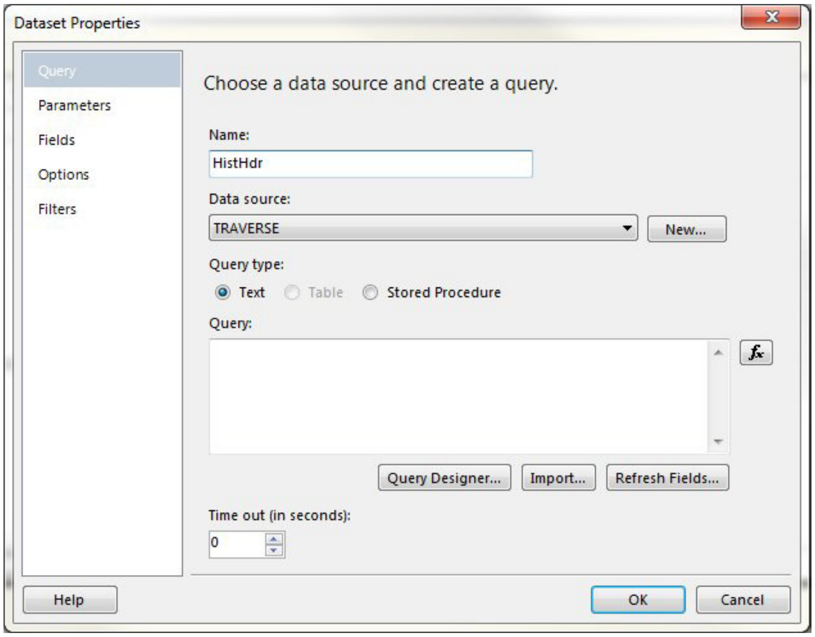
5. Select **CPU**, or your company database in the **Select or enter a database name** drop down box.

6. Click **OK** to take you back to the Data Source window.

7. Click **OK** to exit the Data Source window.

8. You will now see the **Dataset** dialog box.

9. Enter **HistHdr** for the dataset name and select **TRAVERSE** from the Data Source drop down list.



Dataset Properties

Choose a data source and create a query.

Name: HistHdr

Data source: TRAVERSE

Query type: ● Text ○ Table ○ Stored Procedure

Query:

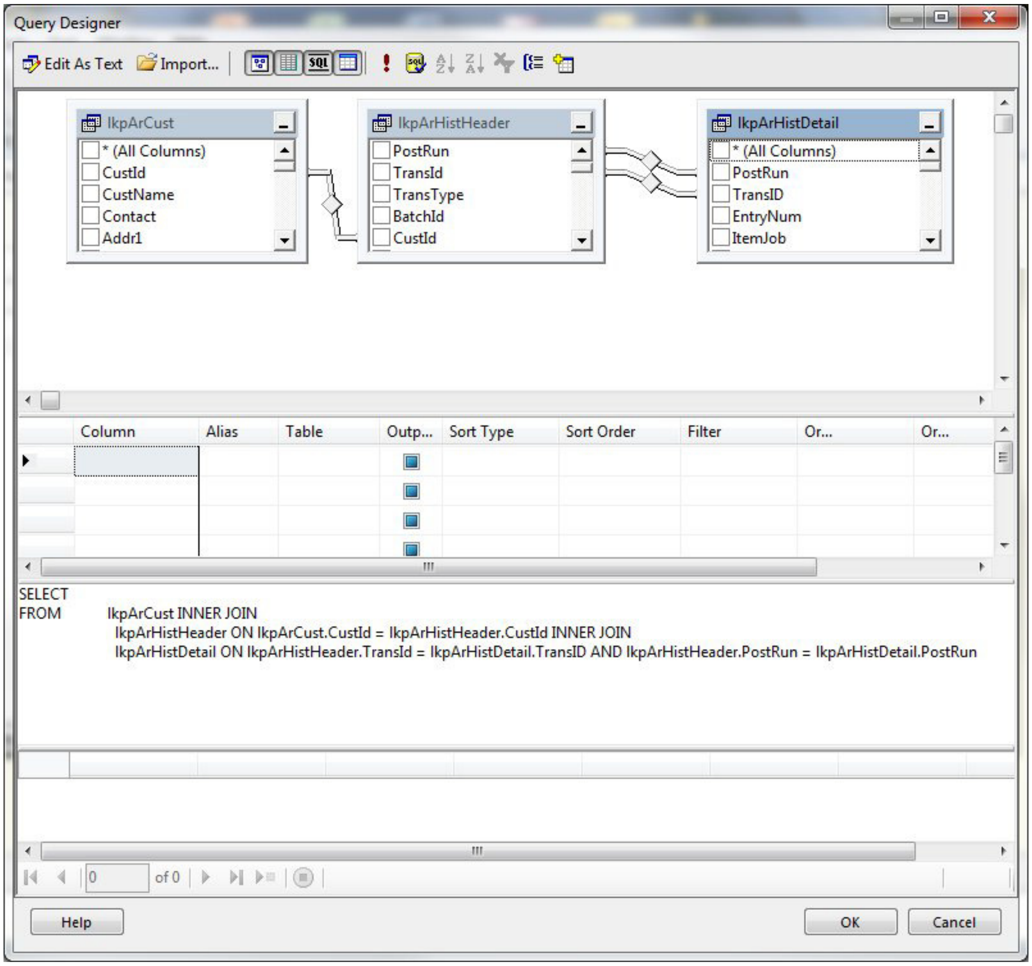
Time out (in seconds): 0

10. Click **Query Designer** to set up your query to bring in data.

11. Click on the Generic Query Designer Button to change to the Graphical Query Designer mode.

12. Click the **Add Table** button to bring up the table list.

13. Click on the **Views** tab and then scroll down and double click on the **lkpArCust**, **lkpArHistHeader** and **lkpArHistDetail** views to bring into the query designer window.



```
SELECT
FROM        lkpArCust INNER JOIN
             lkpArHistHeader ON lkpArCust.CustId = lkpArHistHeader.CustId INNER JOIN
             lkpArHistDetail ON lkpArHistHeader.TransId = lkpArHistDetail.TransID AND lkpArHistHeader.PostRun = lkpArHistDetail.PostRun
```

14. We need to create out joins between the views. Click the **CustId** in the **lkpArCust** view and drag it over to the CustId in the **lkpArHistHeader**.

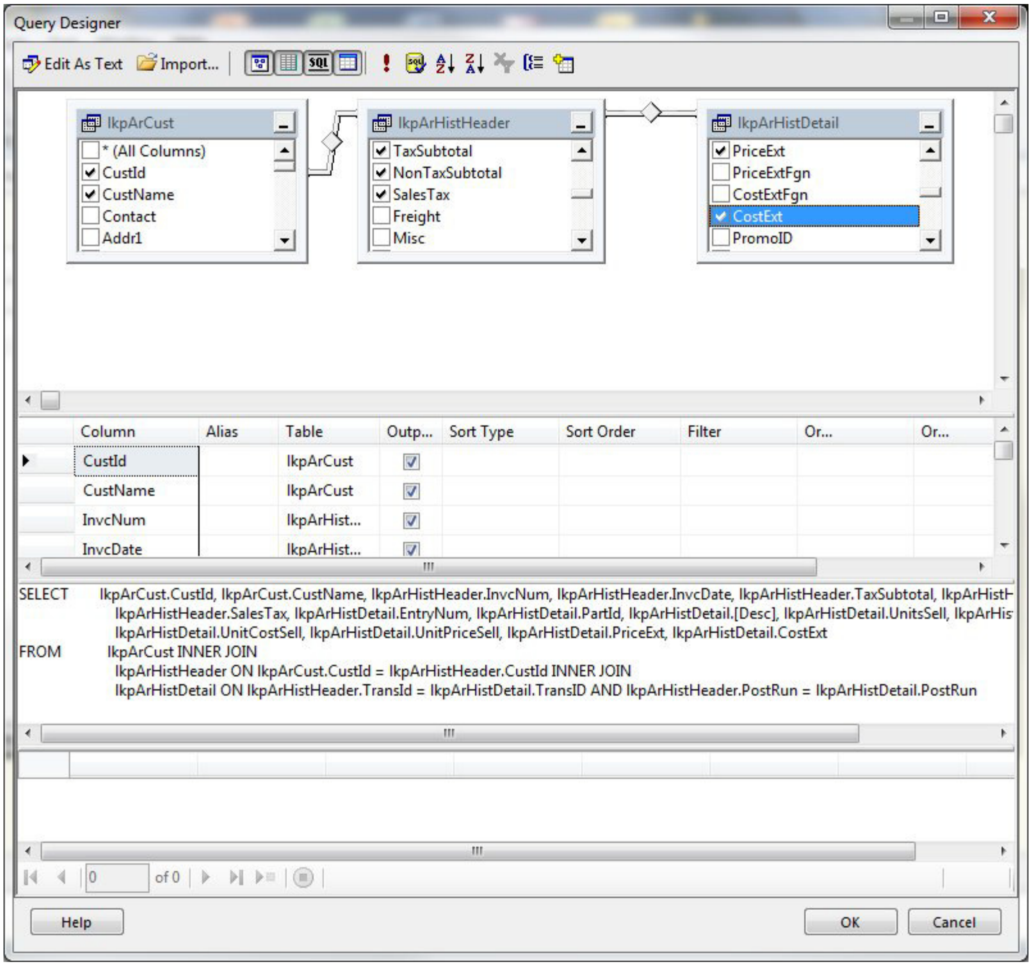
15. We also need to make two joins from the **lkpArHistHeader** to the **lkpArHistDetail**. Drag the **TransId** field from the **lkpArHistHeader** to the **TransId** in the **lkpArHistDetail** and **PostRun** field from the **lkpArHistHeader** to the **PostRun** in the **lkpArHistDetail**.

16. In the **lkpArCust** field list check the following fields: **CustId** and **CustName**.

17. In the **lkpArHistHeader** field list check the following fields: **InvcNum**, **InvcDate**, **TaxSubtotal**, **NonTaxSubtotal** and **SalesTax**.

18. In the **lkplArHistDetail** field list check the following fields: **EntryNum**, **PartId**, **Desc**, **UnitsSell**, **QtyShipSell**, **UnitPriceSell**, **UnitCostSell**, **PriceExt**, **CostExt**.

19. Your Query Designer should now look like this.



Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...	Or...
CustId		lkpArCust	☑					
CustName		lkpArCust	☑					
InvcNum		lkpArHist...	☑					
InvcDate		lkpArHist...	☑					

```
SELECT     lkpArCust.CustId, lkpArCust.CustName, lkpArHistHeader.InvcNum, lkpArHistHeader.InvcDate, lkpArHistHeader.TaxSubtotal, lkpArHistH
             lkpArHistHeader.SalesTax, lkpArHistDetail.EntryNum, lkpArHistDetail.PartId, lkpArHistDetail.[Desc], lkpArHistDetail.UnitsSell, lkpArHis
             lkpArHistDetail.UnitCostSell, lkpArHistDetail.UnitPriceSell, lkpArHistDetail.PriceExt, lkpArHistDetail.CostExt
FROM        lkpArCust INNER JOIN
             lkpArHistHeader ON lkpArCust.CustId = lkpArHistHeader.CustId INNER JOIN
             lkpArHistDetail ON lkpArHistHeader.TransId = lkpArHistDetail.TransID AND lkpArHistHeader.PostRun = lkpArHistDetail.PostRun
```

20. Click **OK** twice to save the data set.

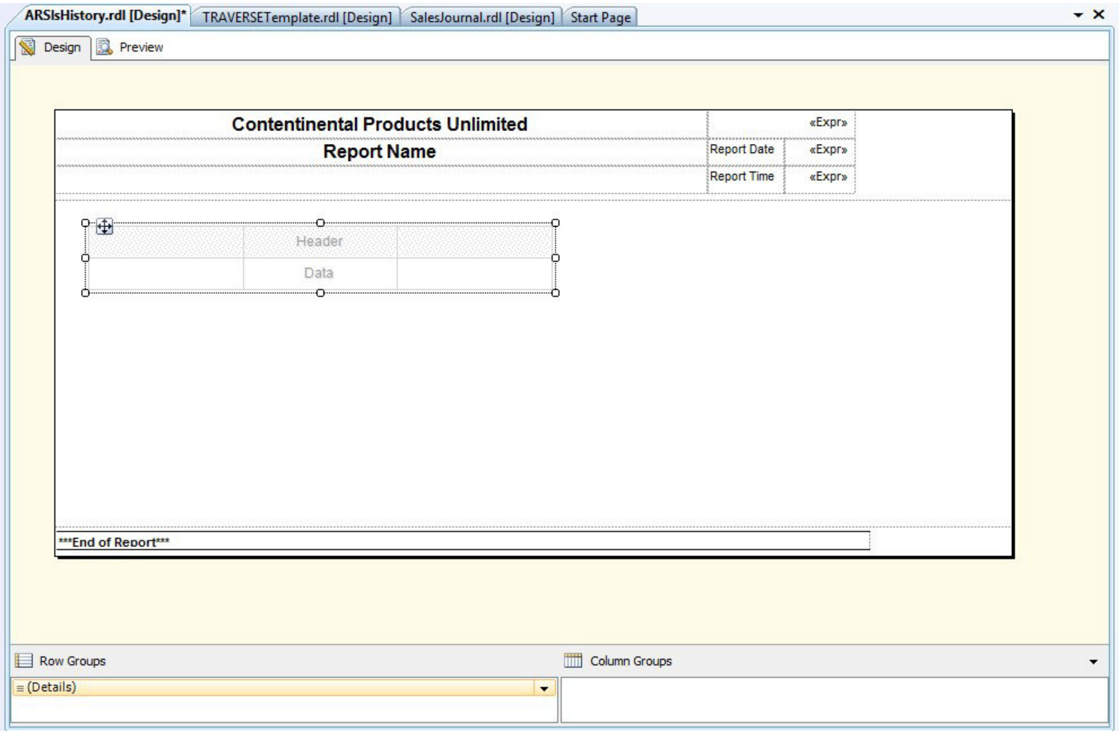
Task Notes:

For this report we needed to use 3 views to get all the information we wanted. You can see as we added each view the query designer did not automatically make joins between the views. We had to manually make the joins. The lkpArCust and lkpArHistHeader are joined by the CustId. The lkpArHistHeader and lkpArHistDetail are joined by the TransID. We must have these joins so when the report is rendered we will get the data we want in the correct groupings and format.

Sales History Report, Task 3: Place a Table Item on the report, Populate It and set up table groupings

1. Click the **Design** tab so we can now design our report.

2. From the **Toolbox** add a **Table** to the **Body** area of the report.



3. Select CustId from the list of fields when you click the Field List button in the first detail cell.

4.

5. On the left side of the screen you will see the Dataset area. Expand the dataset so you can see the fields we added.

6. In the Group 2 row select the **InvcNum** field from the list when you click on the list symbol in the upper right corner of the field in the first cell and **InvcDate** into the second cell.

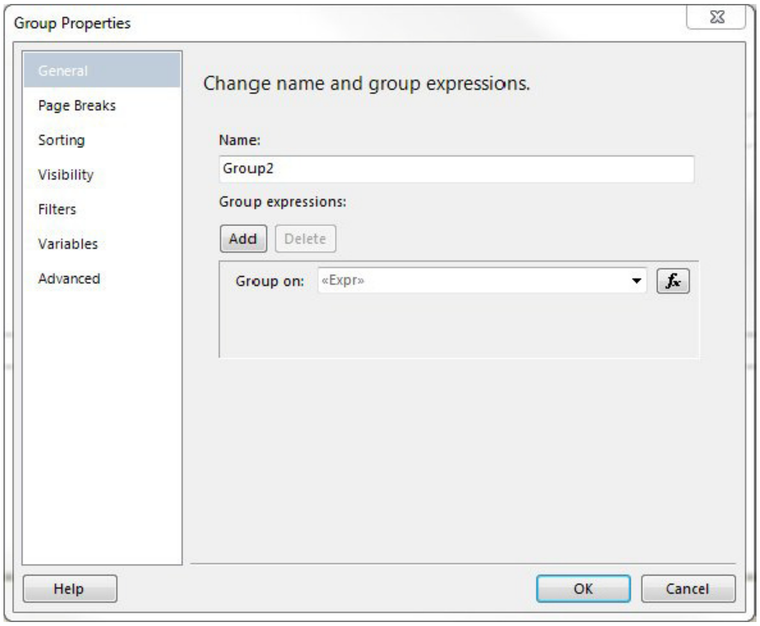
7. Select **EntryNum** in the Sort By field in the window that appears and click **OK**.

8. Right click on the Grouping 2 row and select **Add Group, Adjacent Below**.

9. Right click on the new group row and select **Row Group**, **Group Properties**. The **Grouping and Sorting Properties** dialog box appears.

10. In the Sort By field click on the Expression button (fx).

11. The Expression box appears. Select **Datasets**, then **First InvcNum**. In the expression field enter or edit the text to match `=First(Fields!InvcNum.Value, “HistHdr”) & “”`.



Group Properties

Change name and group expressions.

Name: Group2

Group expressions:

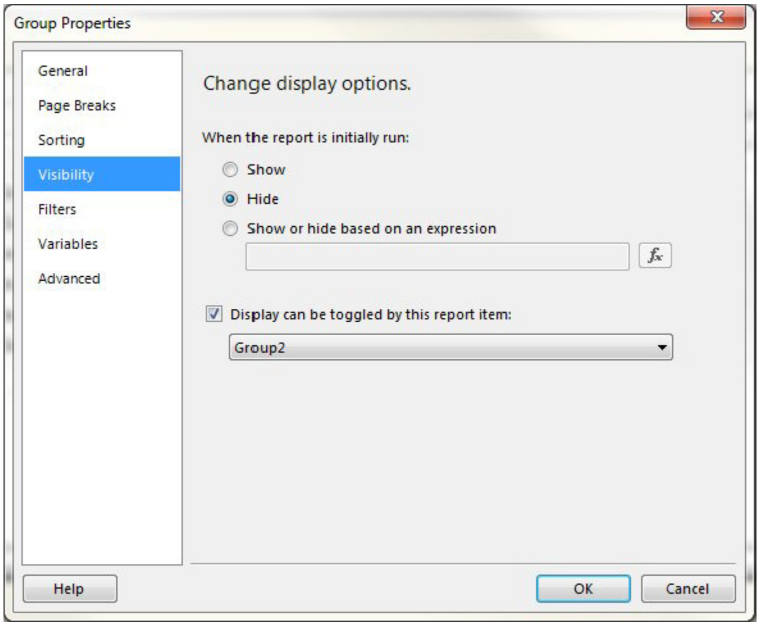
Group on: «Expr»

12. On the **Sorting** selection select **[EntryNum]** from the Sort By list and **A to Z** for the Order.

13. Go to the **Visibility** selection.

14. Select **Hide** from the choices under **When the report is initially run:**.

15. Check the box for **Display can be toggled this report item** and select **InvcNum** from the **Report Item** list.



Change display options.

When the report is initially run:

○ Show

● Hide

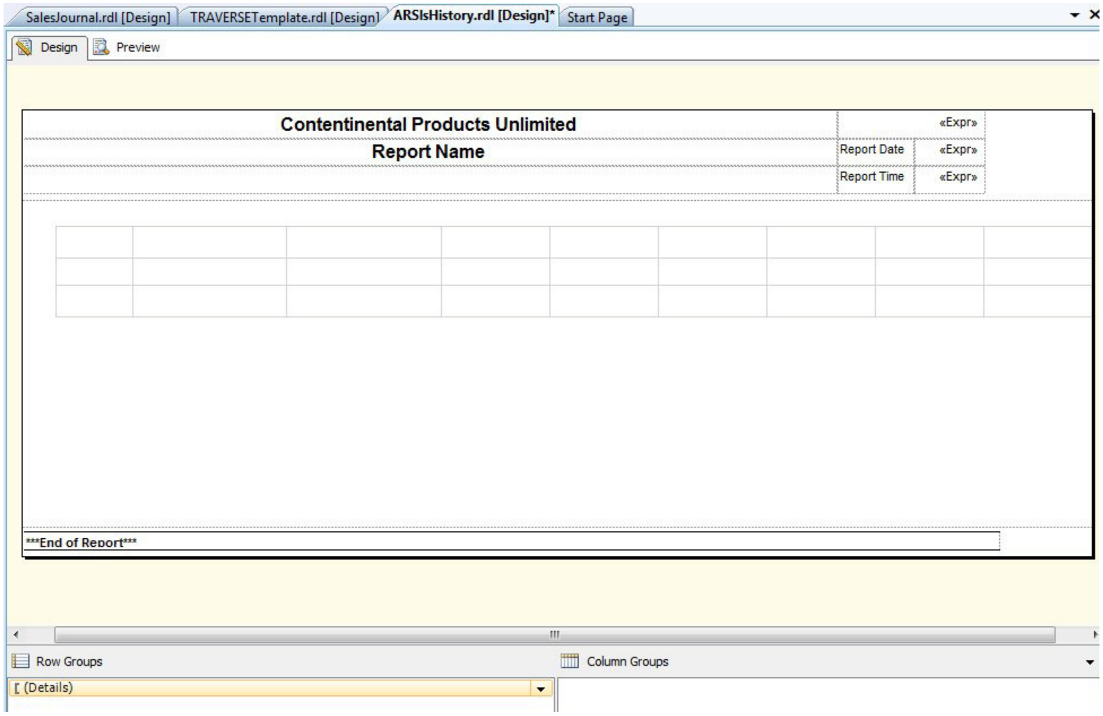
○ Show or hide based on an expression

☑ Display can be toggled by this report item:

Group2

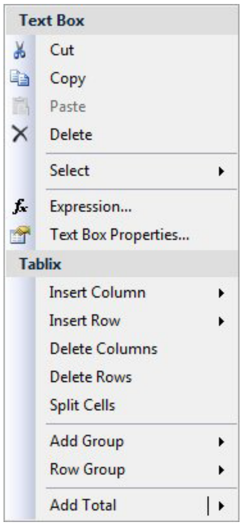
16. Click **OK** to close the Grouping and Sorting Properties box and return to the report layout.

17. Repeat this until we have **9** columns in the table. Size the columns so you can see them in the Layout window.

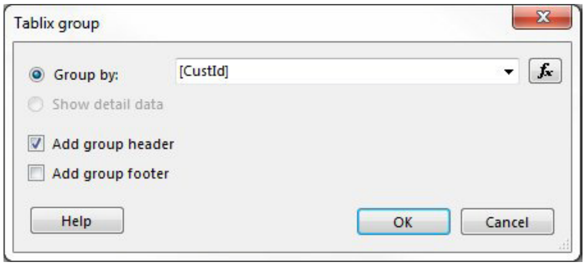


18. Click somewhere in the table to select it. Select **CustID** from the field list button in the first detail cell of the table.

19. Right click in the first cell and select **Add Group**, **Parent Group** from the menu.



20. The **Tablix group** box will appear. Select **CustID** from the **Group By** list. check the **Add group header** box and click **OK**.



Tablix group

● Group by: [CustId]

○ Show detail data

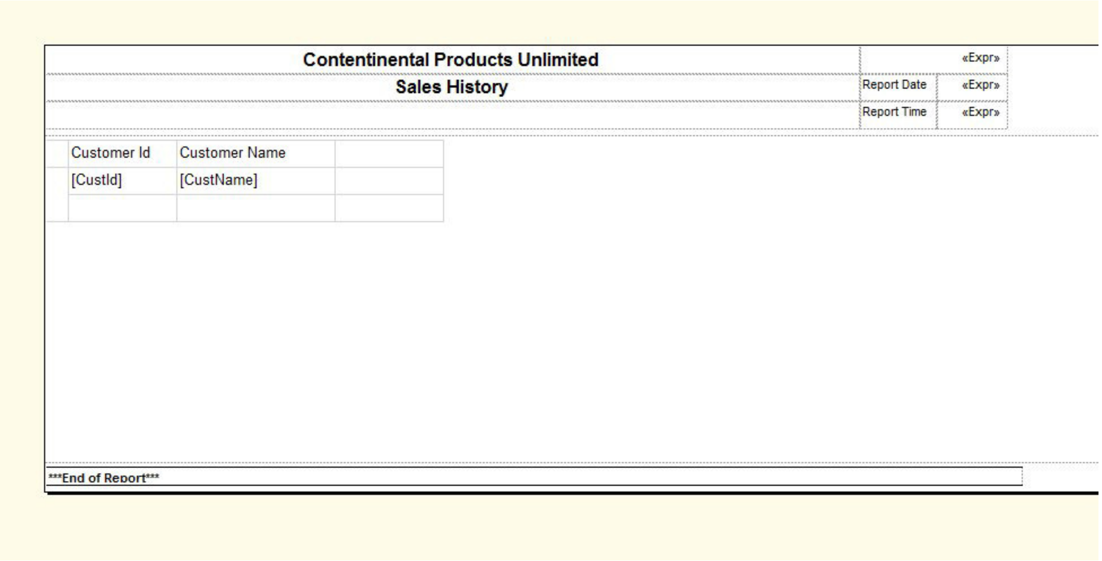
☑ Add group header

□ Add group footer

21. Select **CustId** from the list in the second column, second row and select **CustName** for the next cell.

22. Rename the headings that get filled in to **Customer ID** and **Customer Name**.

23. Delete the words **Group1** from the header row and resize the column to make it smaller.



Contentinental Products Unlimited

Sales History

	«Expr»
Report Date	«Expr»
Report Time	«Expr»

	Customer Id	Customer Name	
	[CustId]	[CustName]	

End of Report

24. Right click in the first cell in the table and select **Add Group**, **Child Group**.

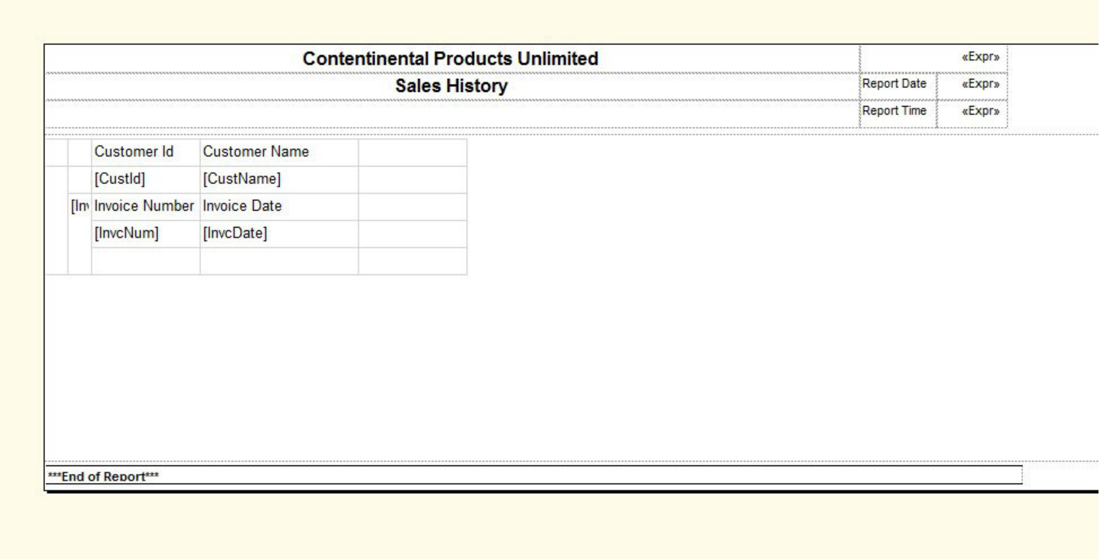
25. The **Tablix group** box will appear. Select **InvcNum** from the **Group By** list. check the **Add group header** box and click **OK**.

26. Select **InvcNum** from the list in the third column, third row and select **InvcDate** for the next cell.

27. Delete the word **Group2** from the top row and resize the column to be small like the first column.

28. Right click on the grey box next to the **InvcNum** group field and select **Insert Row, Above**.

29. Enter **Invoice Number** and **Invoice Date** in the new row above the **InvcNum** and **InvcDate** fields.



Contentinental Products Unlimited «Expr»

Sales History Report Date «Expr»

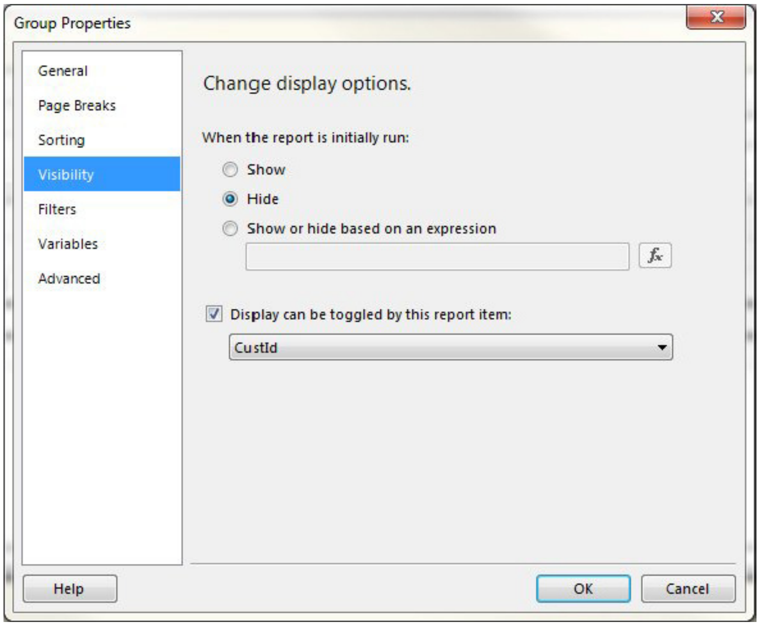
Report Time «Expr»

		Customer Id	Customer Name	
		[CustId]	[CustName]	
	[In	Invoice Number	Invoice Date	
		[InvcNum]	[InvcDate]	

End of Report

30. Right click in the Group2 cell under the heading you deleted and select **Row Group, Group Properties**.

31. Go to the **Visibility** selection.



Change display options.

When the report is initially run:

○ Show

● Hide

○ Show or hide based on an expression

☑ Display can be toggled by this report item:

CustId

32. Select **Hide** from the choices under **When the report is initially run:**.

33. Check the box for **Display can be toggled this report item** and select **Group1** from the **Report Item** list.

34. Click **OK** to save the changes and go back to our report layout.

35. Right click in the second cell in the table (the one for the InvcNum group) and select **Add Group**, **Child Group**.

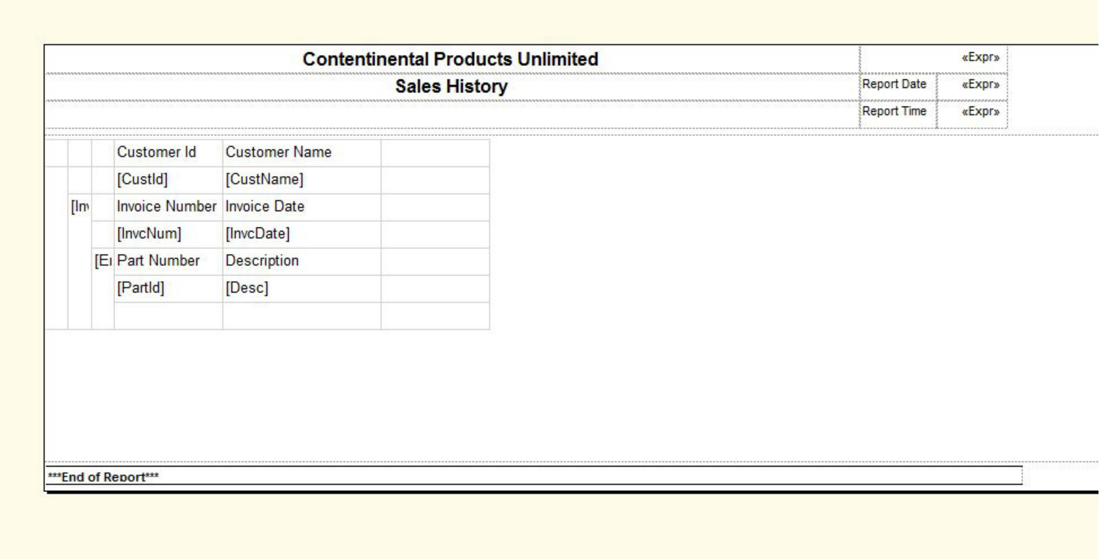
36. The **Tablix group** box will appear. Select **EntryNum** from the **Group By** list. check the **Add group header** box and click **OK**.

37. Select **PartId** from the list in the forth column, fifth row and select **Desc** for the next cell.

38. Delete the word **Group3** from the top row and resize the column to be small like the first column.

39. Right click on the grey box next to the **EntryNum**group field and select **Insert Row, Above**.

40. Enter **Part Number** and **Description** in the new row above the **PartId** and **Desc** fields.



Contentinental Products Unlimited «Expr»

Sales History

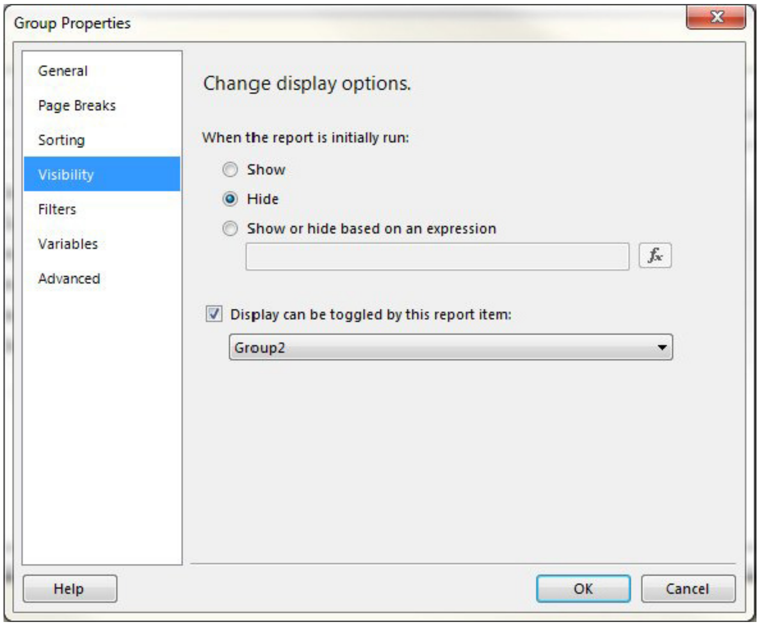
Report Date	«Expr»
Report Time	«Expr»

		Customer Id	Customer Name
		[CustId]	[CustName]
[Im		Invoice Number	Invoice Date
		[InvcNum]	[InvcDate]
	[Ei	Part Number	Description
		[PartId]	[Desc]

End of Report

41. Right click in the Group3 cell under the heading you deleted and select **Row Group, Group Properties**.

42. Go to the **Visibility** selection.



Change display options.

When the report is initially run:

○ Show
● Hide
○ Show or hide based on an expression

☑ Display can be toggled by this report item:

Group2

43. Select **Hide** from the choices under **When the report is initially run:**.

44. Check the box for **Display can be toggled this report item** and select **Group2** from the **Report Item** list.

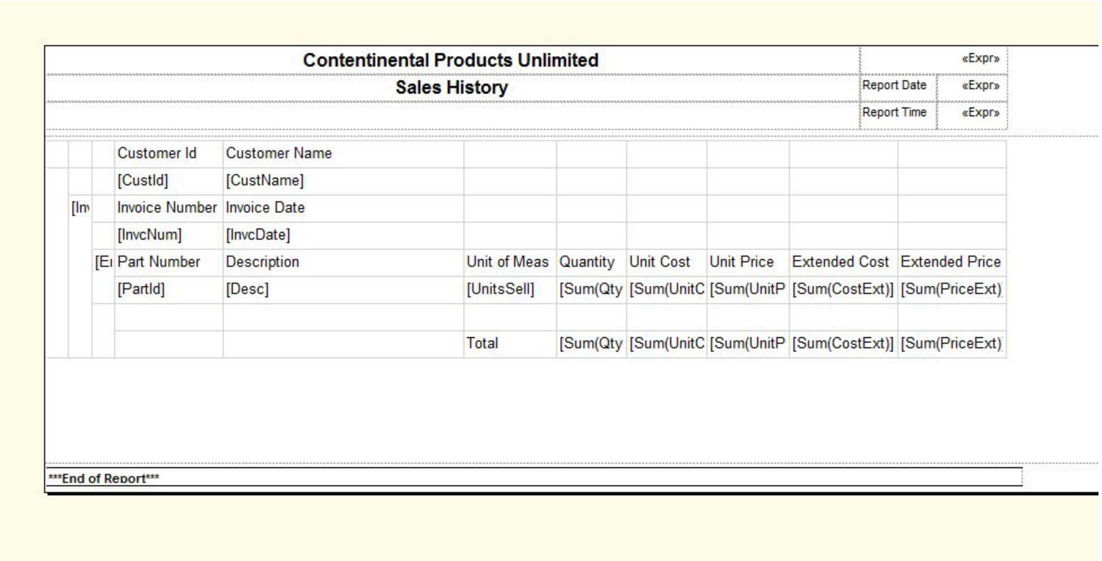
45. Click **OK** to save the changes and go back to our report layout.

46. In the PartId row select the **UnitsSell** for the last cell. Delete the title that gets put into the top cell and enter **Unit of Meas** into the Part Number row.

47. Right click the column header (grey box) of the last column and select Insert Column, Right. Select **QtyShipSell** in the PartId row. Delete the title that gets put into the top cell and enter **Quantity** into the Part Number (Heading) row.

48. Follow the steps in number 47 above for the following fields order: **UnitsSell**, **UnitCostSell**, **UnitPriceSell**, **CostExt** and **PriceExt**.

49. In the heading row type the following into each cell in order: **Unit Cost**, **Unit Price**, **Extended Cost** and **Extended Price.**



Contentinental Products Unlimited «Expr»

Sales History Report Date «Expr»

Report Time «Expr»

			Customer Id	Customer Name						
			[CustId]	[CustName]						
	[In		Invoice Number	Invoice Date						
			[InvcNum]	[InvcDate]						
		[E	Part Number	Description	Unit of Meas	Quantity	Unit Cost	Unit Price	Extended Cost	Extended Price
			[PartId]	[Desc]	[UnitsSell]	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)
					Total	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)

End of Report

50. Right click in the **Invoice Number** labeled field and select Text Box Properties. The Text Box Properties window appears.

51. Go to the **Visibility** selection.

52. Select **Hide** from the choices under **When the report is initially run:**.

53. Check the box for **Display can be toggled by this report item** and select **Group1** from the **Report Item** list.

54. Click **OK** to save the changes and go back to our report layout.

55. Follow the above steps for the **Invoice Date** heading field.

56. Follow the above steps for the, **Part Number**, **Description**, **Unit of Meas**, **Quantity**, **Unit Cost**, **Unit Price**, **Extended Cost** and **Extended Price** fields, selecting **Group2** for the **Display can be toggled by this report item**.

57. Follow these steps also for the **UnitsSell**, **UnitCostSell**, **UnitPriceSell**, **CostExt** and **PriceExt** fields selecting Group2 again from the **Display can be toggled by this report item.**

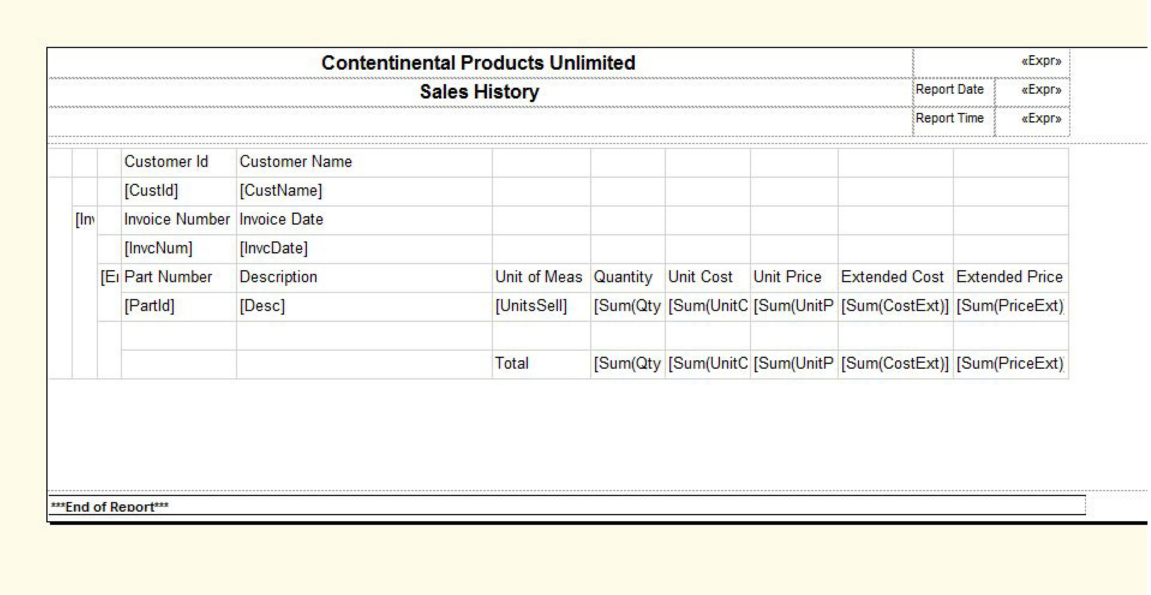
58. For the header fields for groups 2 and 3 in the properties, select **HistHdr** for the **Hide Duplicates** selection.

59. Delete the last row in the table. This is the Details row, which has nothing in it.

60. In the Row Groups section select Add Total After from the menu displayed when clicking on the arrow.

61. Enter the word Total in the total row added under UnitsSell.

62. Your report layout should now look similar to this.



Contentinental Products Unlimited «Expr»

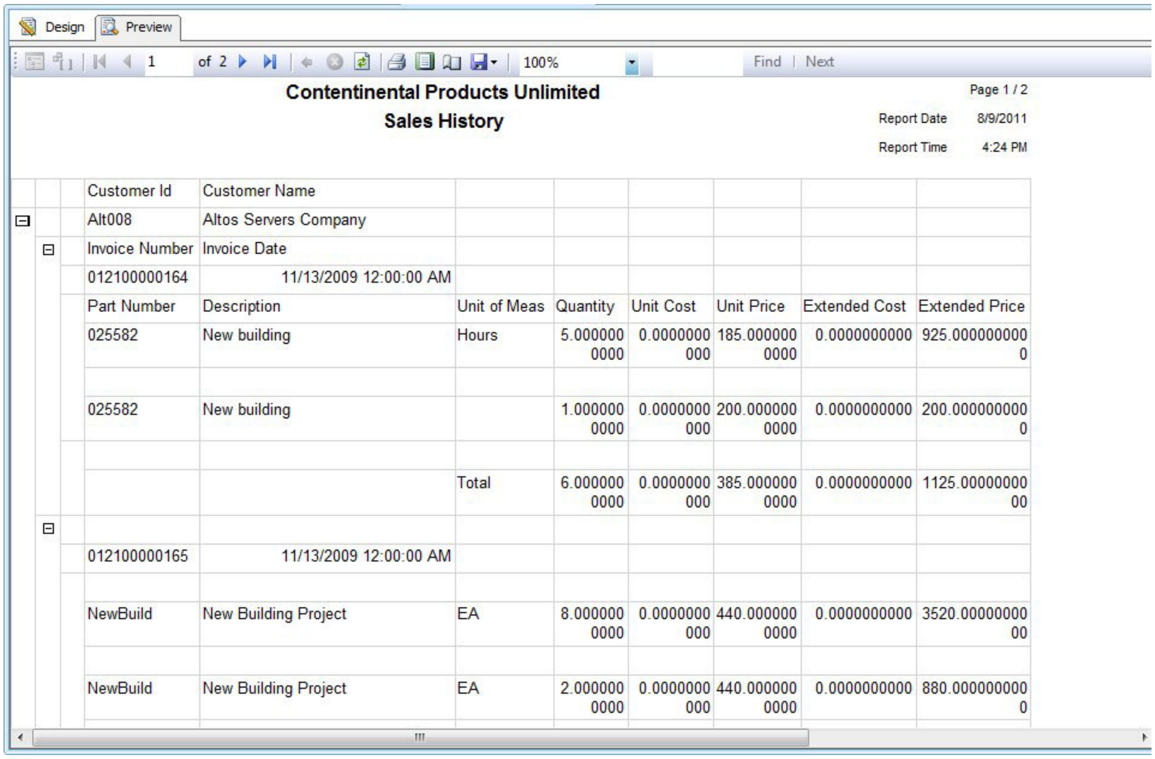
Sales History Report Date «Expr»

Report Time «Expr»

			Customer Id	Customer Name						
			[CustId]	[CustName]						
	[Inv		Invoice Number	Invoice Date						
			[InvcNum]	[InvcDate]						
		[E	Part Number	Description	Unit of Meas	Quantity	Unit Cost	Unit Price	Extended Cost	Extended Price
			[PartId]	[Desc]	[UnitsSell]	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)
					Total	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)

End of Report

63. Select the **Preview** tab and the report will render and should look similar to this if you expand the Customer ID and Invoice Number fields.



Contentinental Products Unlimited

Sales History

Page 1 / 2

Report Date 8/9/2011

Report Time 4:24 PM

	Customer Id	Customer Name						
⊟	Alt008	Altos Servers Company						
⊟	Invoice Number	Invoice Date						
	012100000164	11/13/2009 12:00:00 AM						
	Part Number	Description	Unit of Meas	Quantity	Unit Cost	Unit Price	Extended Cost	Extended Price
	025582	New building	Hours	5.0000000000	0.0000000000	185.0000000000	0.0000000000	925.0000000000
	025582	New building		1.0000000000	0.0000000000	200.0000000000	0.0000000000	200.0000000000
			Total	6.0000000000	0.0000000000	385.0000000000	0.0000000000	1125.0000000000
⊟								
	012100000165	11/13/2009 12:00:00 AM						
	NewBuild	New Building Project	EA	8.0000000000	0.0000000000	440.0000000000	0.0000000000	3520.0000000000
	NewBuild	New Building Project	EA	2.0000000000	0.0000000000	440.0000000000	0.0000000000	880.0000000000

NOTE: We have now designed a report from a template with multiple drill downs. From our results we can see that we have some final formatting to do to make the report look more to our liking.

Task Notes:

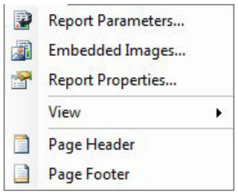
In this task we added a table to the report and added headings and groupings to the table. We used the groupings to be able to drill down from the very summarized list of customer Ids that have history invoices, to see the invoices and then to see the items in the invoices. We used only one group footer to get the totals for our extended costs and price.

Sales History Report, Task 4: Final Report formatting

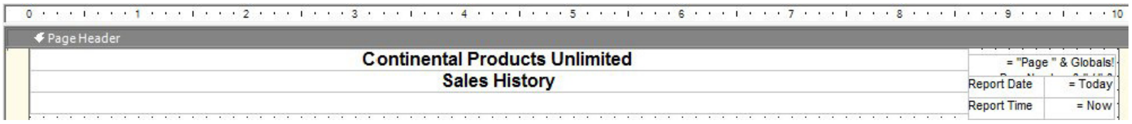
We now want our report to be more pleasing to the eye. We can see from the preview that our dates and numbers are not formatted correctly. Our report header and footer are not centered and our columns will extend beyond a standard portrait sheet of paper. Our column headings need to stand out more so they do not blend in with the rest of the report.

Follow these steps to get our report to have a more appealing look.

1. Go back to the **Layout** tab to view our report layout.
2. The first thing we will do is to get our page orientation correct so we can see all our data and not have some columns go to another page.
3. From the Main menu select **Report** and **Report Properties**.



4. Select the Layout tab and enter **11** for the **Page width** and **8.5** for the **Page height**.
5. Click **OK** to exit the Report Properties dialog box and go back to our report layout.
6. If the report width is larger than the 11 inches we formatted to. Move the scroll bar all the way to the right edge of the report. Move your mouse pointer to the edge of the white spotted area until it changes to a double arrow. Slide the edge of the report to the 10 in the scale at the top of the window.
7. We now need to format our report header to be centered on the page and have the page number and date in the correct place.
8. In the Report Header area we want to select the 5 text boxes in the upper right corner of the report header. To do this select the top box, hold your shift key and select each of the other boxes with our mouse.
9. With all those text boxes selected move your mouse pointer until it displays 4 arrows pointing in each direction and click your mouse and drag the text boxes so the right edge is under the 10 in the scale at the top and the top of the top text box is at the top of the report header window.
10. Select the 3 long heading text boxes using the same method as in step 8 and drag the right edge over to the left edge of the other smaller boxes. In the middle long box (Report Name), highlight the text and change it to say **Sales History**.
11. Your page header should now look similar to this.



Continental Products Unlimited

Sales History

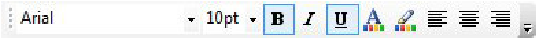
= "Page " & Globals!

Report Date = Today

Report Time = Now

12. Now we will move to the body of the report and format the column field headings to make them stand out in our report.

13. Select the top row in our table, in the text toolbar select the **B**old and **U**nderline symbols to make these column headings bolded and underlined.



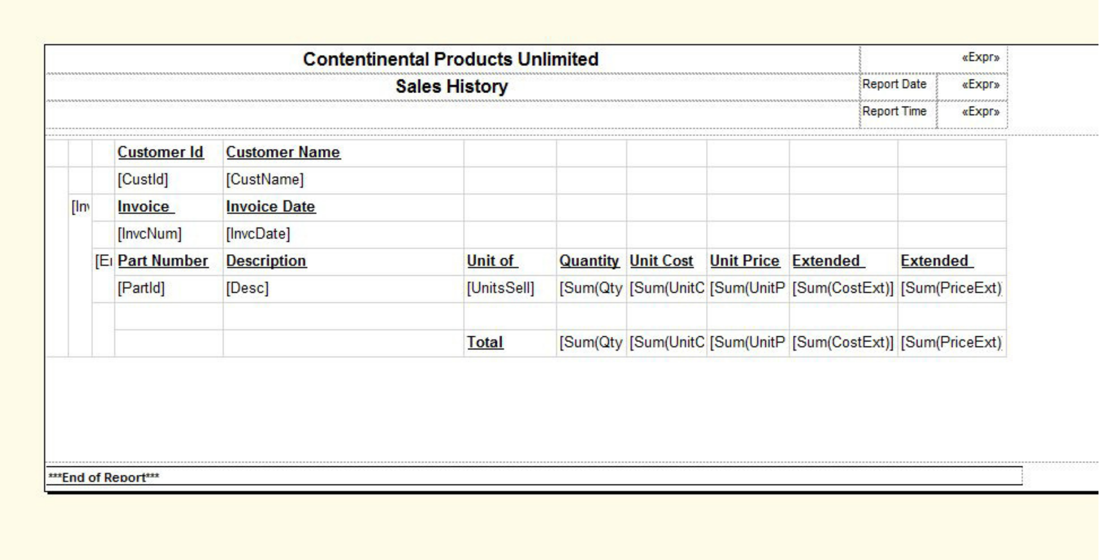
14. Select the Group 2 headings and select the **B**old and **U**nderline buttons in the text toolbar.

15. Select the Group 3headings and select the **B**old and **U**nderline buttons in the text toolbar.

16. Select the cell with the word Total in it and also select the **B**old and **U**nderline buttons in the text toolbar.

17. Resize the columns so they are better formatted to fit the data that will go into them. Make sure the last column does not extend beyond the 10 in the scale or the edge of our report design.

18. Your Body with the table in it should look similar to this.



Contentinental Products Unlimited «Expr»

Sales History Report Date «Expr»

Report Time «Expr»

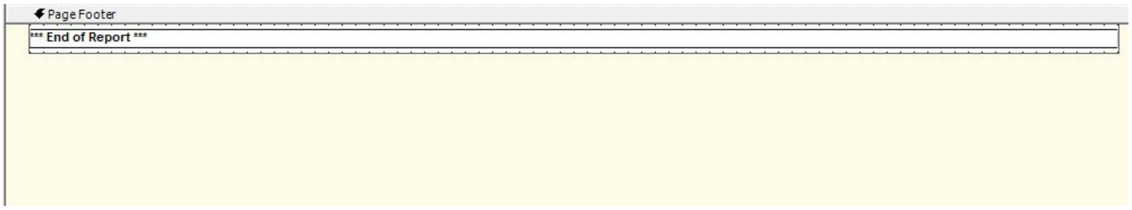
		Customer Id	**Customer Name**						
		[CustId]	[CustName]						
[In		**Invoice**	**Invoice Date**						
		[InvcNum]	[InvcDate]						
	[Ei	**Part Number**	**Description**	**Unit of**	**Quantity**	**Unit Cost**	**Unit Price**	**Extended**	**Extended**
		[PartId]	[Desc]	[UnitsSell]	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)
				Total	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)

End of Report

19. Our Page footer is also not formatted correctly.

20. Select the text box in the page footer and extend the box to the right edge of the report.

21. Your page footer should now look similar to this.

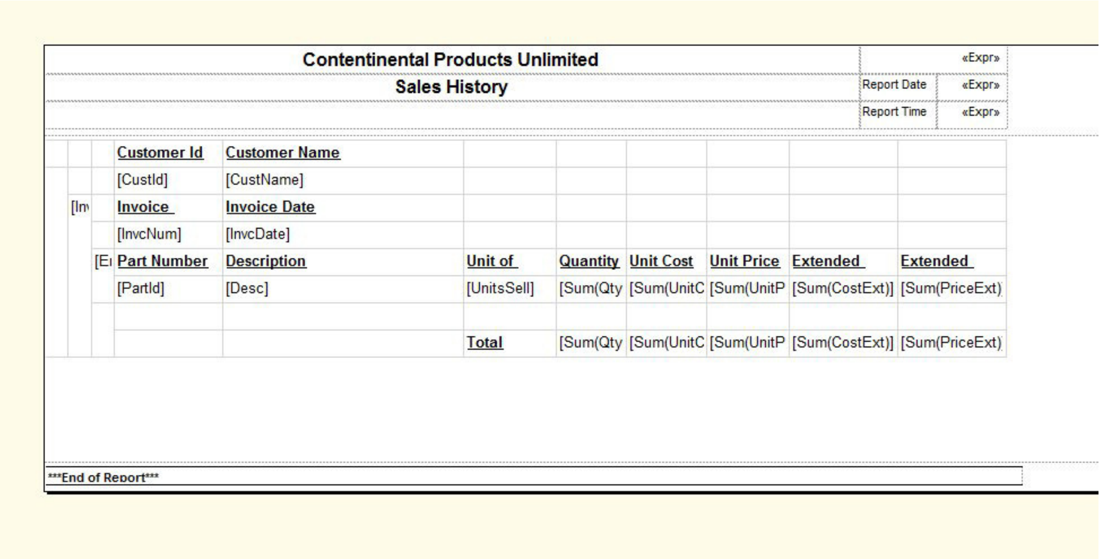


*** End of Report ***

22. In the Table in the Body area, select the 2 cells to the right of the cell with Total, these will be the bottom right 2 cells.

23. In the properties area find the **Border Style** selection and expand it. Select **Solid** for the property of the **Top** field.

24. Your whole report layout should now look similar to this.



Contentinental Products Unlimited «Expr»

Sales History Report Date «Expr»

Report Time «Expr»

			Customer Id	Customer Name						
			[CustId]	[CustName]						
	[In		Invoice	Invoice Date						
			[InvcNum]	[InvcDate]						
		[E	Part Number	Description	Unit of	Quantity	Unit Cost	Unit Price	Extended	Extended
			[PartId]	[Desc]	[UnitsSell]	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)
					Total	[Sum(Qty	[Sum(UnitC	[Sum(UnitP	[Sum(CostExt)]	[Sum(PriceExt)

End of Report

25. We have a couple final things to format.

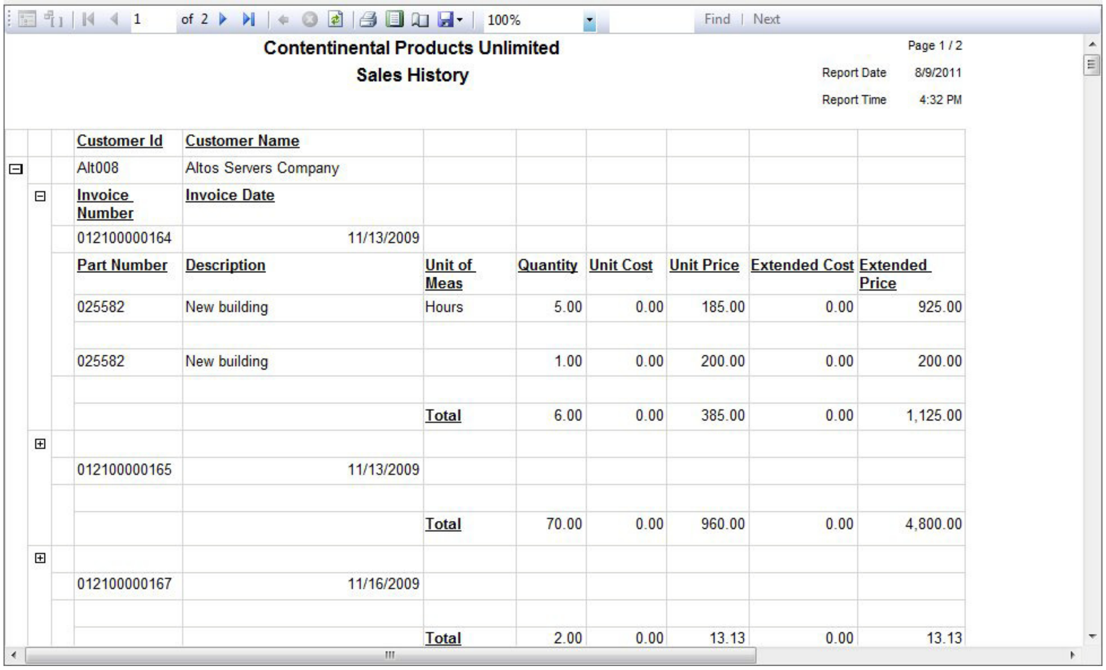
26. Remember when we previewed the report the Invoice Date and amounts fields did not have the correct formatting.

27. Select the cell with **InvcDate** in it. In the properties area in the lower right corner of the screen find the **Format** property and enter **d** into the field.

28. Now select the fields in the group 3 row with the costs and prices and in the properties area enter **n2** into the **Format** property.

29. Do this same thing for the 5 totals fields.

30. Click the **Preview** tab and expand the **Cust ID** and **Invoice Number** fields and your report should now look similar to this.



Contentinental Products Unlimited

Sales History

Page 1 / 2

Report Date 8/9/2011

Report Time 4:32 PM

		Customer Id	**Customer Name**						
⊟		Alt008	Altos Servers Company						
	⊟	**Invoice Number**	**Invoice Date**						
		012100000164	11/13/2009						
		Part Number	**Description**	**Unit of Meas**	**Quantity**	**Unit Cost**	**Unit Price**	**Extended Cost**	**Extended Price**
		025582	New building	Hours	5.00	0.00	185.00	0.00	925.00
		025582	New building		1.00	0.00	200.00	0.00	200.00
				Total	6.00	0.00	385.00	0.00	1,125.00
	⊞								
		012100000165	11/13/2009						
				Total	70.00	0.00	960.00	0.00	4,800.00
	⊞								
		012100000167	11/16/2009						
				Total	2.00	0.00	13.13	0.00	13.13

Task Notes:

When we formatted the fields in the page header, table and page footer we could have used the properties area to manually enter the positions and size of each text box, but it is much easier to select the fields and use the mouse pointer to move the fields to the position we want. We also could have used the properties area to set the table headings to have bold weight and underline, but again it is much simpler to just use the buttons in the font toolbar to set these font properties. We also could have set the column widths in the table using the properties areas, but again it is simpler to size the columns using the mouse pointer to drag the edges of the columns to give us the sizes we want.

When this report is rendered we get all our customers on the report. Maybe we only want to see a specific customer or a range of customers for a range of dates on our report. In the next section we will go through the steps to add selection criteria to the report to be able to limit the range of customers and invoice dates on our report.

PARAMETERIZED SALES HISTORY REPORT

From the users' standpoint, all our reports up to this point have been "what you see is what you get." These reports each ran a predetermined query to create the dataset. No user input was requested.

In the real world, this is not the way things work. Most reports require the user to specify some criteria that can help determine what information is ultimately in the report. The user may need to enter a start and an end date, or they may need to select the customer, sales region or department to be included in the report. Users like to have control over their reports, so they receive exactly the information they are looking for. Our next report enables you to get user input by using report parameters.

Feature Highlighted

- Using report parameters.

Business Need

The account department is pleased with the Sales History Report. Like most users, when they are happy with something, they want to change it. No software or report is ever really completed. It only reaches a resting point until users think of another enhancement.

The accounting department would like to be able to view the Sales History Report for one customer at a time, or a range of customers. They would also like to specify an invoice date from and an invoice date thru, and only view invoices that were entered between those dates.

We can modify the Sales History Report to include these features. We can add a WHERE clause to the SELECT statement that creates the dataset. Then we can send the user's selections for customer and invoice dates to the WHERE clause using report parameters.

Task Overview

1. Reopen the TRAVERSE Project, Open the Sales History Report and add parameters to the Query in the Original Dataset.
2. Create a Second Dataset Containing a list of Customers.
3. Customize the Report Parameters.

Parameterized Sales History Report, Task 1: Reopen the TRAVERSE Project, Open the Sales History Report and add parameters to the Query in the Original Dataset

1. If you closed the **TRAVERSE** project, reopen it.
2. If the Sales History Report is open, you are ready to go. If it is not open, double-click the entry for the **ARSalesHistory** report in the Solution Explorer on the right side of the screen.
3. Click the **Data** tab. You see the Query Designer screen with the query built for this report.
4. Select the **HistHdr** from the list of Datasets and right click, Query to view the graphical query designer.
5. Find the **CustId** record in the list and enter the following into the **Filter** field:

   BETWEEN ISNULL(@CustIDFrom, '') AND ISNULL(@CustIDThru, 'zzzzzzzzzz')

6. In the **InvcDate** row, enter the following into the Filter field:

   >= @StartDate

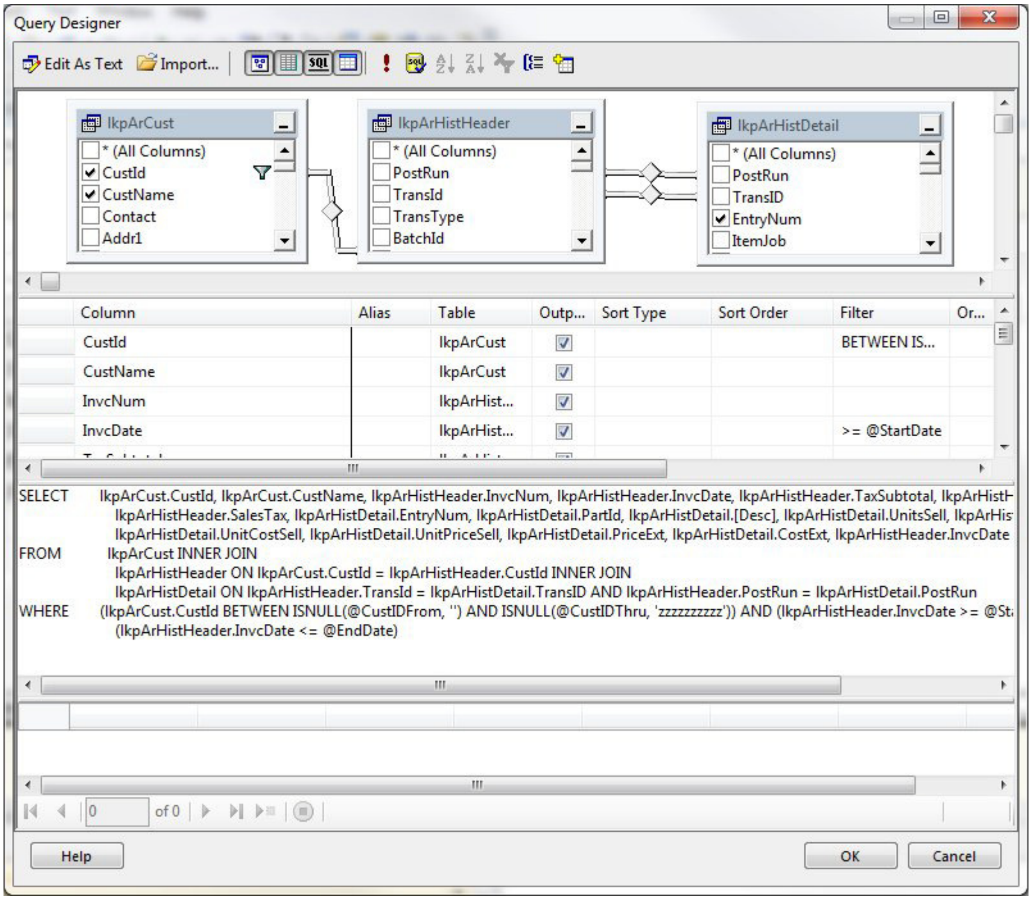
7. Scroll to the bottom of the list to add a new record and select **InvcDate** from the drop down list of fields.
8. Uncheck the Output box and enter the following into the Filter field:

   <= @EndDate

   Notice in the SQL area of the Query Designer the following WHERE clause has been added:

   WHERE (lkpArCust.CustId BETWEEN ISNULL(@CustIDFrom, '') AND ISNULL(@CustIDThru, 'zzzzzzzzzz')) AND (lkpArHistHeader.InvcDate >= @StartDate) AND (lkpArHistHeader.InvcDate <= @EndDate)

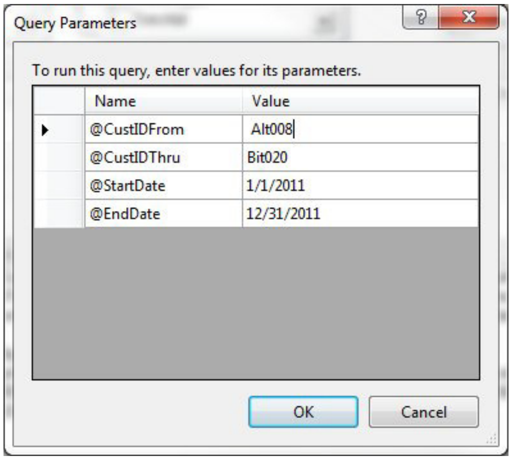
Instead of entering the text into the **Filter** fields in the field list area you could also type this WHERE clause into the SQL area of the screen.



Column	Alias	Table	Outp...	Sort Type	Sort Order	Filter	Or...
CustId		IkpArCust	☑			BETWEEN IS...	
CustName		IkpArCust	☑				
InvcNum		IkpArHist...	☑				
InvcDate		IkpArHist...	☑			>= @StartDate	

```
SELECT     IkpArCust.CustId, IkpArCust.CustName, IkpArHistHeader.InvcNum, IkpArHistHeader.InvcDate, IkpArHistHeader.TaxSubtotal, IkpArHistH
              IkpArHistHeader.SalesTax, IkpArHistDetail.EntryNum, IkpArHistDetail.PartId, IkpArHistDetail.[Desc], IkpArHistDetail.UnitsSell, IkpArHis
              IkpArHistDetail.UnitCostSell, IkpArHistDetail.UnitPriceSell, IkpArHistDetail.PriceExt, IkpArHistDetail.CostExt, IkpArHistHeader.InvcDate
FROM         IkpArCust INNER JOIN
              IkpArHistHeader ON IkpArCust.CustId = IkpArHistHeader.CustId INNER JOIN
              IkpArHistDetail ON IkpArHistHeader.TransId = IkpArHistDetail.TransID AND IkpArHistHeader.PostRun = IkpArHistDetail.PostRun
WHERE      (IkpArCust.CustId BETWEEN ISNULL(@CustIDFrom, '') AND ISNULL(@CustIDThru, 'zzzzzzzzz')) AND (IkpArHistHeader.InvcDate >= @St
              (IkpArHistHeader.InvcDate <= @EndDate)
```

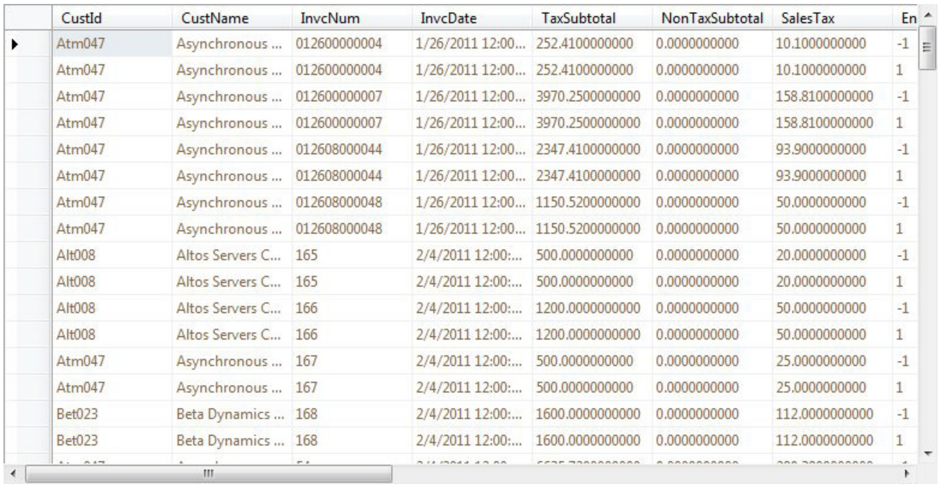
If you want to see the results you can click the Execute button ! and enter **Alt008** into the **CustId From** and **Bit020** for the **CustId Thru**. Enter **1/1/2011** for the **StartDate** and **12/31/2011** for the **EndDate**.



To run this query, enter values for its parameters.

Name	Value
@CustIDFrom	Alt008
@CustIDThru	Bit020
@StartDate	1/1/2011
@EndDate	12/31/2011

Below is an example of what your results might look like.



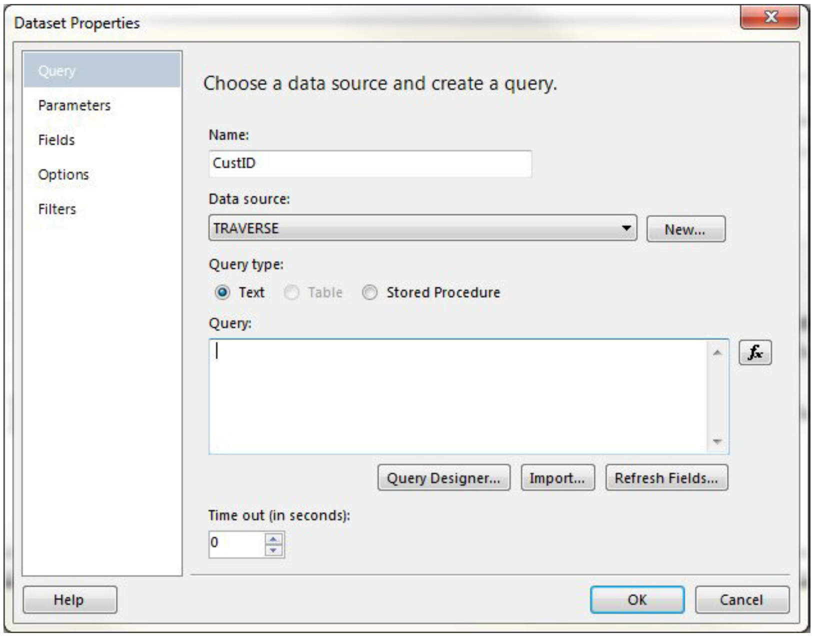
CustId	CustName	InvcNum	InvcDate	TaxSubtotal	NonTaxSubtotal	SalesTax	En
Atm047	Asynchronous ...	012600000004	1/26/2011 12:00...	252.4100000000	0.0000000000	10.1000000000	-1
Atm047	Asynchronous ...	012600000004	1/26/2011 12:00...	252.4100000000	0.0000000000	10.1000000000	1
Atm047	Asynchronous ...	012600000007	1/26/2011 12:00...	3970.2500000000	0.0000000000	158.8100000000	-1
Atm047	Asynchronous ...	012600000007	1/26/2011 12:00...	3970.2500000000	0.0000000000	158.8100000000	1
Atm047	Asynchronous ...	012608000044	1/26/2011 12:00...	2347.4100000000	0.0000000000	93.9000000000	-1
Atm047	Asynchronous ...	012608000044	1/26/2011 12:00...	2347.4100000000	0.0000000000	93.9000000000	1
Atm047	Asynchronous ...	012608000048	1/26/2011 12:00...	1150.5200000000	0.0000000000	50.0000000000	-1
Atm047	Asynchronous ...	012608000048	1/26/2011 12:00...	1150.5200000000	0.0000000000	50.0000000000	1
Alt008	Altos Servers C...	165	2/4/2011 12:00:...	500.0000000000	0.0000000000	20.0000000000	-1
Alt008	Altos Servers C...	165	2/4/2011 12:00:...	500.0000000000	0.0000000000	20.0000000000	1
Alt008	Altos Servers C...	166	2/4/2011 12:00:...	1200.0000000000	0.0000000000	50.0000000000	-1
Alt008	Altos Servers C...	166	2/4/2011 12:00:...	1200.0000000000	0.0000000000	50.0000000000	1
Atm047	Asynchronous ...	167	2/4/2011 12:00:...	500.0000000000	0.0000000000	25.0000000000	-1
Atm047	Asynchronous ...	167	2/4/2011 12:00:...	500.0000000000	0.0000000000	25.0000000000	1
Bet023	Beta Dynamics ...	168	2/4/2011 12:00:...	1600.0000000000	0.0000000000	112.0000000000	-1
Bet023	Beta Dynamics ...	168	2/4/2011 12:00:...	1600.0000000000	0.0000000000	112.0000000000	1

Task Notes:

We have now added two parameters to the WHERE clause of the SELECT statement. Only rows where the CustId is equal to and between the from and thru of the @CustId will be displayed in the result set. If we leave the CustId from and CustIdThru fields empty we will get all customers. When we enter dates into the @StartDate and @EndDate we will only get records with invoice dates between these dates.

Parameterized Sales History Report, Task 2: Create a Second Dataset Containing a list of Customers

1. Select **New Dataset** from the drop down lost at the top of the Report Data window.



Choose a data source and create a query.

Name: CustID

Data source: TRAVERSE

Query type: ● Text ○ Table ○ Stored Procedure

Query:

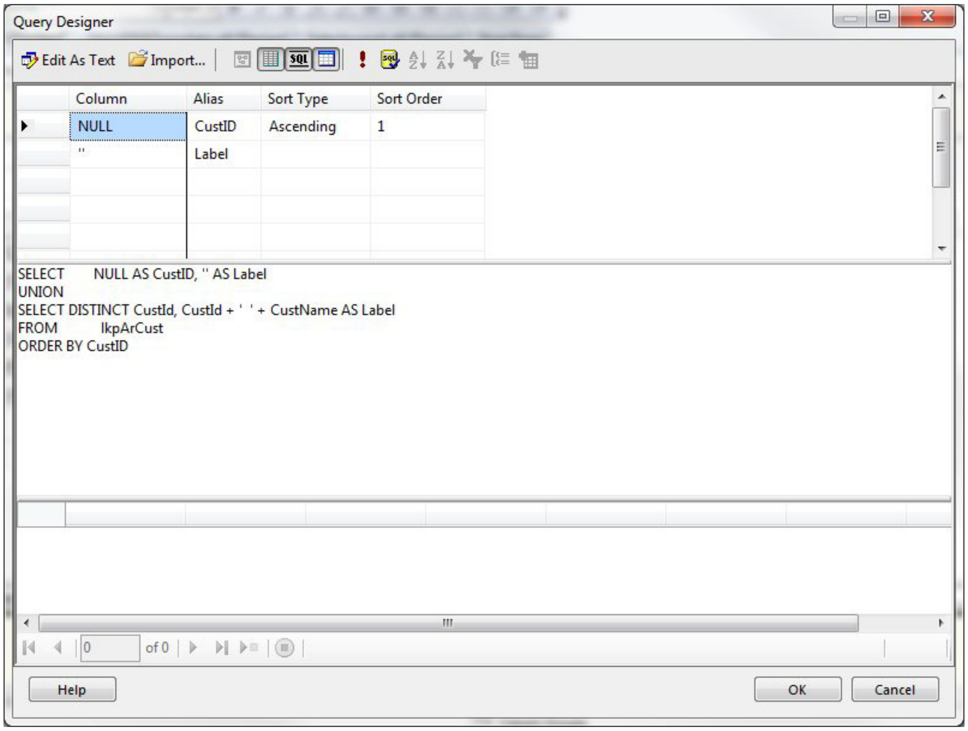
Time out (in seconds): 0

2. Enter **CustID** as the Name, select **TRAVERSE** as the **Data Source** and type the following into the query window:

```
SELECT     NULL AS CustID, '' AS Label

UNION

SELECT DISTINCT CustID, CustID+' '+CustName AS Label

FROM lkpArCust

ORDER BY CustID
```

3. Click OK to return to the report design screen.

This dataset will give us a combo box list of customers with the CustId and CustName displayed in the combo box.



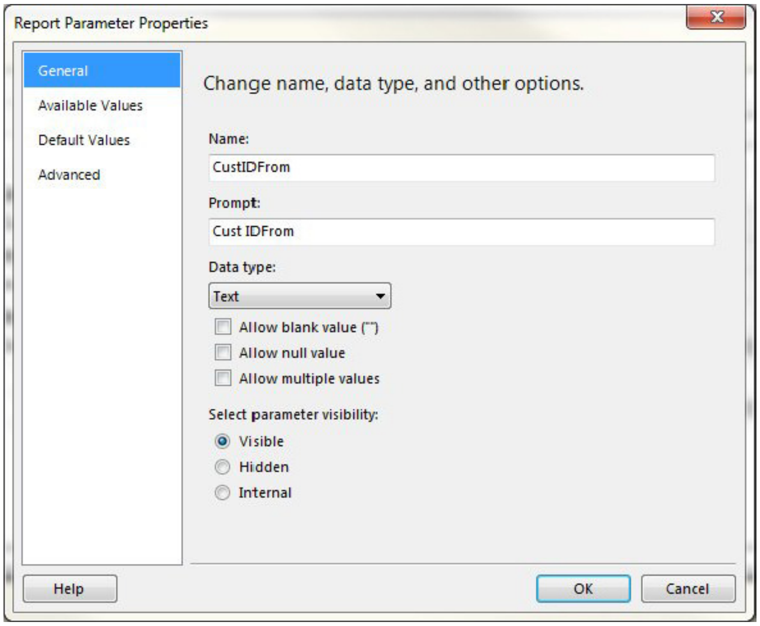
Column	Alias	Sort Type	Sort Order
NULL	CustID	Ascending	1
''	Label		

```
SELECT      NULL AS CustID, '' AS Label
UNION
SELECT DISTINCT CustId, CustId + '  ' + CustName AS Label
FROM        lkpArCust
ORDER BY CustID
```

Parameterized Sales History Report, Task 3: Customize the Report Parameters

1. In the **Report Data** area of the screen expand the **Parameters** selection.

2. Right click the **CustIdFrom** parameter in the Parameters area and select Parameter Properties. The Report Parameter Properties window will appear.



Change name, data type, and other options.

Name: CustIDFrom

Prompt: Cust IDFrom

Data type: Text

□ Allow blank value ("")
□ Allow null value
□ Allow multiple values

Select parameter visibility:
● Visible
○ Hidden
○ Internal

3. Enter **Customer ID From** into the **Prompt** field in the Properties area.

4. Check the **Allow null value** and **Allow blank value** check boxes.

5. Select **Available Values** from the list on the left. Select **Get values from a query**, **CustID** for the **Dataset**, **CustID** for the **Value field** and **Label** for the **Label field** in the Available values area.

6. Select **Null** for the **Default values**. Your screen should look like the one above.

7. Click **OK**.

8. Right click the **CustIDThru** parameter in the Parameters area and select Parameter Properties.

9. Enter **Customer ID Thru** into the **Prompt** field in the Properties area.

10. Select **Available Values** from the list on the left. Select **Get values from a query**, **CustID** for the **Dataset**, **CustID** for the **Value field** and **Label** for the **Label field** in the Available values area.

11. Click **OK**.

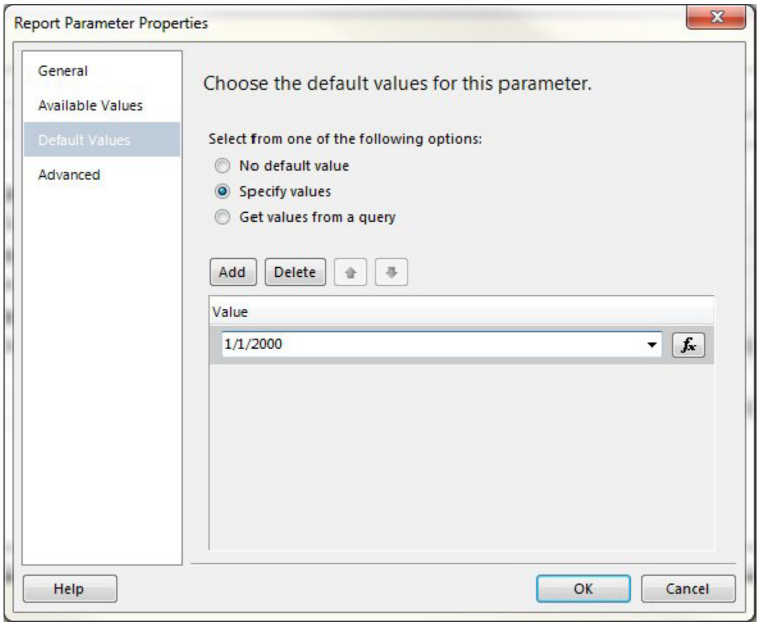
12. Right click the **StartDate** parameter in the Parameters area and select Parameter Properties.

13. Enter **Start Date** into the **Prompt** field in the Properties area.

14. Select **Data/Time** in the **Data Type** area.

15. Select **Default Values** from the list on the left. Select **Specify Values** for the **Select from one of the following options:**.

16. Select **Add** and enter **1/1/2000**, or a date you know is older than your earliest invoice history date.


Report Parameter Properties

Choose the default values for this parameter.

Select from one of the following options:

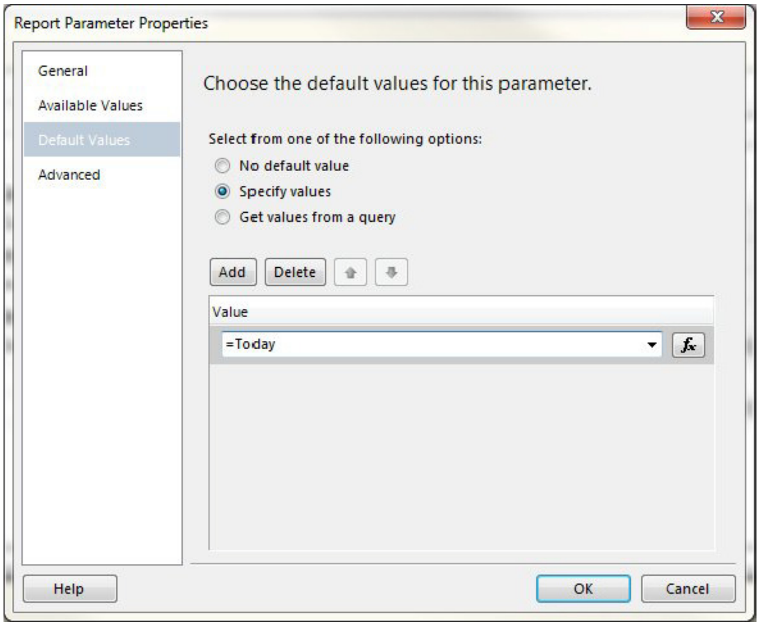
○ No default value
● Specify values
○ Get values from a query

Value: 1/1/2000

17. Right click the **StartDate** parameter in the Parameters area and select Parameter Properties. Select the **EndDate** parameter and enter **Invoice Date Thru** for the **Prompt**. Leave the **Allow null value** unchecked.

18. Select **None** for the **Available values** area.

19. Select **Specify values** for the **Default values** and enter **=Today**. This will default today's date into the invoice date thru field.



Report Parameter Properties

Choose the default values for this parameter.

Select from one of the following options:

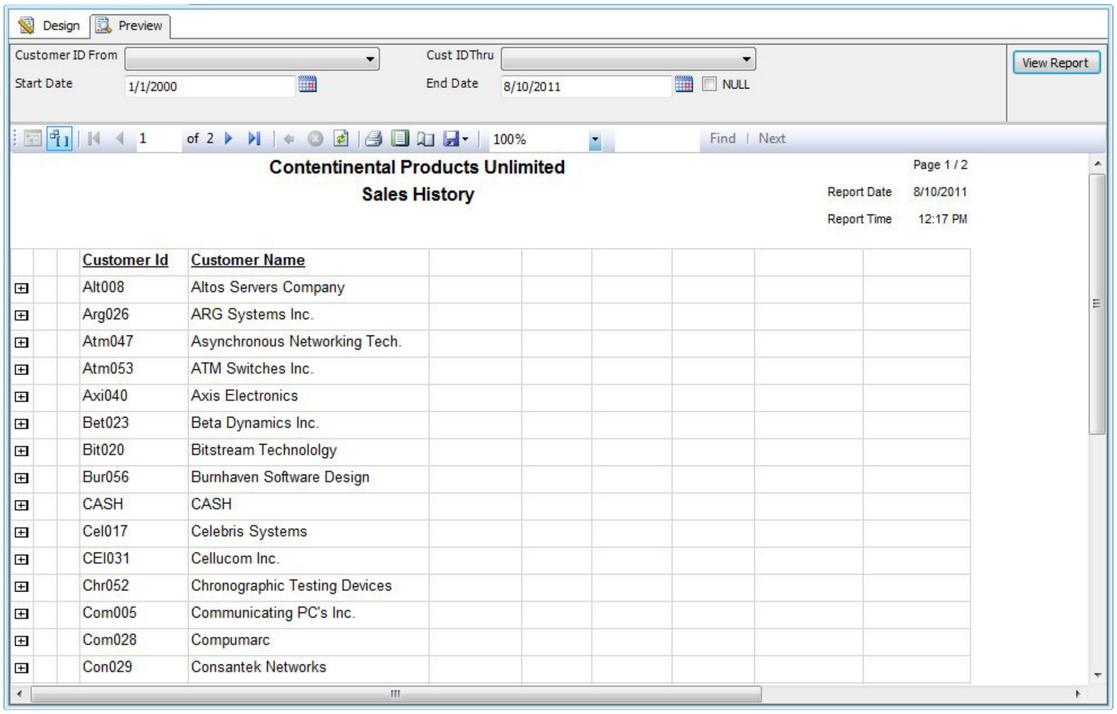
○ No default value
● Specify values
○ Get values from a query

Value

=Today

20. Click **OK** to exit the Report Parameters dialog box.

21. Click the **Preview** tab.



Customer ID From

Cust IDThru

Start Date 1/1/2000

End Date 8/10/2011 □ NULL

Contentinental Products Unlimited
Sales History

Page 1 / 2
Report Date 8/10/2011
Report Time 12:17 PM

Customer Id	Customer Name
Alt008	Altos Servers Company
Arg026	ARG Systems Inc.
Atm047	Asynchronous Networking Tech.
Atm053	ATM Switches Inc.
Axi040	Axis Electronics
Bet023	Beta Dynamics Inc.
Bit020	Bitstream Technololgy
Bur056	Burnhaven Software Design
CASH	CASH
Cel017	Celebris Systems
CEl031	Cellucom Inc.
Chr052	Chronographic Testing Devices
Com005	Communicating PC's Inc.
Com028	Compumarc
Con029	Consantek Networks

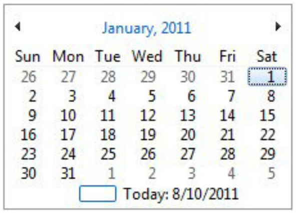
22. The prompts for the four report parameters appear at the top of the preview area. The report is rendered with all customers and dates. It does this because we told it to bring in all values from the first to last in the WHERE clause.

    (lkpArCust.CustId BETWEEN ISNULL(@CustIDFrom, '') AND ISNULL(@CustIDThru, 'zzzzzzzzzz’))

    The ‘“after the @CustIDFrom tells SQL to bring in the first possible value and the 'zzzzzzzzzz’ after the @CustIDThru tells SQL to bring in the last possible value.

23. Let’s select **Alt008** for the **Customer ID From** and **Bet023** for the **Customer ID Thru**.

24. Let’s also change the **Invoice Date From** to **1/1/2011**, notice there is a calendar symbol next to the date selection. If we click on this calendar symbol we will get a calendar to select from.



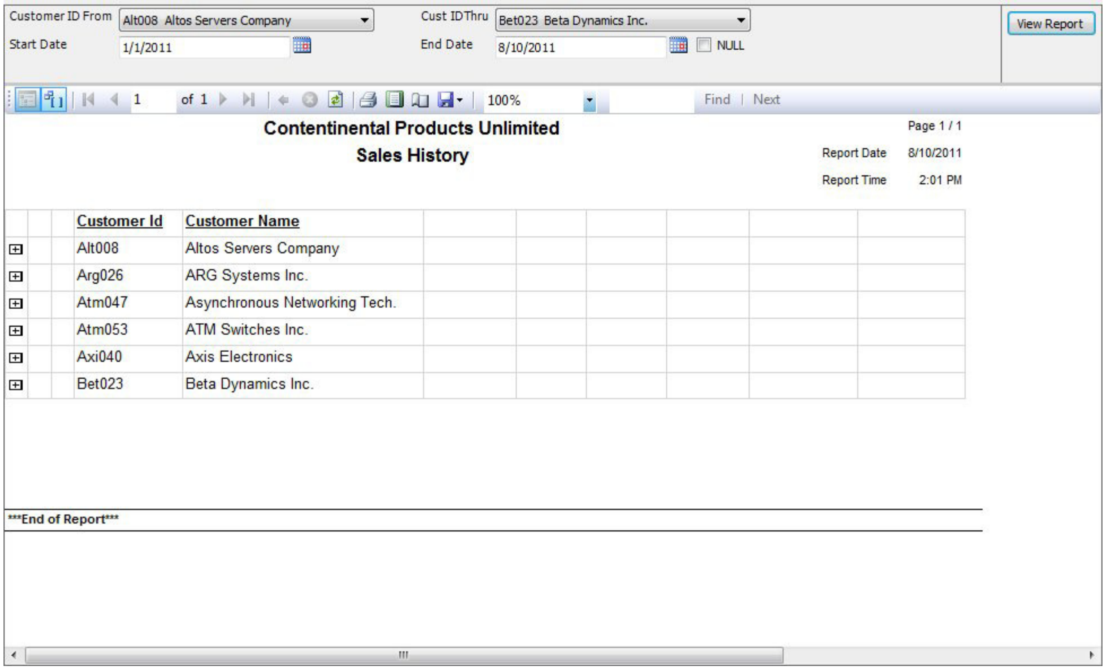
January, 2011

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Today: 8/10/2011

25. Let’s change the **Invoice Date Thru** to 8/31/2011, or click the calendar and use the arrow to go to August 2011 and select the 31st.

26. Click **View Report** and your report will now be filtered.



Customer ID From: Alt008 Altos Servers Company
Cust IDThru: Bet023 Beta Dynamics Inc.
Start Date: 1/1/2011
End Date: 8/10/2011 □ NULL

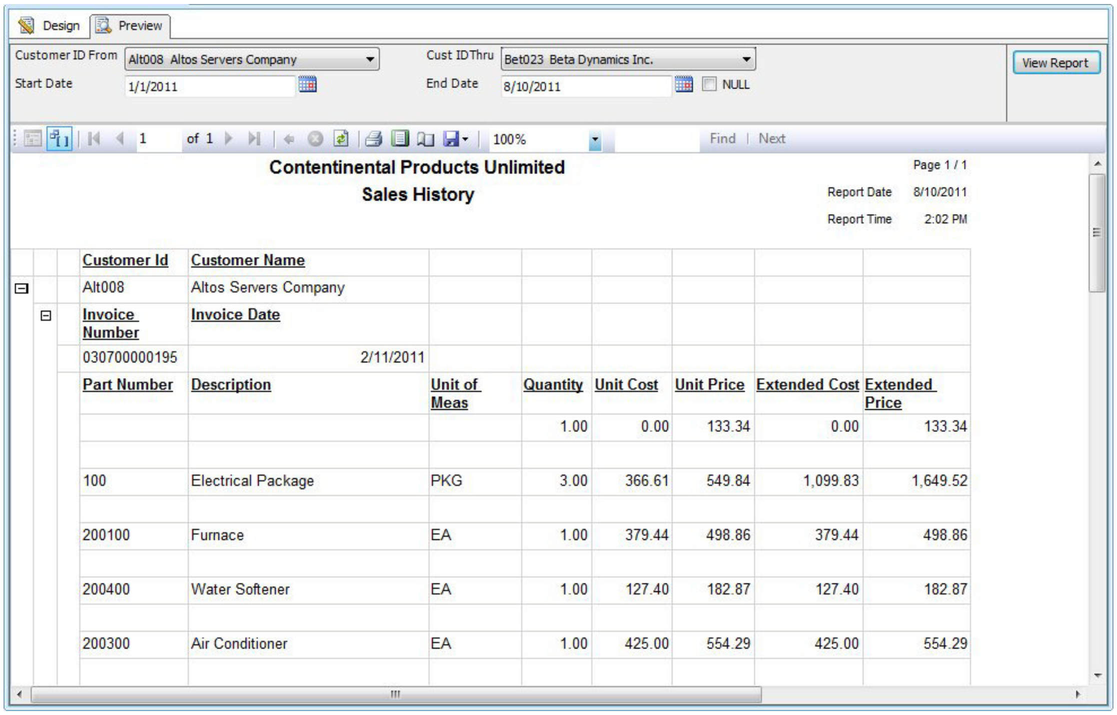
Contentinental Products Unlimited
Sales History

Page 1 / 1
Report Date 8/10/2011
Report Time 2:01 PM

	Customer Id	Customer Name
⊞	Alt008	Altos Servers Company
⊞	Arg026	ARG Systems Inc.
⊞	Atm047	Asynchronous Networking Tech.
⊞	Atm053	ATM Switches Inc.
⊞	Axi040	Axis Electronics
⊞	Bet023	Beta Dynamics Inc.

End of Report

27. Click the plus (+) next to the Customer ID and then the plus (+) next to the invoice number and we will see the items in the invoice.



Customer ID From: Alt008 Altos Servers Company
Cust ID Thru: Bet023 Beta Dynamics Inc.
Start Date: 1/1/2011
End Date: 8/10/2011 □ NULL

Contentinental Products Unlimited
Sales History

Page 1 / 1
Report Date 8/10/2011
Report Time 2:02 PM

	Customer Id	**Customer Name**						
⊟	Alt008	Altos Servers Company						
⊟	**Invoice Number**	**Invoice Date**						
	030700000195	2/11/2011						
	Part Number	**Description**	**Unit of Meas**	**Quantity**	**Unit Cost**	**Unit Price**	**Extended Cost**	**Extended Price**
				1.00	0.00	133.34	0.00	133.34
	100	Electrical Package	PKG	3.00	366.61	549.84	1,099.83	1,649.52
	200100	Furnace	EA	1.00	379.44	498.86	379.44	498.86
	200400	Water Softener	EA	1.00	127.40	182.87	127.40	182.87
	200300	Air Conditioner	EA	1.00	425.00	554.29	425.00	554.29

Task Notes:

Each time we added a parameter to the query in the dataset, the Report Designer created a corresponding report parameter for us. When the report is viewed, the values entered for the report parameters are automatically passed on to the query parameters before the query is executed. In this way, the user can enter information and have it used in the WHERE clause of the SELECT statement to affect the contents of the report.

The Report Parameters dialog box enables you to control the user's interaction with the report parameters. We can change the prompts the user sees. We can even determine the default value for a parameter.

One of the most powerful features of the Report Parameters dialog box is the capability to create a drop-down list from which the user can select a value for a parameter. In many cases, the user will not know values, such as department codes, part numbers, and so forth without looking them up. This capability to enable the user to select valid values from a list makes the reports much more user-friendly.

CHAPTER SUMMARY:

We have seen what the Report Wizard can do for you. It can provide you with a great starting place for a number of reports. However, the Report Wizard does have its limitations and, in most cases, you need to make additions to the reports it generates before they are ready for the end user.

We have also seen how to make a template so we can have a consistent look to your reports and how to save that template so it can be selected when we are making new reports.

The real power of the Reporting Services comes when we build the reports without using the Report Wizard, as we did with the Sales History Report. With just some minor adjustments we can make the report look the way we want and also add selection criteria to help the users get just the information they want on the reports.

Hopefully this will give you a good base knowledge of what the Report Designer can do for you, so you can make reports that will suit your users needs.

Now, it is time to move on to the Report Manager. In the next chapter, we will look at ways to put our reports into Report Manager and ways to administer those reports once they are there.

Steps for Making Reporting Services Report

1. Install and set up Reporting Services and Report Manager
2. Start SQL Server Business Intelligence Development Studio (Visual Studio 2005)
3. Create a Project
4. Create a new Report
5. Establish data connection
6. Select information to include on report
7. Determine grouping and sorting
8. Choose report style
9. Test and preview
10. Modify design if needed
11. Deploy

DESIGNING REPORTS

Chapter Summary:

REPORT MANAGER

4

OVERVIEW

In the previous chapter we focused on report authoring. We learned techniques for creating basic reports. However, the fact is, reports are not much good if you cannot easily share them with end users. In this chapter we learn how to do just that. We move from authoring to managing reports and delivering them to the end users. This is done through the Report Server and its Report Manager web interface.

We took a brief look at the Report Server and the Report Manager in Chapter 1. Now, we take a more detailed look. Much of our examination focuses on the Report Manager, and how it is used to access and control the Report Server.

The first step is moving our report definitions and supporting files from the development environment to the Report Catalog. Recall that the report catalog is the SQL Server 2005 database where the Report Server keeps all its information. This information includes the definitions of the reports it is managing. We look at several ways to accomplish this report deployment.

In short, in this chapter, we take our reports from a single-user development environment to a secure, managed environment where they can be executed by a number of users.

Before we deploy reports to the Report Server, we need to have an understanding of the way the Report Server organized reports in the Report Catalog. In the Report Catalog, reports are arranged into a system of folders similar to the Windows file system. Folders can contain reports, supporting files such as external images and shared data sources, end even other folders. The easiest way to create, view and maintain these folders is through the Report Manager.

Although the Report Catalog folders look and act like Windows file system folders, they are not actual file system folders. You cannot find them anywhere in the file system on the computer running the Report Server. *Report Catalog folders* are screen representations of record in the Report Catalog database.

Each folder is assigned a name. Folder names can include just about any character, including spaces. However, folder names cannot include any of the following characters:

; ? : @ & , \ * < > | “/

In addition to a name, folders can also be assigned a description. The description can contain a long explanation of the contents of the folder. The description can help users determine what type of reports are in a folder without having to open that folder and look at the contents. Both the folder name and the description can be searched by a user to help them find a report.

THE REPORT MANAGER

The Report Manager web application provides a straightforward method for creating and navigating folders in the Report Catalog. When you initially install Reporting Services, the Home folder is created by default. This is the only folder that exists at first.

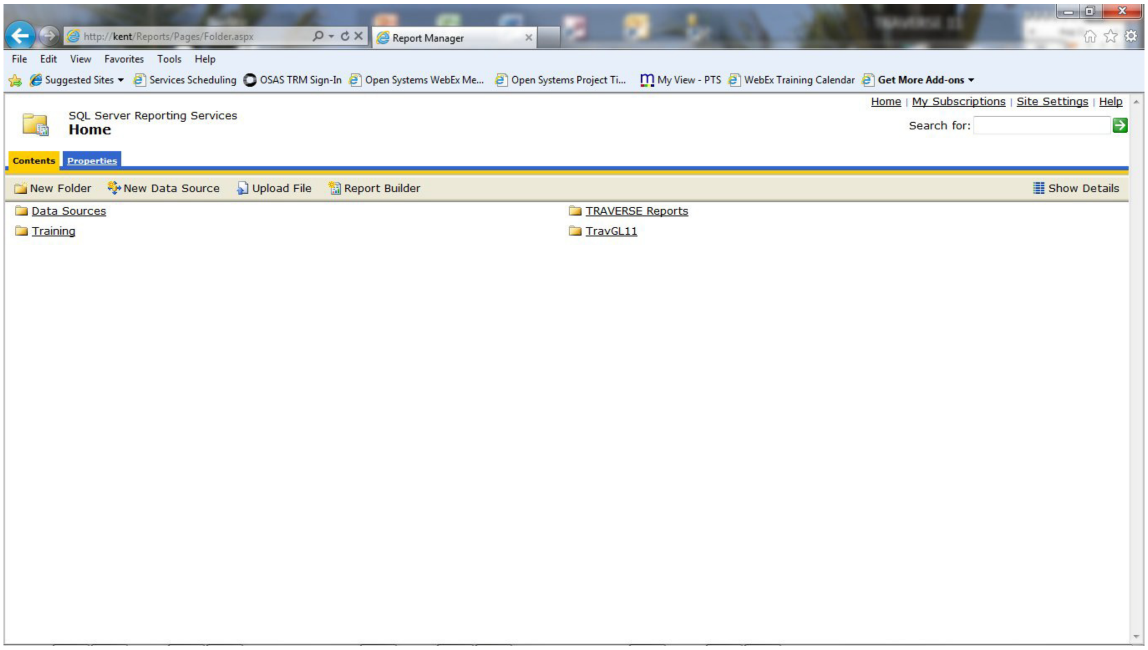
Use the following URL to access the Report Manager site on the computer running Reporting Services:

http://ComputerName/reports

In this case, ComputerName is the name of the computer where Reporting Services is installed and reports is the Report Manager Virtual Directory you set up in the Reporting Server Configuration. If you are using a secure connection to access the Report Manager site, replace http: with https:. If you are on the same computer where Reporting Services is running you can use the following URL:

http://localhost/reports

No matter how you get there, when you initially access the Report Manager it appears similar to the picture below.



Notice that the URL shown is a bit different from the URLs given previously. This is because the Report Manager web application redirects you to the Pages/Folder.aspx web page.The folder.aspx page is used to display folder contents.

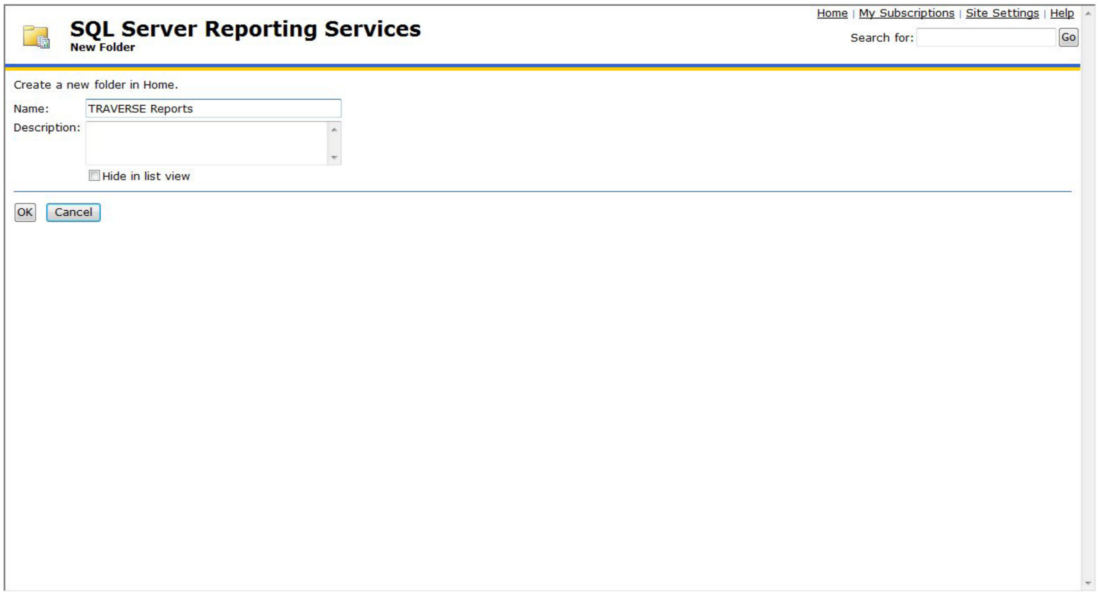
NOTE: This picture shows the Report Manager as it appears for a user with content manager privileges. If you do not see the New Folder, New Data Source, Upload File and Report Builder buttons in the toolbar on the Contents tab, you do not have content manager privileges and will be unable to complete the exercises in this section. If possible, log out and log in with a Windows login that has local administration privileges on the computer running the Report Server.

To use the Report Manager, you must be using Microsoft Internet Explorer 6.0 with Service Pack 1 (SP1) or Internet Explorer 7. In either case, you must have scripting enabled.

Adding a New Folder Using the Report Manager

Let's create a new folder into which we will deploy some of the Continental Products Unlimited reports from the previous chapter. Here are the steps to follow:

1. Click the **New Folder** button in the toolbar on the Content tab. The New Folder page appears, as shown below.



SQL Server Reporting Services
New Folder

Create a new folder in Home.

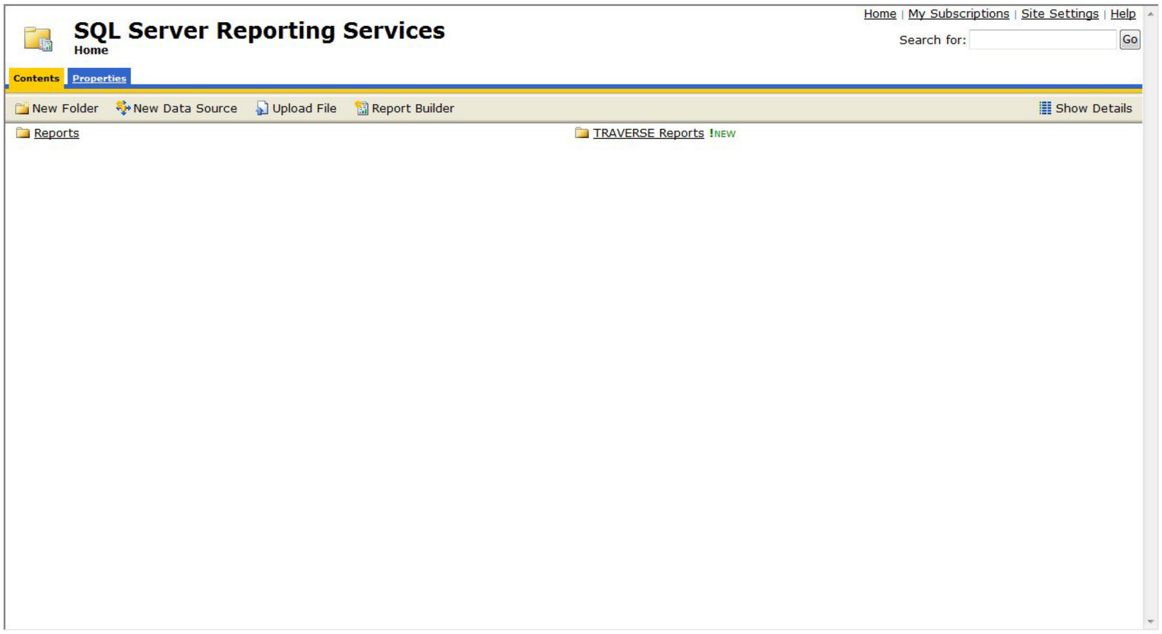
Name: TRAVERSE Reports

Description:

□ Hide in list view

2. Type **TRAVERSE Reports** for the name and leave the **Description** blank.

3. Click **OK** to create the new folder and return to the Home folder.



You see an entry for your new folder with its name and description and the Contents tab of the Home folder. The text !New next to the folder name remains there for 48 hours. This helps to notify users of new content added to your Report Server.

If you were observant, you noticed one item on the New Folder page we did not use. (If you missed it, look at the image following step 1.) This is the Hide in List View check box. When the Hide in List View check box is checked, the new folder does not appear on the Content tab. This is useful when you want to make the report in a folder available through a custom interface, but unavailable through the Report Manager.

To view the contents of the new folder, click the folder name. The name of the current folder appears in bold text near the top of the page. Immediately above the name of the current folder is the path from the Home folder to the current folder. Because the TRAVERSE Reports folder is in the home folder, the path only contains Home>. You can return to any folder in the current path by clicking that folder name in the path shown near the top of the page. You can return to the Home folder by clicking Home at the beginning of the current path or by clicking Home in the upper-right corner of the page.

Moving Reports to the Report Server

Now that you know how to create folders, it is time to put some content in those folders. You do this by moving reports from the development environment to the Report Server. This can be done using a number of different methods. We look at two of those methods now: using the Report Designer and using the Report Manager.

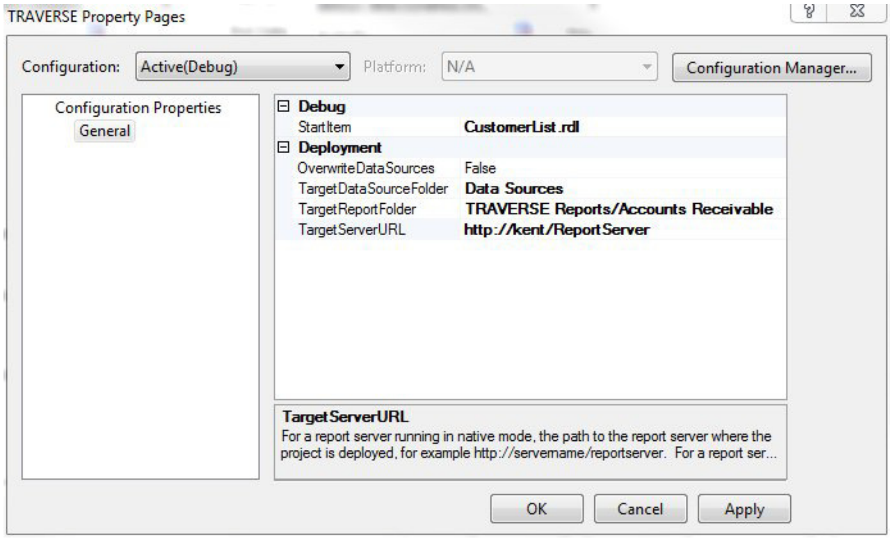
Deploying Reports Using the Report Designer

The most common method of moving reports to the Report Server is by using the Report Designer. Once you are satisfied with a report you developed, you can make it available to your users without leaving the development environment. This capability to create, preview, and deploy a report from a single authoring tool is a real plus.

Let’s try deploying the report project from Chapter 3. To do so, follow these steps:

1. Start Visual Studio or the Business Intelligence Development Studio and open the TRAVERSE project.

2. Select **Project** - **TRAVERSE Properties** from the Main menu. The **TRAVERSE Properties Pages** dialog box appears.



Configuration: Active(Debug)   Platform: N/A

Debug	
StartItem	**CustomerList.rdl**
Deployment	
OverwriteDataSources	False
TargetDataSourceFolder	**Data Sources**
TargetReportFolder	**TRAVERSE Reports/Accounts Receivable**
TargetServerURL	**http://kent/ReportServer**

TargetServerURL
For a report server running in native mode, the path to the report server where the project is deployed, for example http://servername/reportserver. For a report ser...

3. Type **TRAVERSE Reports/Accounts Receivable** for the TargetReportFolder. This is the folder into which the report is going to be deployed.

4. Type **http://ComputerName/ReportServer** for the TargetServerURL, where ComputernName is the name of the computer where the Report Server is installed. You should replace http: with https: if you are using a secure connection. You can use localhost in place of the computer name if the Report Server is installed on the same computer you are using to run Visual Studio.

5. Click **OK** to exit the TRAVERSE Property Pages dialog box.

6. Right-click the TRAVERSE project entry in the Solution Explorer and select **Deploy** from the menu.

7. The Report Designer builds all the reports in the project, and then deploys all the reports to the Report Server. (During the build process, the Report Designer checks each report for any errors that would prevent it from executing properly on the Report Server.) The results of the build and deploy are shown in the Output window at the bottom of the screen.

8. Open Report Manager in your browser. Click the TRAVERSE Reports folder to view its content. You see that Visual Studio created a new folder in the TRAVERSE Reports folder called Accounts Receivable.

9. Click the Accounts Receivable folder to view its content. All the items in the TRAVERSE project, four reports, were deployed.

10. Click the Customer List report. You see the HTML version of the Customer List report.

NOTE: You can also deploy the contents of a project by selecting Build - Deploy Solution or Build - Deploy {Project Name} form the Main menu.

Working through the web service

When the Report Designer deploys reports, it works through the Reporting Services web service. The Report Manager web application provides a human interface to Reporting Services. Because the Report Designer falls into the latter of these two categories, it uses the web services to deploy reports.

The web service has a different URL than the Report Manager. You must enter the URL for the web service and not the Report Manager in the properties pages dialog box for the deployment to work properly. The default URL for the web service is shown in Step 4 in the previous section.

Creating folders while deploying

In steps 2 through 5, we entered information into properties of the TRAVERSE project. These values tell the Report Designer where to put the reports when the project is deployed. In this case, we instructed the Report Designer to put our reports in the Accounts Receivable folder within the TRAVERSE Reports folder.

We created the TRAVERSE Reports folder in the previous section. We did not create the Accounts Receivable folder. Instead, the Report Designer created that folder for us as it deployed the items in the project. In fact, the Report Designer creates folders for any path you specify.

Deploying a single report

In Step 6, we used the project's Right-click menu to deploy all the items in the project. Alternatively, we could have right-clicked a report and selected Deploy from the report's Right-click menu. However, this would have deployed only this report, not the entire project.

On some occasions, we might want to deploy a single report rather than the entire project. At times, one report is going to be completed and ready for deployment, while the other reports in the project are still under construction. At other times, one report will be revised after the entire project has already been deployed. In these situations, it is only necessary to redeploy the single revised report.

Uploading Reports Using Report Manager

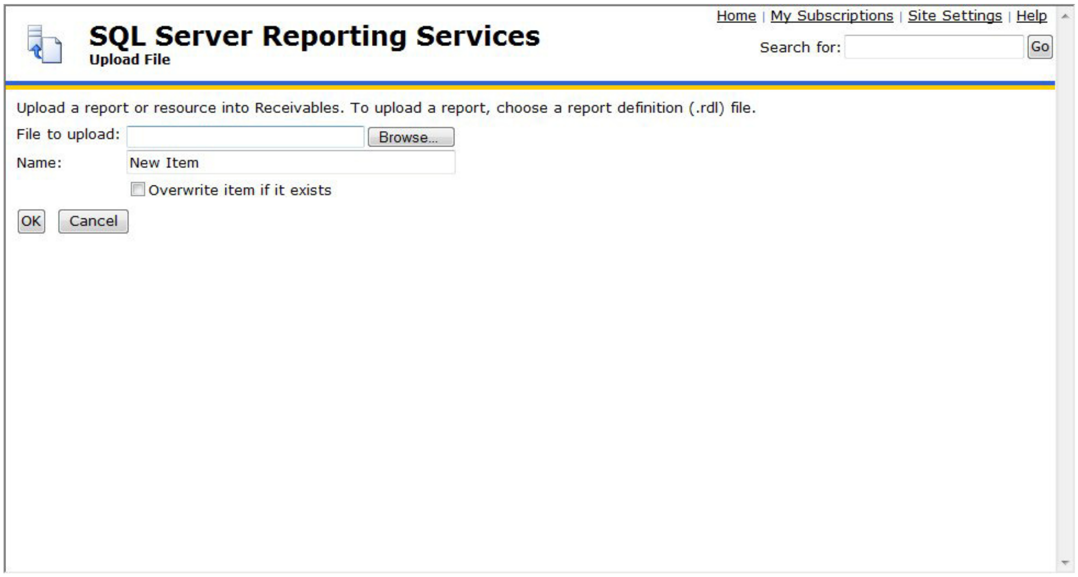
Another common method of moving a report to the Report Server is by using the Report Manager. This is known as *uploading* the report. Deploying reports from the Report Designer can be thought of as pushing the reports from the development environment to the Report Server, whereas uploading reports from the Report Manager can be thought of as pulling the reports from the development environment to the Report Server.

We may need to use the Report Manager upload in situations where our report authors do not have rights to deploy reports on the Report Server. The report authors create their reports and test them within the Report Designer. When a report is completed, the report author can place the RDL file for the report in a shared directory or send it as an e-mail attachment to the Report Server administrator. The Report Server administrator can upload the RDL file to a quality assurance Report Server and test the report for clarity, accuracy, and proper use of database resources. Once the report has passed this review, the Report Server administrator can upload the report to the production Report Server.

Let's try uploading some of the reports from the TRAVERSE report project:

1. Open the Report Manager in your browser. Click the TRAVERSE Reports folder to view its content.
2. Create a new folder called **Receivables.**
3. Select the new folder to view its contents.

4. Click the **Upload File** button in the toolbar on the **Contents** tab. The Upload File page appears, as shown below.


Home | My Subscriptions | Site Settings | Help
SQL Server Reporting Services
Upload File
Search for:
Go

Upload a report or resource into Receivables. To upload a report, choose a report definition (.rdl) file.

File to upload: Browse...

Name: New Item

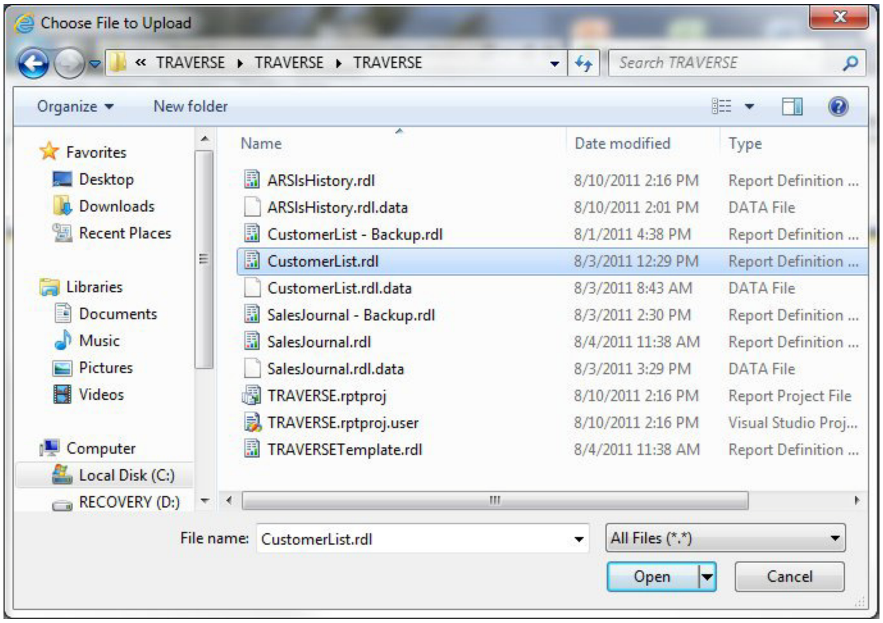
□ Overwrite item if it exists

5. Click **Browse**. The Choose file dialog box appears.

6. Navigate to the folder where we created our solution for TRAVERSE. If this folder is in the default location, you can find it under the following path:

   My Documents\Visual Studio 2008\Projects\MSSQLRS\TRAVERSE

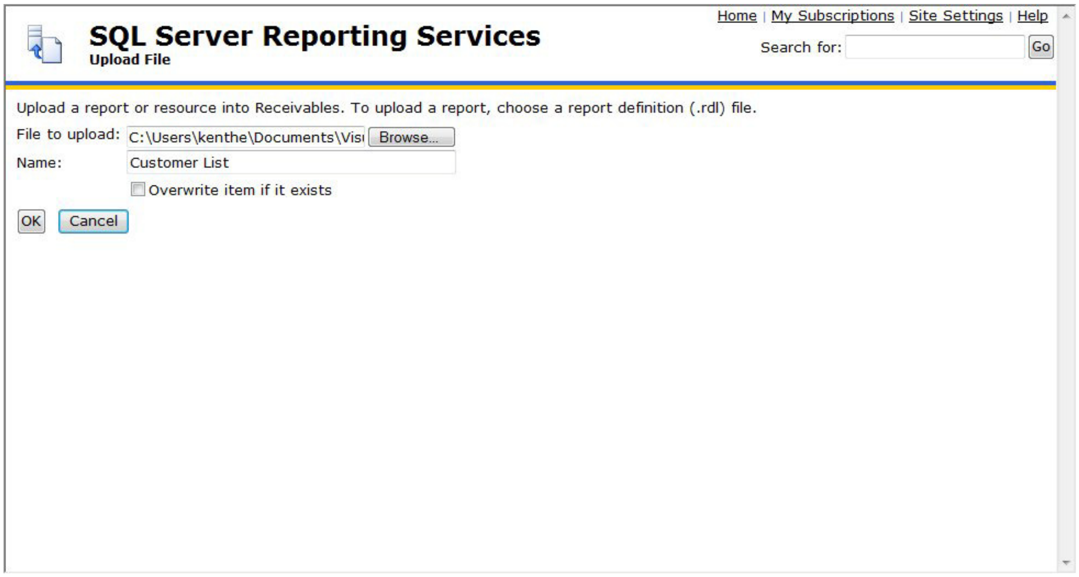
7. Select the **Customer List** report (**CustomerList.rdl**) and click **Open** to edit the Choose file dialog box.



Name	Date modified	Type
ARSIsHistory.rdl	8/10/2011 2:16 PM	Report Definition ...
ARSIsHistory.rdl.data	8/10/2011 2:01 PM	DATA File
CustomerList - Backup.rdl	8/1/2011 4:38 PM	Report Definition ...
CustomerList.rdl	8/3/2011 12:29 PM	Report Definition ...
CustomerList.rdl.data	8/3/2011 8:43 AM	DATA File
SalesJournal - Backup.rdl	8/3/2011 2:30 PM	Report Definition ...
SalesJournal.rdl	8/4/2011 11:38 AM	Report Definition ...
SalesJournal.rdl.data	8/3/2011 3:29 PM	DATA File
TRAVERSE.rptproj	8/10/2011 2:16 PM	Report Project File
TRAVERSE.rptproj.user	8/10/2011 2:16 PM	Visual Studio Proj...
TRAVERSETemplate.rdl	8/4/2011 11:38 AM	Report Definition ...

File name: CustomerList.rdl

8. Click **OK** to upload the file.



SQL Server Reporting Services
Upload File

Upload a report or resource into Receivables. To upload a report, choose a report definition (.rdl) file.

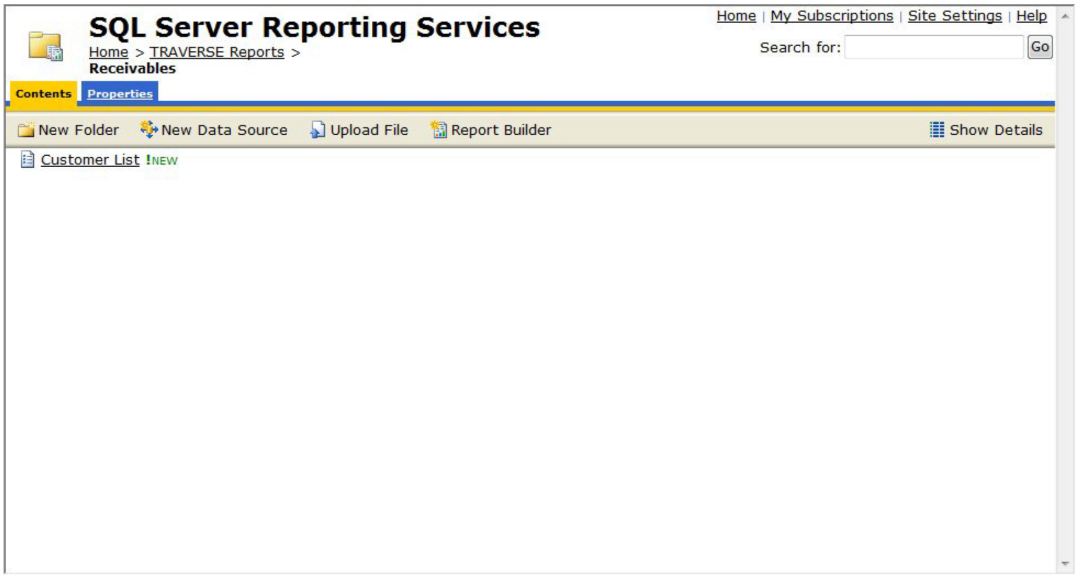
File to upload: C:\Users\kenthe\Documents\Visı

Name: Customer List

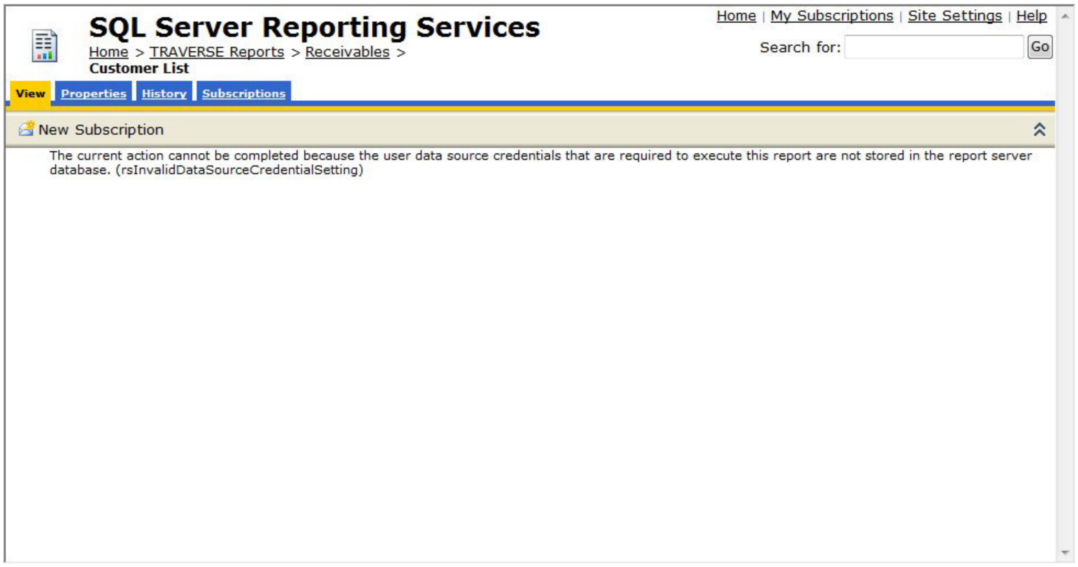
☐ Overwrite item if it exists

NOTE: You can upload the rest of your reports to folders using these steps.

9. The **Customer List** report has been uploaded to the **Receivables** folder.

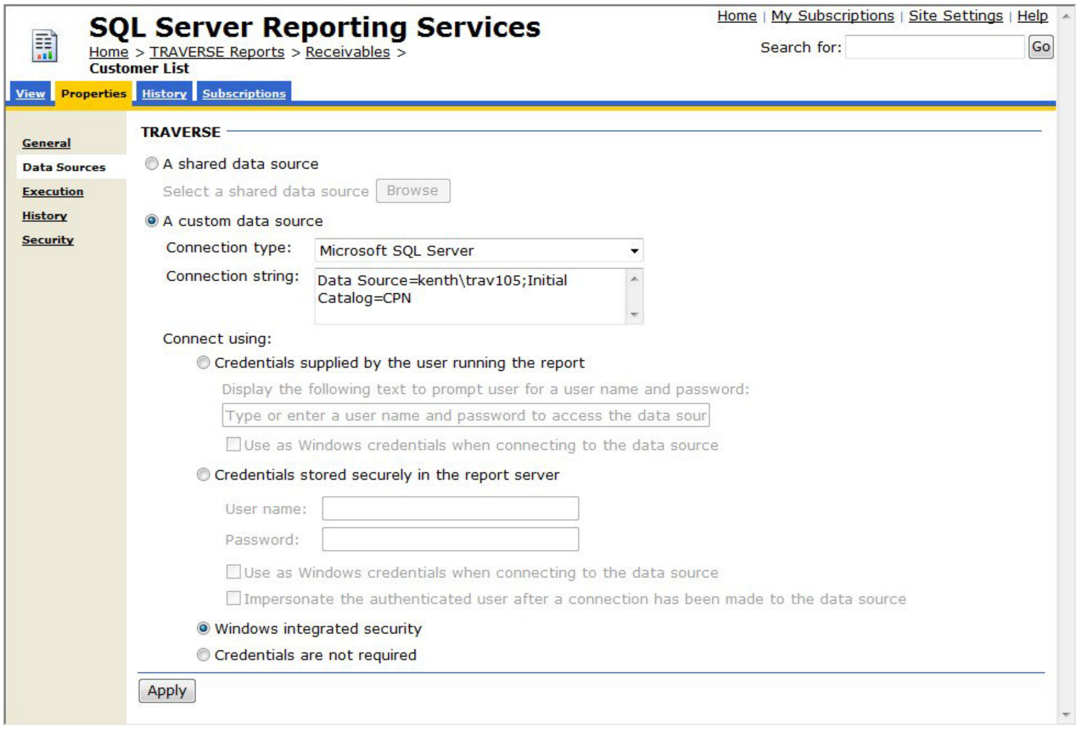


10. Click the **Customer List** to execute it. You see an error similar to the one shown below. You received this because, when the report was uploaded it did not bring the server login credentials with it.


SQL Server Reporting Services
Home > TRAVERSE Reports > Receivables >
Customer List
Home | My Subscriptions | Site Settings | Help
Search for:
Go
View
Properties
History
Subscriptions
New Subscription

The current action cannot be completed because the user data source credentials that are required to execute this report are not stored in the report server database. (rsInvalidDataSourceCredentialSetting)

11. Click the **Properties** tab for the Customer List. Then select **Data Source** on the left side of the window to display the Data Sources window.



SQL Server Reporting Services

Home > TRAVERSE Reports > Receivables >
Customer List

Home | My Subscriptions | Site Settings | Help

Search for: Go

View | Properties | History | Subscriptions

General
Data Sources
Execution
History
Security

TRAVERSE

○ A shared data source

Select a shared data source Browse

● A custom data source

Connection type: Microsoft SQL Server

Connection string: Data Source=kenth\trav105;Initial Catalog=CPN

Connect using:

○ Credentials supplied by the user running the report

Display the following text to prompt user for a user name and password:

Type or enter a user name and password to access the data sour

□ Use as Windows credentials when connecting to the data source

○ Credentials stored securely in the report server

User name:

Password:

□ Use as Windows credentials when connecting to the data source

□ Impersonate the authenticated user after a connection has been made to the data source

● Windows integrated security

○ Credentials are not required

Apply

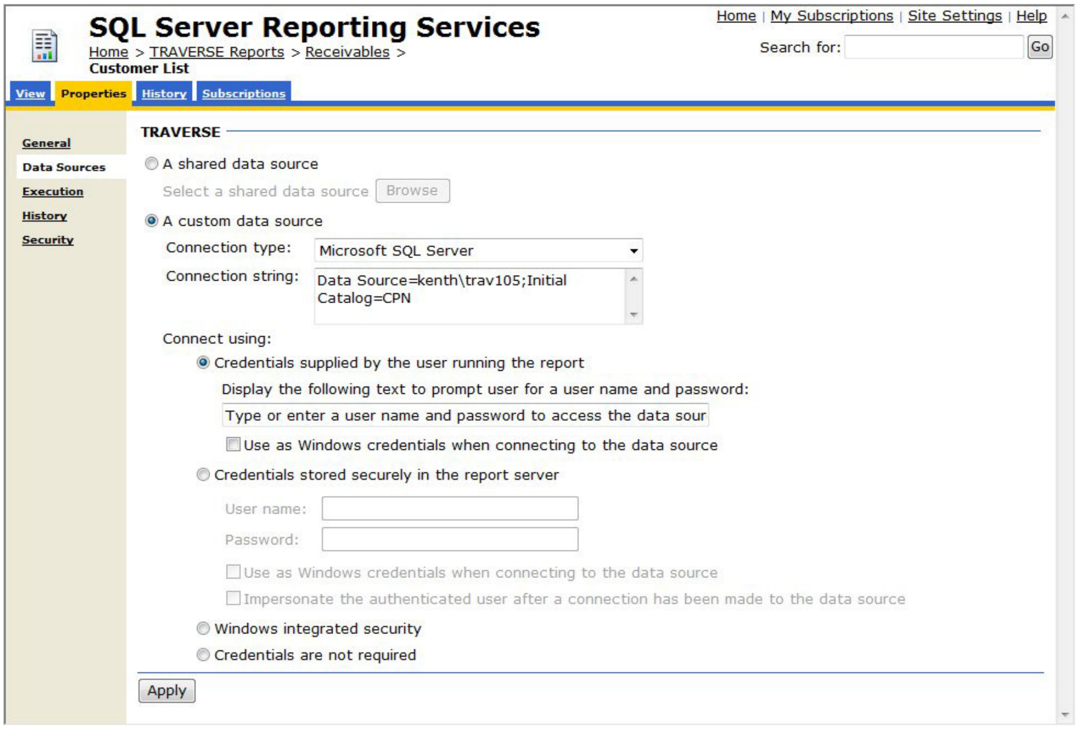
12. You should see the **A custom data source** radio button selected. The **Connection type** should be **Microsoft SQL Server**. The Connection string should read something like this:

    Data Source=TRAVERSE SQL Server\Instance;Initial Catalog=CPU

    Where TRAVERSE SQL Server\Instance is your TRAVERSE SQL server name and instance.

13. If you are using the Windows Authentication (Trusted Connection) to log into your TRAVERSE menu, select the **Windows integrated security** radio button at the bottom of the screen. This will require no further login.

14. If you are using the SQL Authentication to log into your TRAVERSE menu, select the **Credentials supplied by the user running the report**. When the report is rendered the user will be asked to enter their Log In Name and Password.


SQL Server Reporting Services
Home > TRAVERSE Reports > Receivables >
Customer List
Home | My Subscriptions | Site Settings | Help
Search for:
Go
View
Properties
History
Subscriptions
General
Data Sources
Execution
History
Security
Browse
Apply

TRAVERSE

○ A shared data source

Select a shared data source

● A custom data source

Connection type: Microsoft SQL Server

Connection string: Data Source=kenth\trav105;Initial Catalog=CPN

Connect using:

● Credentials supplied by the user running the report

Display the following text to prompt user for a user name and password:

Type or enter a user name and password to access the data sour

□ Use as Windows credentials when connecting to the data source

○ Credentials stored securely in the report server

User name:

Password:

□ Use as Windows credentials when connecting to the data source

□ Impersonate the authenticated user after a connection has been made to the data source

○ Windows integrated security

○ Credentials are not required

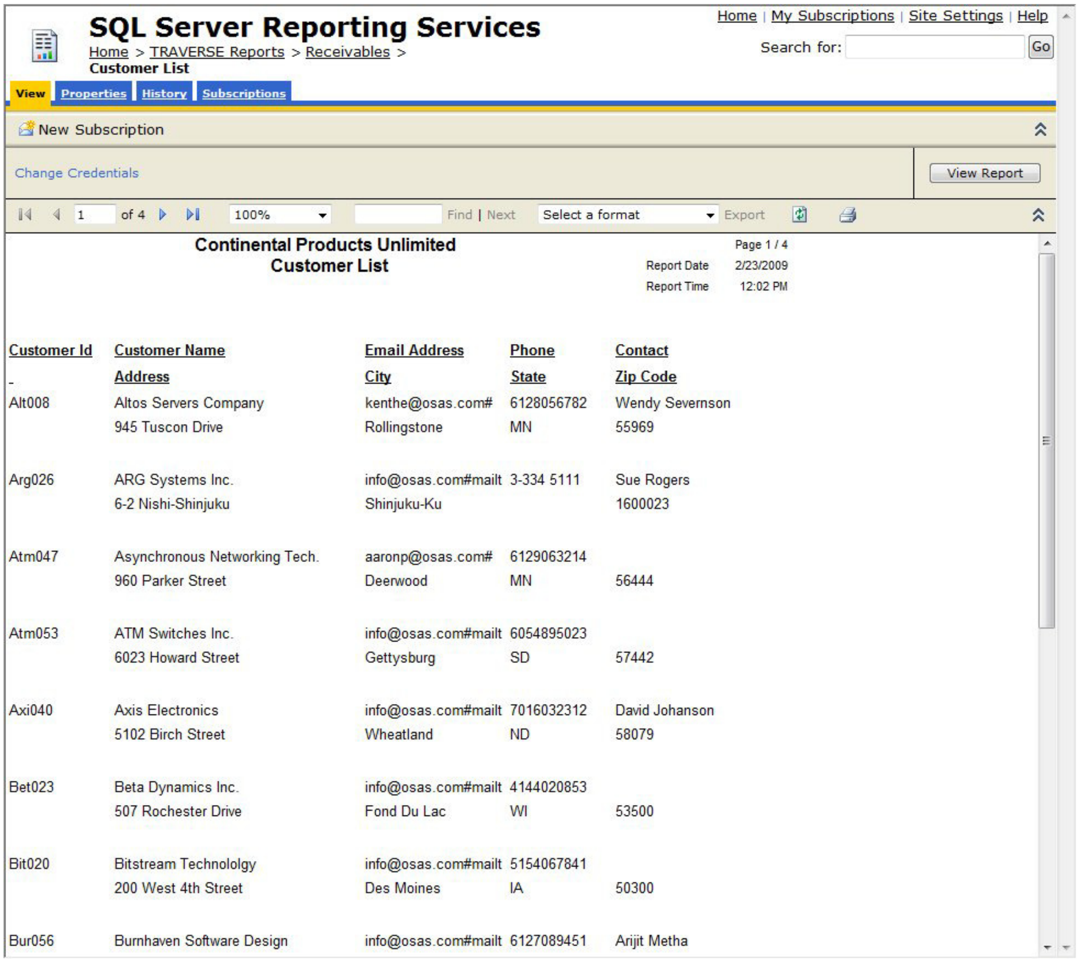
15. Click **Apply** to apply the data source changes we made.

16. Click the **View** tab to view your report.

If you selected to use the **Windows integrated security**, you will see the report generated without requiring any further login.

If you selected to use the **Credentials supplied by the user running the report**, you will see **Log In Name** and **Password** fields to enter your TRAVERSE user name and password.

Once you have entered your user name and password click the **View Report** button.



Continental Products Unlimited
Customer List

Page 1 / 4
Report Date 2/23/2009
Report Time 12:02 PM

Customer Id	Customer Name	Email Address	Phone	Contact
-	Address	City	State	Zip Code
Alt008	Altos Servers Company	kenthe@osas.com#	6128056782	Wendy Severnson
	945 Tuscon Drive	Rollingstone	MN	55969
Arg026	ARG Systems Inc.	info@osas.com#mailt	3-334 5111	Sue Rogers
	6-2 Nishi-Shinjuku	Shinjuku-Ku		1600023
Atm047	Asynchronous Networking Tech.	aaronp@osas.com#	6129063214	
	960 Parker Street	Deerwood	MN	56444
Atm053	ATM Switches Inc.	info@osas.com#mailt	6054895023	
	6023 Howard Street	Gettysburg	SD	57442
Axi040	Axis Electronics	info@osas.com#mailt	7016032312	David Johanson
	5102 Birch Street	Wheatland	ND	58079
Bet023	Beta Dynamics Inc.	info@osas.com#mailt	4144020853	
	507 Rochester Drive	Fond Du Lac	WI	53500
Bit020	Bitstream Technololgy	info@osas.com#mailt	5154067841	
	200 West 4th Street	Des Moines	IA	50300
Bur056	Burnhaven Software Design	info@osas.com#mailt	6127089451	Arijit Metha

Connect Using Options

When you are accessing data from a server-based database, such as SQL Server or Oracle, you need to provide some type of credentials, usually a user name and password, to show you have rights to access the data. Keeping these credentials secure is an important concern. The shared data sources created on the Report Server provide several methods for specifying these credentials.

When entering the connection string into a data source, it is best not to include the credentials in the connection string itself. The connection string is displayed as plain text to anyone who views the Data Source Properties page. To better protect password information, always enter the credential information under one of the Connect Using options described here.

Credentials supplied by the user

The first Connect Using option is to have the user enter the credentials required by the data source each time the report is run. This is the “Credentials supplied by the user running the report” option. You can specify the prompt to be presented each time the user must enter these credentials. If the “Use as Windows Credentials When Connection to the Data Source” check box is checked, the user name and password entered by the user are treated as a Windows login. This means the user name and password provide database access using Windows Integrated security. If this check box is not checked, the user name and password are treated as a database login.

Having the user enter the credentials each time the report is run is the most secure option. No login information is stored with the data source, but most users are not pleased with a system where they must enter login information each time they run a report. This option may be appropriate when your organization’s security policy forbids storing login information in any way. In most other cases the other Connect Using options provide a better solution.

Credentials stored in the report server

The next option enables you to have the user name and password stored in the Report Catalog on the Report Server. This is the “Credentials stored securely in the report server” option. The user name and password entered with this option are encrypted when they are stored in the Report Catalog. Also, the password is not displayed to the user in the Data Source Properties page.

This Connect Using option is convenient for the user because they do not need to remember and enter credentials to run reports using this data source. It also provides the required security for most situations through the measures noted in the previous paragraph.

As with the first Connect Using option, there is a “Use as Windows Credentials When Connecting to the Data Source” check box here as well. If this check box is checked, the user name and password stored in the Report Catalog are treated as a Windows login. If this check box is not checked, the user name and password are treated as a database login.

The second check box under this Connect Using option is “Impersonate the Authenticated User After a Connection Has Been Made to the Data Source”. If this check box is checked, the data source can use these credentials to impersonate this user. Not all database servers support this type of delegation of credentials. Consult the documentation for your specific database server for more information.

Integrated Security

If you are not comfortable with storing credentials in the Report Catalog, but you do not want your users entering credentials every time a report is run, integrated security may be the solution for you. The “Windows integrated” security option does not require the user to enter credentials. Instead, it takes the Windows login credentials that let the user access the Report Manager and passes them along to the database server. Your database server, of course, needs to be set up to accept these credentials.

Integrated security always works when the data source exists in the same server as the Report Server. It may run into problems, however if the data source is on another server. The problems are caused by the way integrated security works between servers.

For a better understanding of the problems with integrated security, let’s look at an example of the way integrated security works. The user logs in to their computer. This computer knows everything about this user because the original authentication occurs here.

When the user accesses the Report Manager application, the user’s credentials are passed from the original computer to the computer hosting the Report Server. However, using standard Windows security, not everything about this login is passed to the Report Server computer - only enough to authenticate the user is passed. Some sensitive information does not make this hop across the network.

When the user runs a report with a data source using integrated security, the Report Server must pass on the credentials to the database server. However, the Report Server does not have the complete credentials to pass along. In fact, it does not know enough about the user to successfully authenticate them on the database server. The authentication on the database server fails. Using standard Windows security, integrated security only works across one hop, from the original authenticating computer to a second computer. In the case of the Report Manager, this is the hop from the user’s computer to the Report Server.

To get integrated security to work across more than one hop, your Windows domain must use a special kink of security known as Kerberos, which allows authentication across multiple hops. Using Kerberos security, integrated security works across any number of servers in the network.

Credentials not required

The final Connect Using option is for data sources that do not require any authentication. This option would be used for connection to some Access databases, Fox Pro databases, and others that do not require any login or password. This option could also be used if you insist, despite prior warnings here, on putting your credentials right in the connection string.

Managing Items in Folders

You now know how to load items into folders on the Report Server. Of course, we live in a dynamic world, so things seldom stay where they are originally put. We need to be able to move items around as we come up with better ways of organizing them. We also need to be able to delete items as they are replaced by something better or are simply not needed anymore. Fortunately, the Report Manager provides ways for us to do this housekeeping in an efficient manner.

Moving Items Between Folders

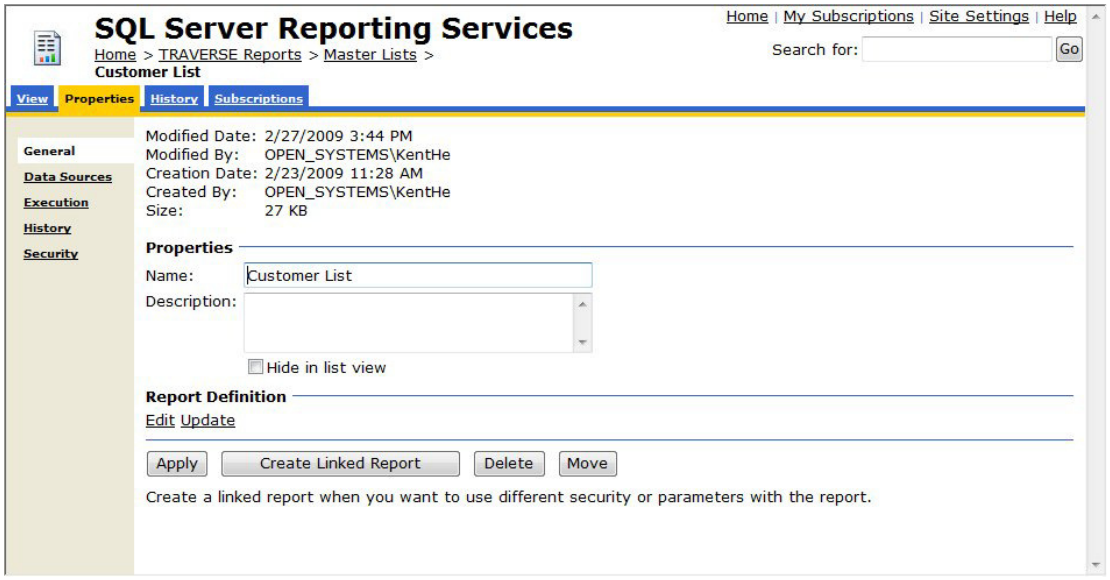
As an example, let's create a more descriptive folder for our Customer List report. We begin by moving a single item to this new folder. Then, we look at a method for moving multiple items at the same time.

Moving a single item

Here are the steps to follow to move a single item:

1. Open the **Report Manager** in your browser and navigate to the **TRAVERSE Reports** folder.
2. Click **New Folder**. The New Folder page appears.
3. Type **Master Lists** for the **Name** and type **Lists for setup items like Customers and Vendors** for the **Description**.
4. Click **OK** to create the new folder.
5. Click **Receivables** to view the contents of this folder.
6. Click **Show Details** on the right side of the page.

7. Click the icon in the **Edit** column for the **Customer List** report. The Customer List report **Properties** tab appears.

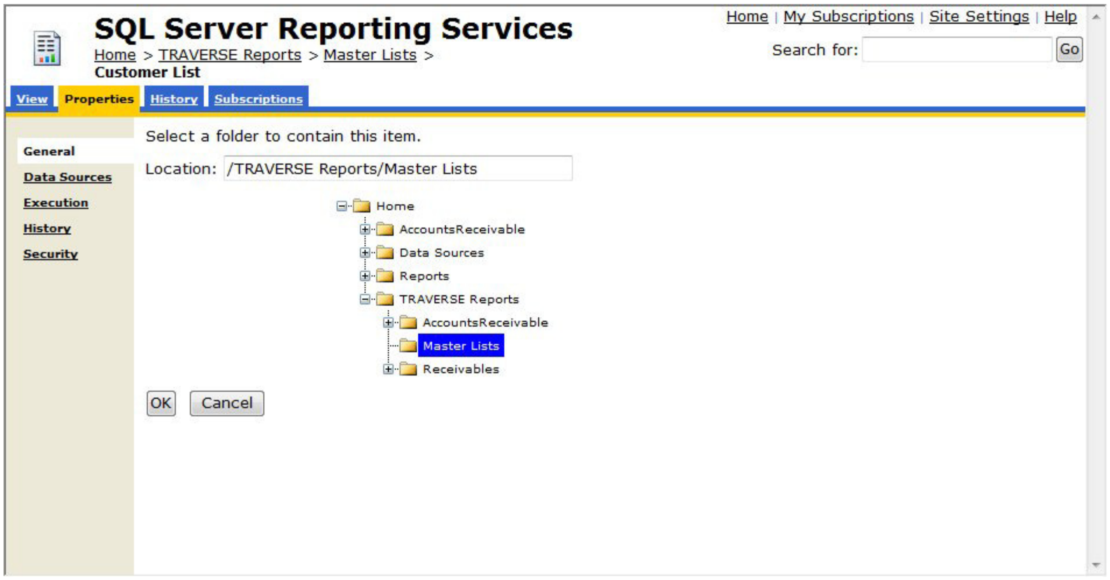


Modified Date: 2/27/2009 3:44 PM
Modified By: OPEN_SYSTEMS\KentHe
Creation Date: 2/23/2009 11:28 AM
Created By: OPEN_SYSTEMS\KentHe
Size: 27 KB

Name: Customer List
Description:
□ Hide in list view

Create a linked report when you want to use different security or parameters with the report.

8. Click **Move**. The Move Items page appears.



Select a folder to contain this item.
Location: /TRAVERSE Reports/Master Lists

9. Select the **Master Lists** folder in the tree view.

10. Click OK to move the report to this folder.

11. Click the **Master Lists** link at the top of the page to view the contents of this folder.

Moving multiple items

You can see the Customer List report has been moved to the Master Lists folder. However we have other reports that are not grouped the way we want them to be. We could move each report individually, as we did with the Customer List report, this could be time consuming. Fortunately there is another way:

1. Click the TRAVERSE Reports link at the top of the page.
2. Click **New Folder** to add a new folder to our **TRAVERSE Reports** folder. Type **History Reports** for the **Name** and **Reports that will show transaction history** for the Description.
3. Click **OK** to add the new folder to our TRAVERSE Reports page.
4. Click **Accounts Receivable** to view the contents of this folder.
5. In the **Detail** view, you can see check boxes next to each item in the folder. These check boxes work with the **Delete** and **Move** buttons in the Contents tab toolbar. When you click **Delete**, any checked items are deleted. Likewise, when you click **Move**, any checked items are moved.

   You can click the uppermost check box (the check box to the left of the word “Edit”). Checking this check box checks all items in the folder

6. Click the boxes next to the **ARSalesHistory** and **SalesJournal** reports. Checking multiple boxes will allow us to move multiple items at the same time in one step.
7. Click **Move** in the Contents tab toolbar. The Move Multiple Items page appears.
8. Select the **History Reports** folder in the tree view.
9. Click **OK** to move the items to this folder.

This method works for moving a single item, multiple items, or the entire contents of a folder. Just check the items you want to move and click the move button. Remember, you need to be in the Detail view when using this method.

This section demonstrated moving reports. You can also move whole folders using the same techniques.

Deleting a folder

The Accounts Receivable folder is now empty and ready to be deleted. As with the Move function, you can accomplish this in two ways. The first way is to view the Properties tab for the folder you want to delete, and then click the Delete button. Just for fun we'll try the second method.

Deleting a folder using the check boxes and toolbar

1. Click the **TRAVERSE Reports** link at the top of the page to view the contents of this folder.
2. Check the **Accounts Receivable** folder.
3. Click **Delete**. The confirmation dialog box appears.
4. Click **OK** to confirm your deletion. The **Accounts Receivable** folder is deleted.

Folders do not need to be emptied before they are deleted. If the Accounts receivable folder had contained reports, the supporting items, or even other folders, would have been deleted along with the folder.

Renaming a folder

In addition to moving and deleting items, we may also want to rename items. Let's give the Receivables folder a more descriptive name:

1. Click the icon in the **Edit** column for the Receivables folder. The **Receivables Properties** tab appears.
2. Replace the contents of Name by typing **Journals**. Then type **Reports that show current existing transactions waiting to be posted** for the Description.
3. Click **Apply**.
4. Click the **TRAVERSE Reports** link at the top of the page.
5. Click **Hide Details**.

This same technique makes it just as easy to change the names and descriptions for reports and other items. Just because it is easy to make these changes does not mean you should do it often. Once users become familiar with a folder name, a report name, or a report's location within the folder structure, you should change it only if you have a good reason to do so.

Printing from Report Manager

No matter how coinvent you make it for your users to access reports in a browser, and no matter how many interactive drill-down and drill-through features you provide, your users always want to print their reports on paper. You can explain all the wonders of the multiple, cascading parameters you have created until you are blue in the face, but some users always need to touch and feel the numbers on paper. They need to be able to put something in a briefcase and take it home with them at night. It doesn't matter that they could receive up-to-date numbers through their VPN at home. They want ink on paper.

Printing options

Reporting Services provides several options for printing a report from Report Manager. Each provides some advantages and disadvantages for the user.

HTML Printing

These users could just press the print button on their browser and get whatever type of printout HTML printing provides. As you are probably aware, HTML printing is not a good choice when formatting is important as it usually is for reports. Lines of text can wrap in unusual ways or simply be cut off. A line of text at the bottom of the page can even be cut right in half, with the top-half on one page and the bottom-half on the next page.

Fortunately, the Report Manager provides a couple of alternatives to HTML printing.

Printing from a PDF Document or TIFF File

A PDF document or TIFF file does an excellent job of maintaining report format when a report is printed. Therefore, when users want to have a high-quality report printout, they can export the report to a PDF document or a TIFF file. Once this is complete, they can view the exported report using the appropriate viewer: Adobe Acrobat Reader for the PDF document and the Windows Picture and Fax Viewer for a TIFF file. The report can then be printed using the view.

This process provides the user with a quality printout. However, not all users are comfortable with saving a file to a local disk, finding that file and opening it in the appropriate viewer, and then printing the report. There is another printing alternative which is even more straightforward.

Client-Side Printing

You may have noticed a button with a printer icon on the report toolbar. this button is for the client-side printing feature of Reporting Services. Client-side printing works through an ActiveX object downloaded to the user's computer. From then on, whenever the Client-Side Printing button is clicked, this ActiveX object provides the user interface and controls the printing.

The first time a user activates the client-side printing feature, they may be prompted with a security warning about the AcitveX download. After taking the appropriate precautions, such as making sure the ActiveX object is signed by Microsoft, the user should approve the download to enable client-side printing. Once the ActiveX has been downloaded by the first use, it does not need to be downloaded again.

If a user has trouble downloading the AcitveX control, they may need to set the Report Manager as a trusted site in their browser. This is done on the Security tab of the Internet Options dialog box. the user should not lower their security setting for all sites in general to accomplish the ActiveX download.

Once downloaded, client side printing enables users to set various report attributes. These include margins, page size, and even page orientation. Users can also preview a report before putting it on paper.

Security

In Reporting Services, security was designed with both flexibility and ease of management in mind. Flexibility is provided by the fact that individual access rights can be assigned to each folder and to each item within a folder. An item is either a report or a resource. You can specify exactly who has rights to each item and exactly what those rights are. Ease of management is provided by security inheritance, security roles, and integration with Windows security.

NOTE: Remember, although we are creating and maintaining these role assignments using the Report Manager, the security rights apply to Reporting Services as a whole. No matter how you access folders and items - through the Report Manager or through the web service - these security rights are enforced.

Integration with Windows Security

Reporting Services does not maintain its own list of users and passwords. Instead, it depends entirely on integration with Windows security. When a user accesses either the Report Manager web application or the web service, that user must authenticate with the Report Server. In other words, the user must have a valid domain user name and password, or a local user name and password, to log on to the Report Server. Both the Report Manager web application and the web service are set up requiring integrated Windows authentication to ensure this logon takes place.

NOTE: If it is possible for each report user to have their own credentials on the Report Server, it is possible to create your own custom security. You can create a security scheme such as forms-based security to enable the users to authenticate and access reports.

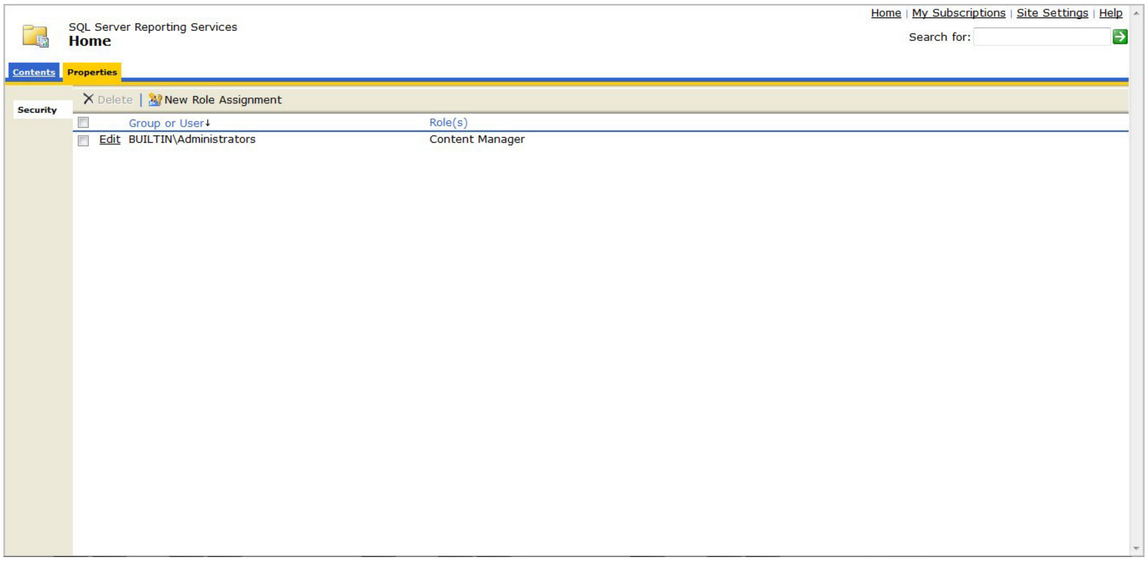
Once this logon occurs, Reporting Services utilizes the user name and the user’s group memberships to determine what rights the user possesses. The user can access only those folders and items they have rights to. In Report Manager, users do not even see the folders they cannot browse and reports they cannot run. There is no temptation for the user to try and figure out how to get into places they are not supposed to go, because they do not even know these places exist.

Local Administrator Privileges

In most cases, rights must be explicitly assigned to folders and items. One exception to this rule, however, is local administrator privileges. Any user who is a member of the local administrators group on the computer hosting the Report Server has content manager rights to all folders and all items. These automatic rights cannot be modified or removed.

Let’s look at the security page:

1. Open the Report Manager in your browser and navigate to the **Home** folder.
2. Select the **Properties** tab. You see the security page for the home folder, as shown below.


Home | My Subscriptions | Site Settings | Help
SQL Server Reporting Services
Home
Search for:
Contents
Properties
Security
Delete
New Role Assignment

		Group or User	Role(s)
□	Edit	BUILTIN\Administrators	Content Manager

The Report Server maintains a security page for each item in the Report Catalog - every folder, every report, and every supporting item. The security page lists all the role assignments for an item. Each role assignment is made up of two things: a Windows user or group and a security role. The rights associated with the security role are assigned to the Windows user or group.

Initially, one role assignment is on the security page for each item. This entry assigns the Content Manager security role to the BUILTIN\Administrators group. This entry is a reminder that any user who is a member of the local administrators group has rights to manage the contents of this folder.

NOTE: You could delete the role for BUILTIN\Administrators, and the members of the local administrators group would still have rights to manage the contents of this folder. These rights are hardwired into Reporting Services. The BUILTIN\Administrators assignment on the security pages, in most cases, just a reminder of the rights held by anyone in the local administrators group.

Tasks and Rights

You can perform a number of tasks in Reporting Services. Each task has a corresponding right to perform that task. For example, you can view reports. Therefore a corresponding right exists to view reports. The tasks within Reporting Services are shown in below in table1.

Table 1:

NameTask	NameDescription
Consume reports	Read report definitions.
Create linked reports	Create linked reports and publish them to a folder.
Manage all subscriptions	View, modify, and delete any subscription, regardless of who owns the subscription.
Manage data sources	Create, modify, and delete shared data sources.
Manage folders	Create, view, and delete folders. View and modify folder properties.
Manage individual subscriptions	Create, view, modify and delete your own subscriptions
Manage models	Create, view, and delete models. Modify model properties.
Manage report history	Create, view, and delete report history snapshots. Modify report history properties.
Manage reports	Create, view, and delete reports. Modify report properties.
Manage resources	Create, modify and delete resources. View and modify resources properties.
Set security for individual items	View and modify security settings for reports, folders, resources, and shared data sources.

Table 1:

NameTask	NameDescription
View data sources	View shared data sources and their properties.
View folders	View folders and their properties.
View models	View models. Use models as report data sources. Query models for data.
View reports	View reports and linked reports along with their report history shapshots and properties.
View resources	View resources and their properties.

You are probably not familiar with some of these tasks. We will not discuss these in this document. For now, you simply need to know these are tasks with associated rights within Reporting Services.

In addition to the tasks listed in the table, there are system-wide tasks with associated rights. These system-wide tasks deal with the management and operation of Reporting Services as a whole. The system-wide tasks within Reporting Services are shown below in table 2.

Table 2:

NameTask	NameDescription
Execute Report Definitions	Start execution of a report from a report definition without deploying it to the Report Server.
Generate events	Provide an application with the capability to generate events within the Report Server.
Manage jobs	View and cancel running Report Server jobs.
Manage Report Server properties	View and modify configuration properties for the Report Server.
Manage Report Server security	View and modify system-wide role assignments.
Manage roles	Create, view, modify, and delete role definitions.

Table 2:

NameTask	NameDescription
Manage shared schedules	Create, view, modify, and delete shared schedules used for shapshots and subscriptions.
View Report Server properties	View properties that apply to the Report Server.
View shared schedules	View a shared schedule.

Roles

The rights to perform tasks are grouped together to create *roles*. Reporting Services includes several predefined roles to help you with security management. In addition, you can create your own custom roles, grouping together any combination of rights that you like. The predefined roles and their corresponding rights are listed below.

The Browser role

The *Browser* role is the basic role assigned to users who are going to view reports, but who are not going to create folders or upload new reports. The browser role has rights to perform the following tasks:

- Manage individual subscriptions
- View folders
- View models
- View reports
- View resources

The Publisher role

The *Publisher* role is assigned to users who are going to create folders and upload reports. The Publisher role does not have rights to change security settings or manage subscriptions and report history. The Publisher role has rights to perform the following tasks:

- Create linked reports
- Manage data sources
- Manage folders
- Manage models

- Manage reports
- Manage resources

THE MY REPORTS ROLE

The *My Reports* role is designed to be used only with a special folder called the My Reports folder. Within this folder, the My Reports role gives the user rights to do everything except change security settings. the My Reports role has rights to perform the following tasks:

- Create linked reports
- Manage data sources
- Manage folders
- Manage individual subscriptions
- Manage report history
- Manage reports
- Manage resources
- View data sources
- View folders
- View reports
- View resources

THE CONTENT MANAGER ROLE

The *Content Manager* role is assigned to users who are managing the folders, reports, and resources. All members of the Windows local administrators group on the computer hosting the Report Server are automatically members of the content Manager role for all folders, reports and resources. The Content Manager has rights to perform all tasks, excluding system-wide tasks.

THE SYSTEM USER ROLE

The system-wide security tasks have two predefined roles. The *System User* role has rights to perform the following system-wide tasks:

- Execute Report Definitions
- View report server properties
- View shared schedules

THE SYSTEM ADMINISTRATOR ROLE

The *System Administrator* role provides the user with rights to complete any of the tasks necessary to manage the Report Server. All members of the Windows local administrators group on the computer hosting the Report Server are automatically members of the System Administrator role. This role has rights to perform the following system-wide tasks:

- Execute Report Definitions
- Manage jobs
- Manage report server properties
- Manage report server security
- Manage roles
- Manage shared schedules

Creating Role Assignments

As stated previously, role assignments are created when a Windows user or a Windows group is assigned a role for a folder, a report, or a resource. Role assignments are created on the security page for the folder, report or resource. These role assignments control what the user can see within a folder and what tasks the user can perform on the folder, report or resource.

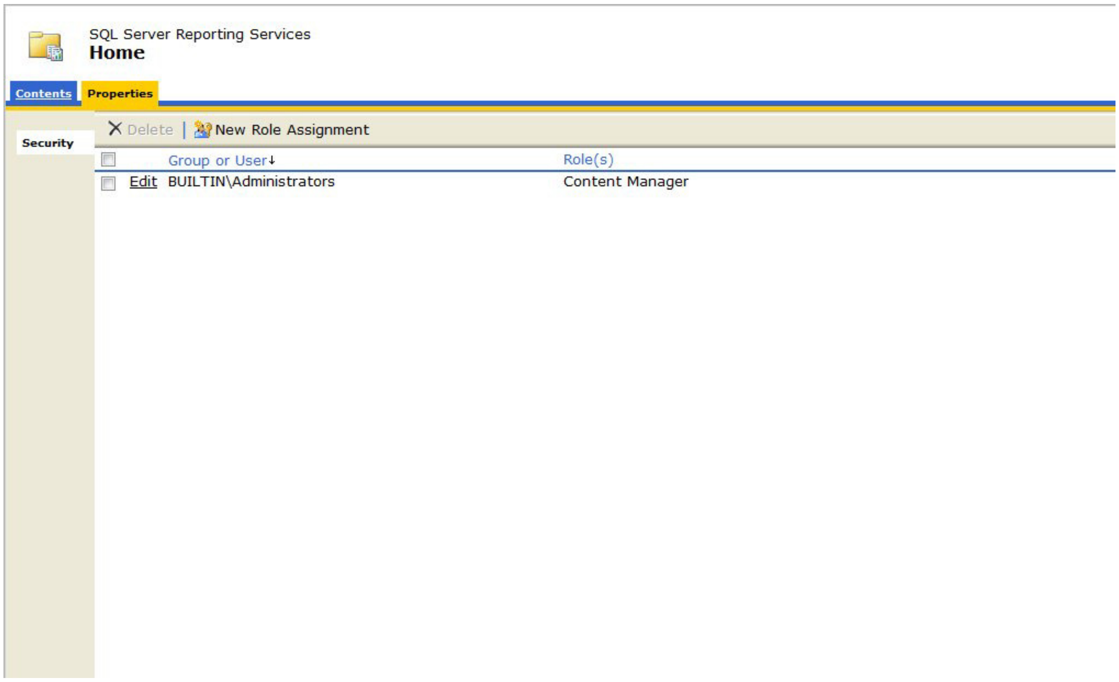
Let’s try creating role assignments for some of our folders and reports.

Creating a role assignment for a folder

Let’s try creating a new role assignment for the Home folder:

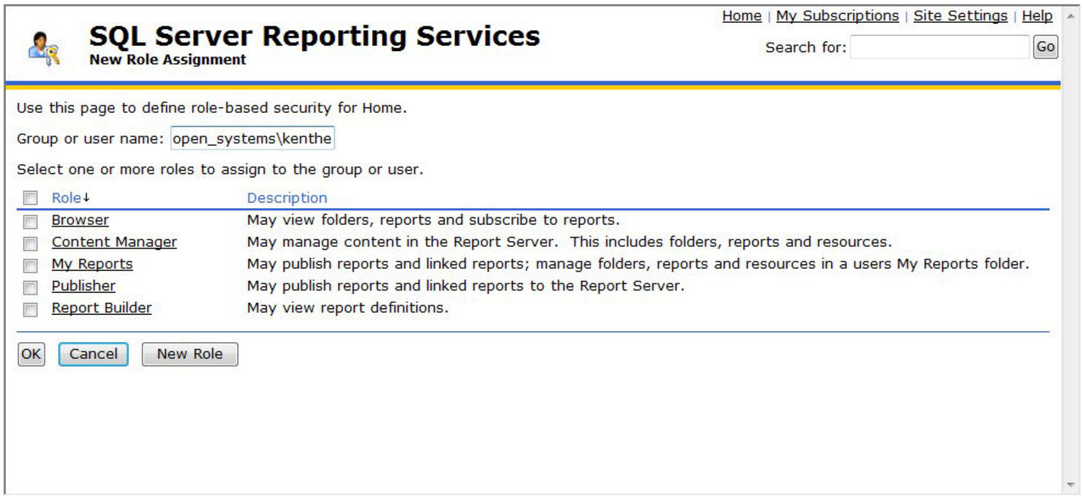
1. Open the Report Manager in your browser. You should be viewing the contents of the Home folder.

2. Select the **Properties** tab. You see the security page for this folder.



□	Group or User↓	Role(s)
□ Edit	BUILTIN\Administrators	Content Manager

3. Click **New Role Assignment**. the New Role Assignment page appears, as show below.


SQL Server Reporting Services
New Role Assignment
Home | My Subscriptions | Site Settings | Help
Search for:
Go

Use this page to define role-based security for Home.

Group or user name: open_systems\kenthe

Select one or more roles to assign to the group or user.

□ Role↓	Description
□ Browser	May view folders, reports and subscribe to reports.
□ Content Manager	May manage content in the Report Server. This includes folders, reports and resources.
□ My Reports	May publish reports and linked reports; manage folders, reports and resources in a users My Reports folder.
□ Publisher	May publish reports and linked reports to the Report Server.
□ Report Builder	May view report definitions.

4. Type the name of a valid user for **Group or User Name**. If you are using a domain user or domain group, this must be in the format DomainName\UserName or DomainName\GroupName. If you are using a local user or local group, this must be in the format ComputerName\UserName or ComputerName\GroupName.

5. Check the box for the **Role** you want to assign for this user. (Browser for example.)

6. Click **OK** to save your role assignment and return to the security page. Reporting Services checks to ensure you entered a valid user or group for the role assignment. If this is not a valid user or group, you receive an error message and our role assignment is not saved.

NOTE: A user needs to have at least viewing rights in the Home folder to view other folders and navigate to them.

Inherited role assignments

By default, folders (other than the Home folder), reports and resources inherit their role assignments from the folder that contains them. You can think of the nested folders as branches of a tree, with the reports and resources as the leaves. Inherited security means, if you make security changes to one folder and have those changes take effect for all branches and leaves further along the tree.

This makes managing security easy. You can maintain security for all the reports and resources within a folder simply by modifying the role assignments for the folder itself. You can maintain security for an entire branch of the tree structure by modifying the role assignments for the folder that forms the base of that branch. Let’s look at the security for the TRAVERSE Reports folder.

1. Select the **Contents** tab of the **Home** folder.

2. Select the **TRAVERSE Reports** folder to view its contents.

3. Select the **Properties** tab. You see the properties page for this folder.

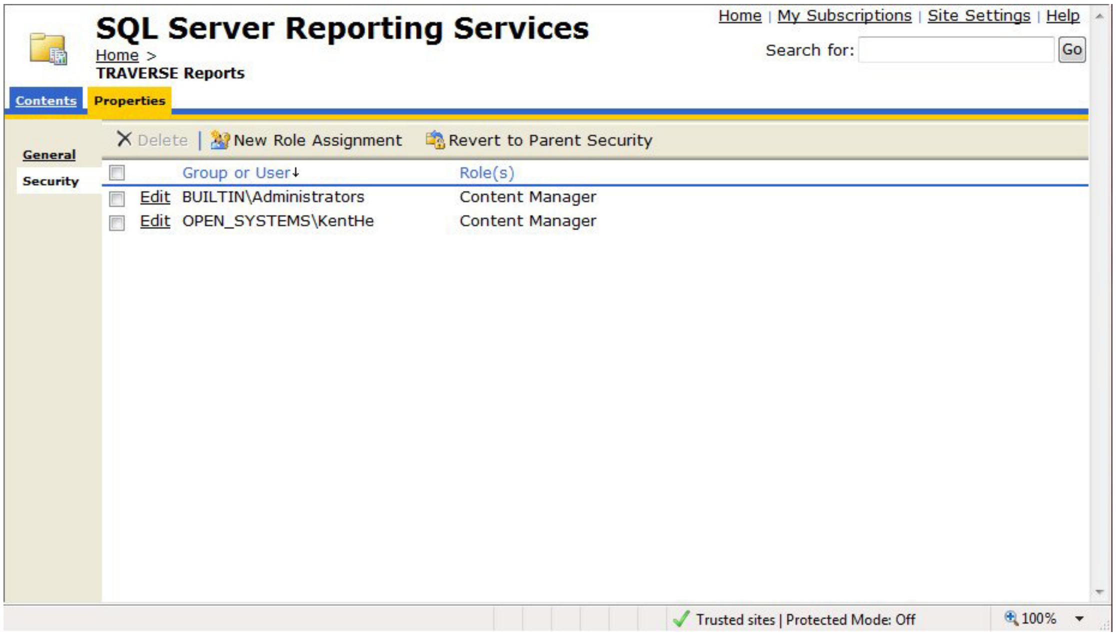
4. Select **Security** from the left side of the page. You see the security page for this folder.

The TRAVERSE Reports folder is inheriting its role assignments from the Home folder. You did not add a role assignment giving Browser rights to your user in this folder and, yet there it is! As soon as you added the role assignment to the Home folder, it appeared for all items within the Home folder.

You gave your user Browser rights in the Home folder, so they could view the contents of the Home folder, and then navigate into other folders to find the reports they need. You may want to give this user additional rights in folders further along in the tree. Perhaps the user can manage the content of certain folders that belong to their department, but can only browse when in the Home folder.

To accomplish this task, you must first break the inherited security for the TRAVERSE Reports folder:

1. Click **Edit Item Security**. A dialog box with an inherited security message appears. The Report Manager is confirming you want to break that inheritance by creating your own role assignments for this folder.


SQL Server Reporting Services
Home >
TRAVERSE Reports
Home | My Subscriptions | Site Settings | Help
Search for:
Go
Contents
Properties
General
Security
Delete
New Role Assignment
Revert to Parent Security
Trusted sites | Protected Mode: Off
100%

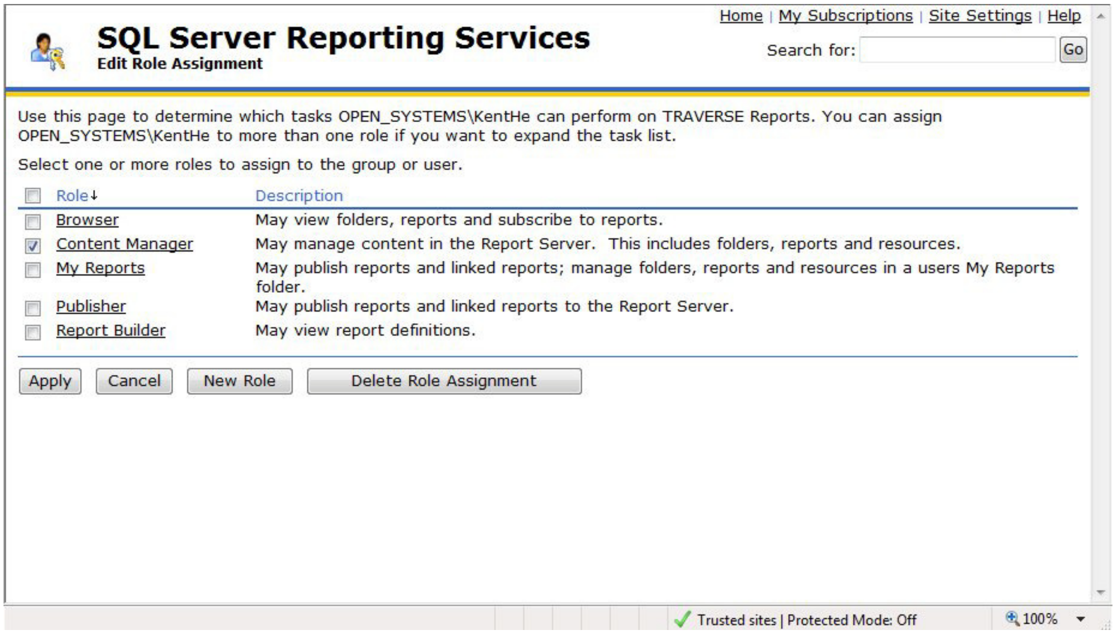
		Group or User↓	Role(s)
□	Edit	BUILTIN\Administrators	Content Manager
□	Edit	OPEN_SYSTEMS\KentHe	Content Manager

2. Click **OK** to confirm you want to break the inherited security.

Now that you have broken the inherited security, you have new buttons on the toolbar for adding a new role assignment, deleting existing role assignments, and reverting to inherited security.

Now you can edit the role assignment for your user:

1. Click the **Edit** link next to the role assignment giving your user Browser rights. The **Edit Role Assignment** page appears.



SQL Server Reporting Services
Edit Role Assignment

Use this page to determine which tasks OPEN_SYSTEMS\KentHe can perform on TRAVERSE Reports. You can assign OPEN_SYSTEMS\KentHe to more than one role if you want to expand the task list.

Select one or more roles to assign to the group or user.

□ Role↓	Description
□ Browser	May view folders, reports and subscribe to reports.
☑ Content Manager	May manage content in the Report Server. This includes folders, reports and resources.
□ My Reports	May publish reports and linked reports; manage folders, reports and resources in a users My Reports folder.
□ Publisher	May publish reports and linked reports to the Report Server.
□ Report Builder	May view report definitions.

2. Uncheck the check box for the **Browser** role.

3. Check the check box for the **Content Manager** role.

4. Click **Apply** to save the changes to your role assignment and return to the security page. The user now has Content Manager rights to the **TRAVERSE Reports** folder.

5. Click the **Contents** tab.

6. Select the **History Reports** folder to view its content.

7. Select the **Properties** tab. You see the properties page for this folder.

8. Select **Security** from the left side of the page. You see the security page for this folder.

You can see the History Reports folder is inheriting its role assignments from the TRAVERSE Reports folder.

NOTE: Although we do not do so in these exercises, you can check more than one role when creating or editing a role assignment. The user's rights are then a sum of the rights granted by each role.

Role assignments using windows groups

As mentioned previously, role assignments can be made to Windows users or to Windows groups. If you create your role assignments using Windows users, you need to create a new set of role assignments every time a new user needs to access Reporting Services. This can be extremely tedious if you have a complex set of role assignments for various folders, reports and resources.

In most cases, creating role assignments using Windows groups is better. Then, as new users come along, you simply need to add them to the Windows group that has the appropriate rights in Reporting Services. This is much easier!

NOTE: In some cases, Internet Information Services (IIS) and, therefore, Reporting SErvices do not immediately recognize changes to group membership. This is because IIS caches some Windows security information, and then works from that cache. Stopping and starting the IIS service causes the IIS security cache to be reloaded with the latest greatest group membership information.

TRAVERSE VIEWING

5

USING TRAVERSE TOOLS

Overview

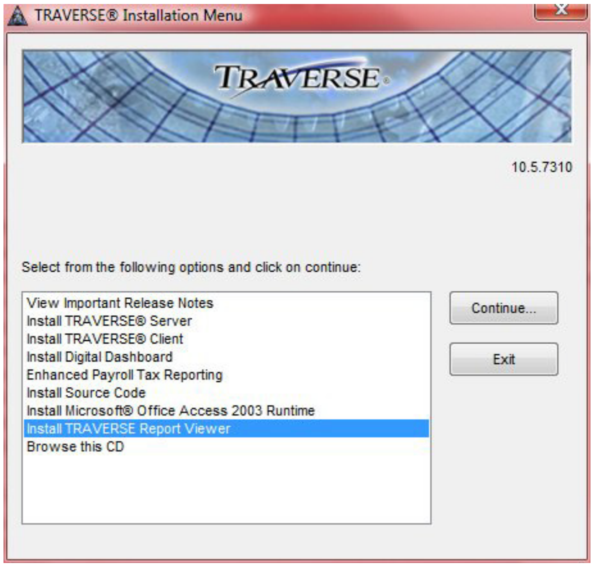
We have see how to make reports and deploy them to the Report Manager. Now it is time to see how we can view these reports using some tools in TRAVERSE.

We will first look at using the 10.5 TRAVERSE Report Viewer to see the reports we have made using the Reporting Services.

We will then see how to add the Reporting Services reports to the TRAVERSE menu and view them through the web based report viewer.

Installing TRAVERSE Report Viewer (10.5 only)

1. To install TRAVERSE Report viewer, put the TRAVERSE CD in the CD drive and the CD Menu should appear.



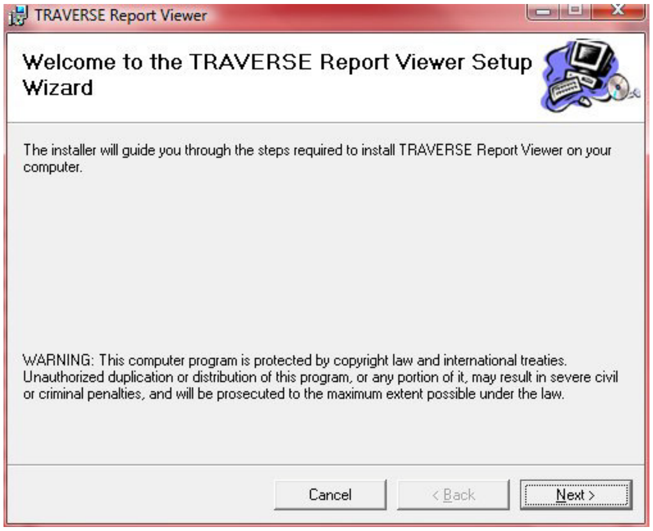
10.5.7310

Select from the following options and click on continue:

View Important Release Notes
Install TRAVERSE® Server
Install TRAVERSE® Client
Install Digital Dashboard
Enhanced Payroll Tax Reporting
Install Source Code
Install Microsoft® Office Access 2003 Runtime
Install TRAVERSE Report Viewer
Browse this CD

2. Highlight **Install TRAVERSE Report Viewer** and click **Continue**.

3. The **Welcome** screen appears.



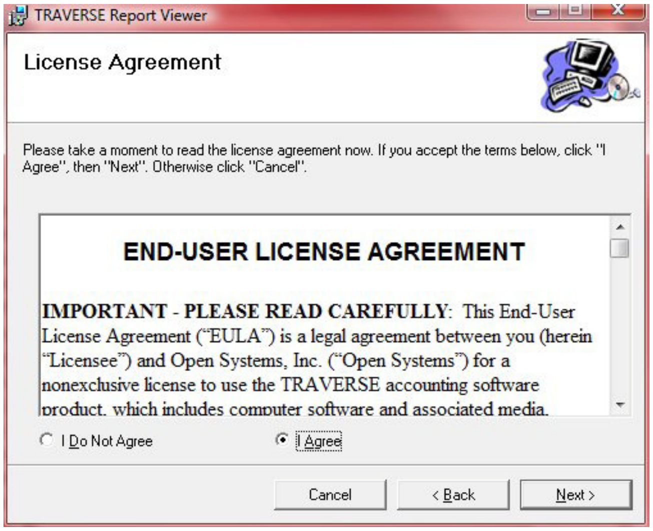
Welcome to the TRAVERSE Report Viewer Setup Wizard

The installer will guide you through the steps required to install TRAVERSE Report Viewer on your computer.

WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.

4. Click on **Next** to continue with the installation.

5. The **License Agreement** screen appears.



License Agreement

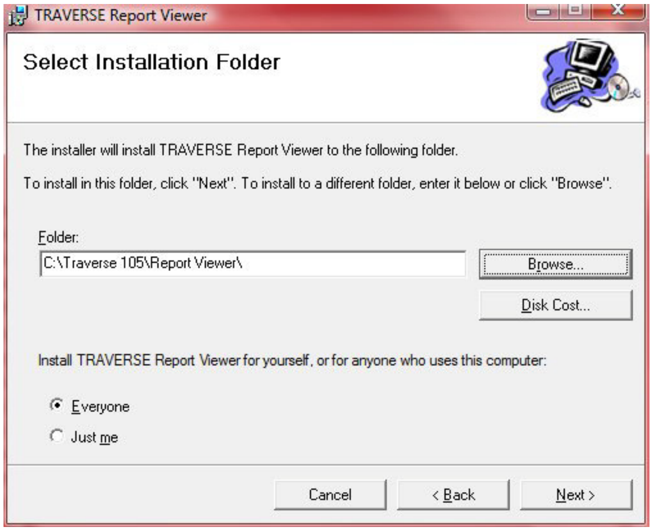
Please take a moment to read the license agreement now. If you accept the terms below, click "I Agree", then "Next". Otherwise click "Cancel".

END-USER LICENSE AGREEMENT

IMPORTANT - PLEASE READ CAREFULLY: This End-User License Agreement ("EULA") is a legal agreement between you (herein "Licensee") and Open Systems, Inc. ("Open Systems") for a nonexclusive license to use the TRAVERSE accounting software product, which includes computer software and associated media.

○ I Do Not Agree ● I Agree

6. Click **Next** to continue the installation. The **Select Installation Folder** screen appears.



TRAVERSE Report Viewer

Select Installation Folder

The installer will install TRAVERSE Report Viewer to the following folder.

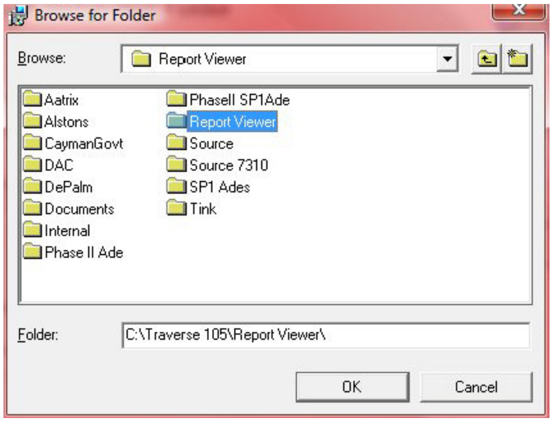
To install in this folder, click "Next". To install to a different folder, enter it below or click "Browse".

Folder: C:\Traverse 105\Report Viewer\

Install TRAVERSE Report Viewer for yourself, or for anyone who uses this computer:

● Everyone
○ Just me

7. Choose the directory where you want to install the Report Viewer. Choose who will have the menu option to run the Viewer. Click **Next** to continue.

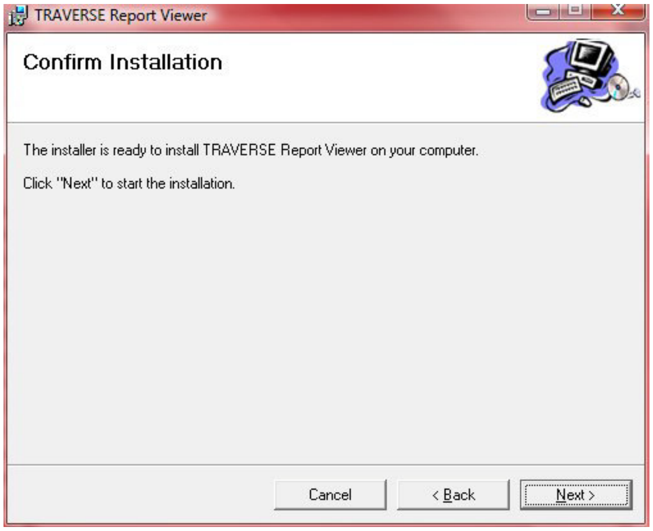


Browse for Folder

Browse: Report Viewer

Folder: C:\Traverse 105\Report Viewer\

8. The **Confirm Installation** screen appears.



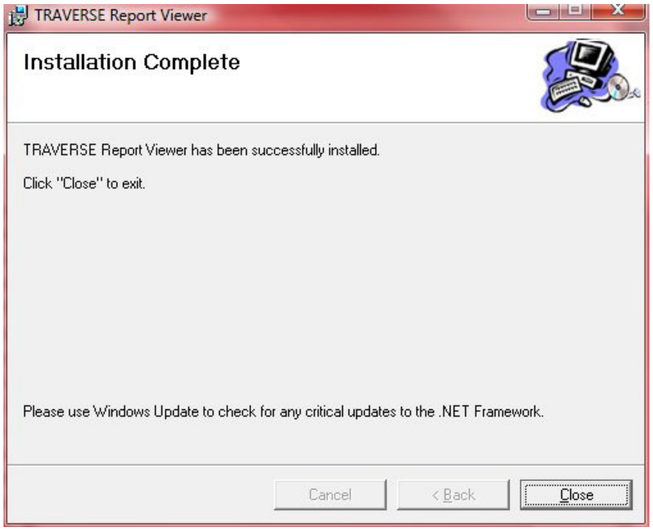
TRAVERSE Report Viewer

Confirm Installation

The installer is ready to install TRAVERSE Report Viewer on your computer.

Click "Next" to start the installation.

9. Click **Next** to Install the Report Viewer.



TRAVERSE Report Viewer

Installation Complete

TRAVERSE Report Viewer has been successfully installed.

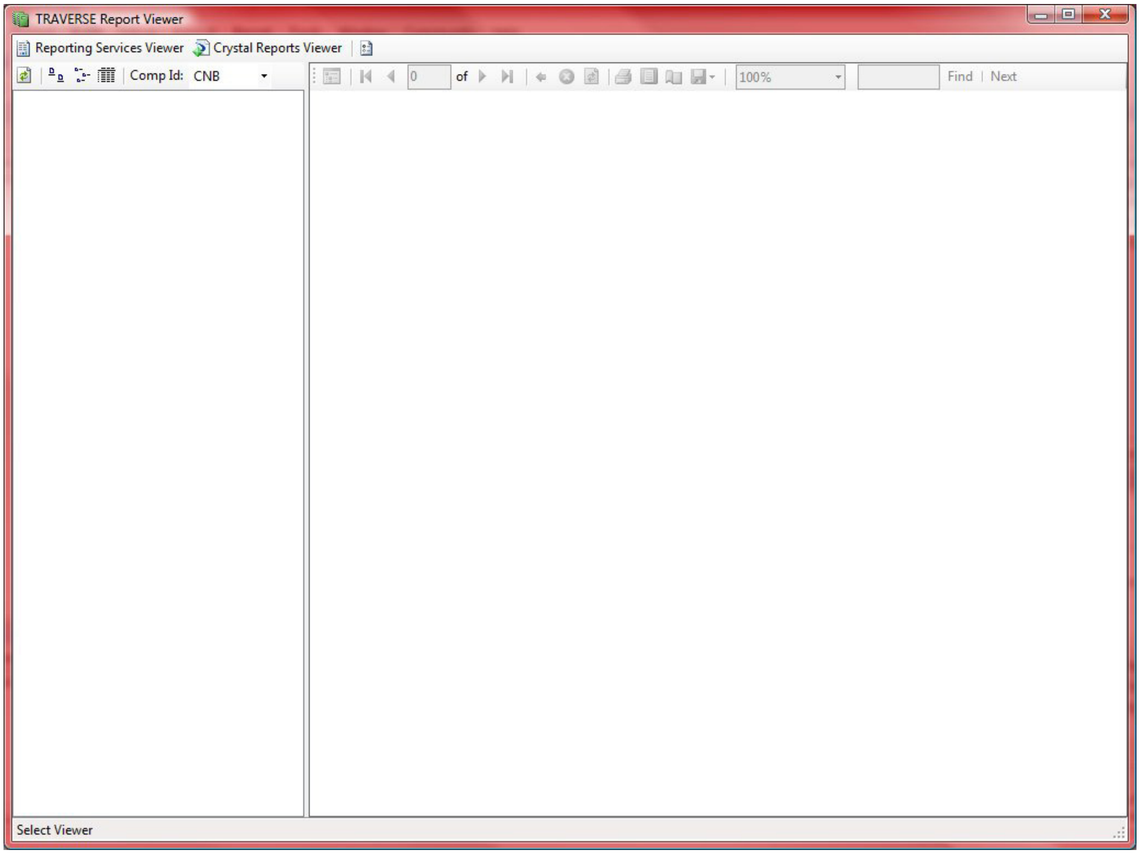
Click "Close" to exit.

Please use Windows Update to check for any critical updates to the .NET Framework.

10. When the installation is complete, click **Close** to exit the Wizard.

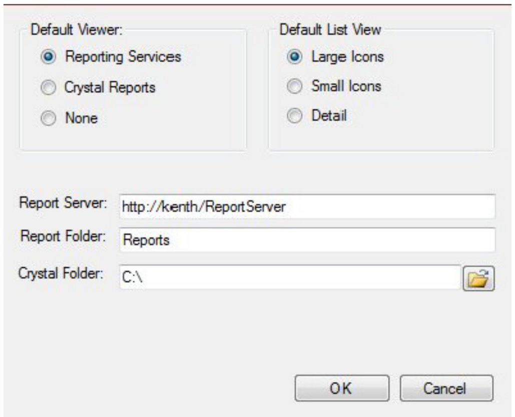
11. To configure the TRAVERSE Report Viewer, go to **Start** – **Programs** – **TRAVERSE** – **TRAVERSE Report Viewer**.

12. Login to the TRAVERSE Server and click **OK**. This will bring up a blank viewer the first time by default.



13. Click on the **Preferences** button.

14. Default Viewer: Set the preferences for how you want the viewer to open from now on.



Default Viewer:
- ● Reporting Services
- ○ Crystal Reports
- ○ None

Default List View
- ● Large Icons
- ○ Small Icons
- ○ Detail

Report Server: http://kenth/ReportServer

Report Folder: Reports

Crystal Folder: C:\

15. Default List View: Set what type of view you want to see from now on.

16. Report Server: based on a default installation, the URL should be http://machinename/ReportServer.

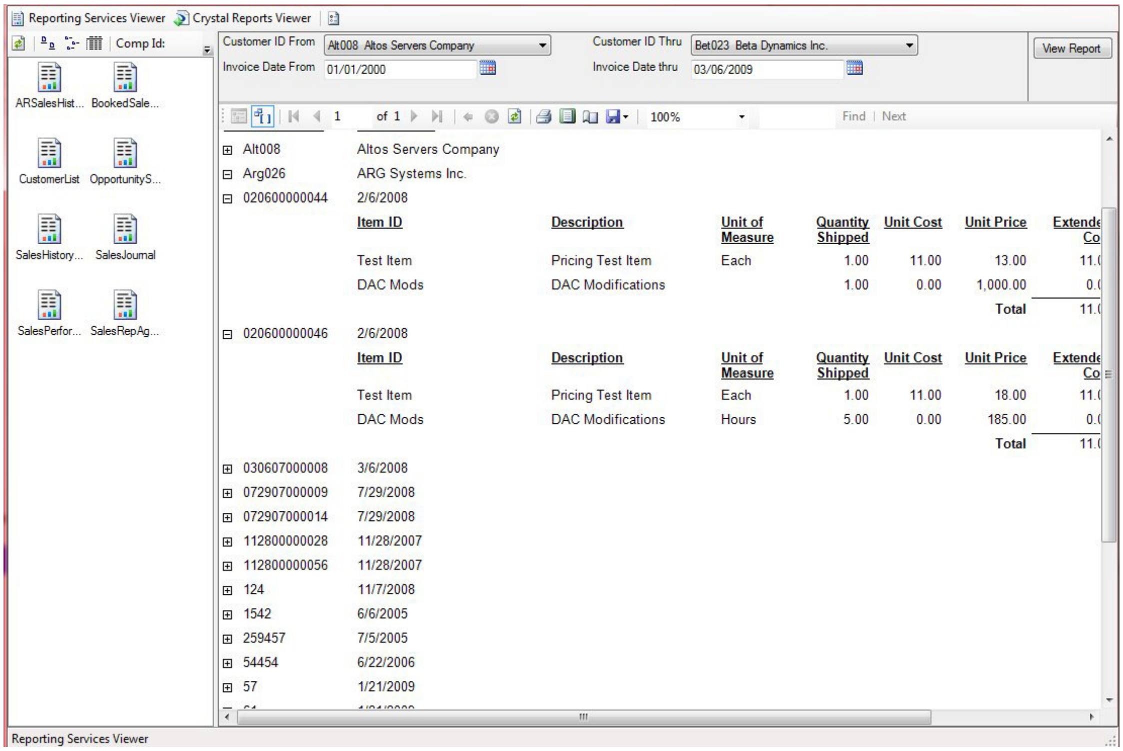
17. Report Folder: this should be the folder name you assigned in both the Report Manager Virtual Directory and the folder you created in IE earlier. By default this is named Reports.

18. Crystal Folder: this can be left blank, or if you are using the viewer to view Crystal Reports, enter the shared network directory where those reports are stored.

NOTE: If using the viewer for only Crystal Reports, that is the only option that needs to be set. You also do not need to perform any of the earlier configuration steps, other than installing the Viewer.

19. Click **OK** to close Preferences. Click on Reporting Services Viewer to start the viewer and view the reports you just added.

20. You can now double click on a report to view it.



Customer ID From: Alt008 Altos Servers Company
Customer ID Thru: Bet023 Beta Dynamics Inc.
Invoice Date From: 01/01/2000
Invoice Date thru: 03/06/2009

⊞ Alt008	Altos Servers Company							
⊟ Arg026	ARG Systems Inc.							
⊟ 020600000044	2/6/2008							
	Item ID	Description	Unit of Measure	Quantity Shipped	Unit Cost	Unit Price	Extende Co
	Test Item	Pricing Test Item	Each	1.00	11.00	13.00	11.0
	DAC Mods	DAC Modifications		1.00	0.00	1,000.00	0.0
						Total	11.0
⊟ 020600000046	2/6/2008						
	Item ID	Description	Unit of Measure	Quantity Shipped	Unit Cost	Unit Price	Extende Co
	Test Item	Pricing Test Item	Each	1.00	11.00	18.00	11.0
	DAC Mods	DAC Modifications	Hours	5.00	0.00	185.00	0.0
						Total	11.0
⊞ 030607000008	3/6/2008						
⊞ 072907000009	7/29/2008						
⊞ 072907000014	7/29/2008						
⊞ 112800000028	11/28/2007						
⊞ 112800000056	11/28/2007						
⊞ 124	11/7/2008						
⊞ 1542	6/6/2005						
⊞ 259457	7/5/2005						
⊞ 54454	6/22/2006						
⊞ 57	1/21/2009						

Adding a Reporting Services report to the TRAVERSE Menu

You can add your Reporting Services report to your TRAVERSE menu by using the tools in System Manager to update your menu.

To add your Reporting Services reports to the TRAVERSE Menu follow these steps:

1. Open your Report Server web site from your report manager setup.

   http://ComputerName/ReportServer

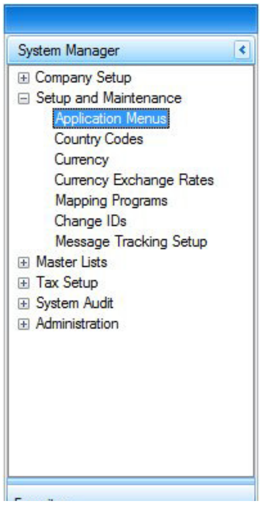
   Where ComputerName is the name of the computer that has report manager configured on it, and ReportServer as the path your report server is set up.

2. Browse to the report you want to add to your menu and view the report.

3. Copy the URL path from the path field in your browser. An example of the path would be as follows:

http://ComputerName/ReportServer/Pages/Report-Viewer.aspx?%2fTRAVERSE+Reports%2fCustList&rs:Command=Render

4. Open your TRAVERSE menu and go to **System Manager** - **Setup and Maintenance** - **Application Menus**.



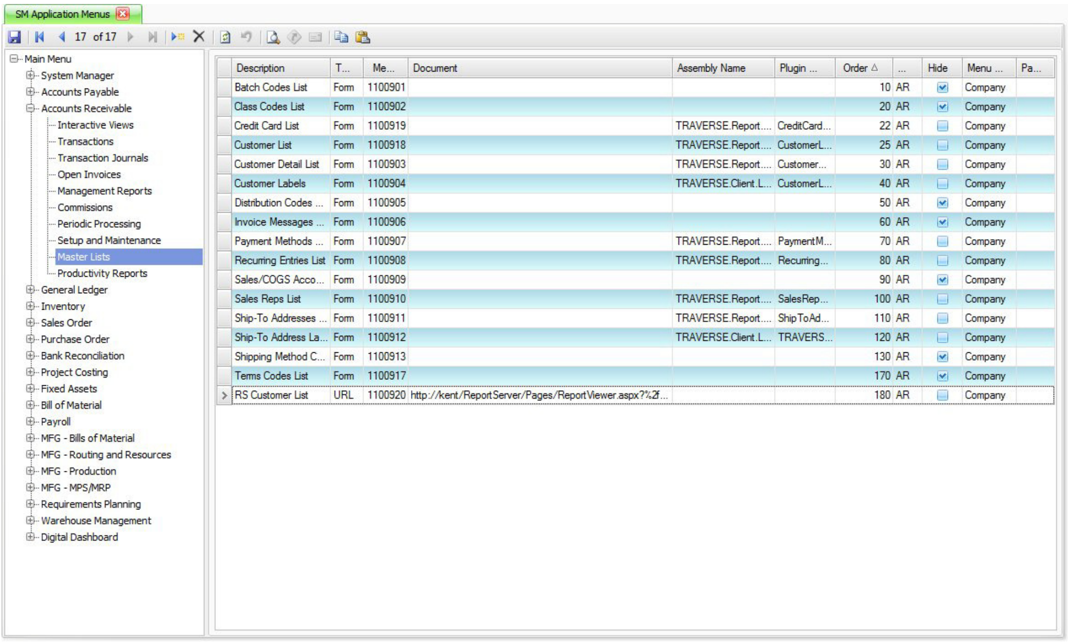
5. When the **Application Menus** screen opens expand **Accounts Receivable**, click the plus (+) sign, expand the second level menu, for example Master Lists. Enter the name of the report the way you want it to appear on your TRAVERSE menu.

6. Select **URL** as the **Type** of menu to use.

7. Paste the URL path you copied from step 3 above into the **Object** field.

8. Enter a Menu **Order** number to place your report where you want it on your menu.

9. Tab through the remaining fields to accept the defaults.



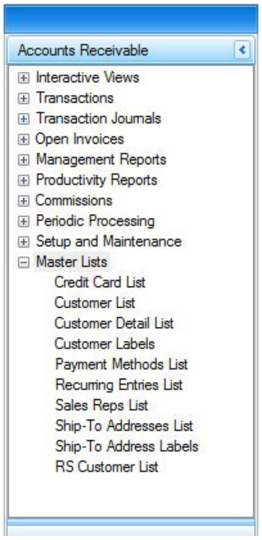
Description	T...	Me...	Document	Assembly Name	Plugin ...	Order △	...	Hide	Menu ...	Pa...
Batch Codes List	Form	1100901				10	AR	☑	Company	
Class Codes List	Form	1100902				20	AR	☑	Company	
Credit Card List	Form	1100919		TRAVERSE.Report....	CreditCard...	22	AR	□	Company	
Customer List	Form	1100918		TRAVERSE.Report....	CustomerL...	25	AR	□	Company	
Customer Detail List	Form	1100903		TRAVERSE.Report....	Customer...	30	AR	□	Company	
Customer Labels	Form	1100904		TRAVERSE.Client.L...	CustomerL...	40	AR	□	Company	
Distribution Codes ...	Form	1100905				50	AR	☑	Company	
Invoice Messages ...	Form	1100906				60	AR	☑	Company	
Payment Methods ...	Form	1100907		TRAVERSE.Report....	PaymentM...	70	AR	□	Company	
Recurring Entries List	Form	1100908		TRAVERSE.Report....	Recurring...	80	AR	□	Company	
Sales/COGS Acco...	Form	1100909				90	AR	☑	Company	
Sales Reps List	Form	1100910		TRAVERSE.Report....	SalesRep...	100	AR	□	Company	
Ship-To Addresses ...	Form	1100911		TRAVERSE.Report....	ShipToAd...	110	AR	□	Company	
Ship-To Address La...	Form	1100912		TRAVERSE.Client.L...	TRAVERS...	120	AR	□	Company	
Shipping Method C...	Form	1100913				130	AR	☑	Company	
Terms Codes List	Form	1100917				170	AR	☑	Company	
RS Customer List	URL	1100920	http://kent/ReportServer/Pages/ReportViewer.aspx?%2f...			180	AR	□	Company	

10. Click the Save button.

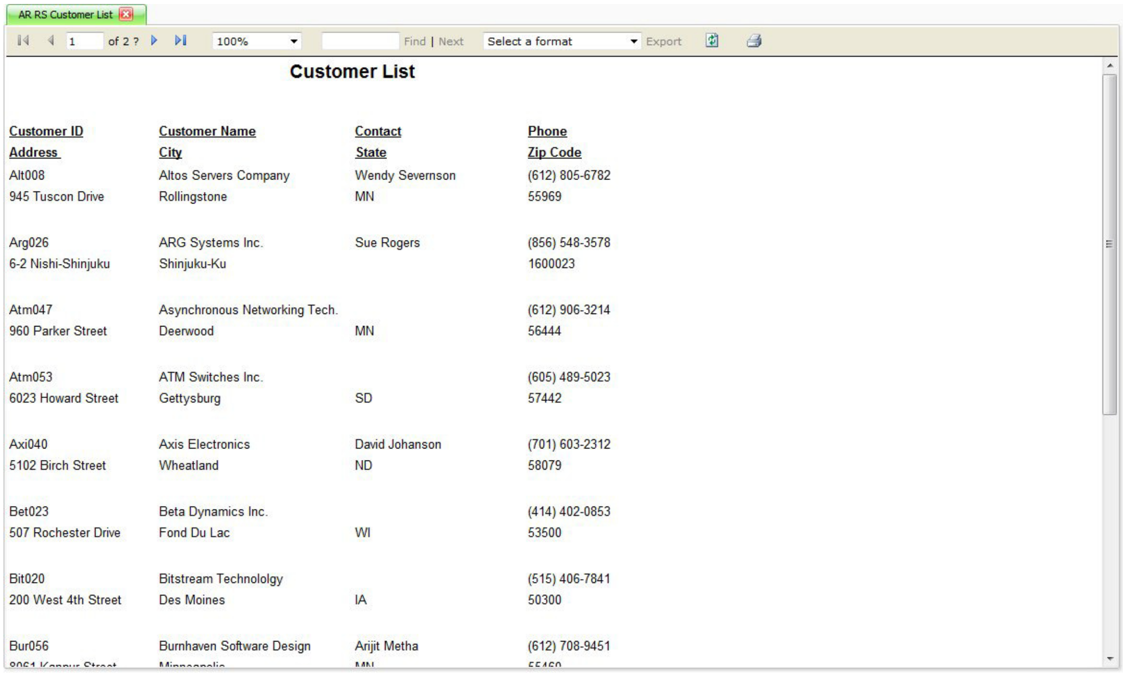
11. **Close** the Application Menus screen to return to the main menu.

12. Use Server Manager to set permissions on your new menu items to those groups you want to be able to print the report.

13. Your new menu should now appear in the menu you placed it.



14. Click your menu selection to view your report in a new tab just as you would if you went to the ReportServer site from your browser.



Customer List

Customer ID	Customer Name	Contact	Phone
Address	City	State	Zip Code
Alt008	Altos Servers Company	Wendy Severnson	(612) 805-6782
945 Tuscon Drive	Rollingstone	MN	55969
Arg026	ARG Systems Inc.	Sue Rogers	(856) 548-3578
6-2 Nishi-Shinjuku	Shinjuku-Ku		1600023
Atm047	Asynchronous Networking Tech.		(612) 906-3214
960 Parker Street	Deerwood	MN	56444
Atm053	ATM Switches Inc.		(605) 489-5023
6023 Howard Street	Gettysburg	SD	57442
Axi040	Axis Electronics	David Johanson	(701) 603-2312
5102 Birch Street	Wheatland	ND	58079
Bet023	Beta Dynamics Inc.		(414) 402-0853
507 Rochester Drive	Fond Du Lac	WI	53500
Bit020	Bitstream Technololgy		(515) 406-7841
200 West 4th Street	Des Moines	IA	50300
Bur056	Burnhaven Software Design	Arijit Metha	(612) 708-9451
8061 Kanpur Street	Minneapolis	MN	55460

A

ACKNOWLEDGEMENTS

Microsoft SQL Server 2005 Reporting Services

Author: Brian Larson MCSD, MCDBA

McGraw Hill Osborne

Visual Studio 2005 Help Online

SQL Server 2005 Online Books and Help Online

A

ACKNOWLEDGEMENTS