

SedonaOffice

13TH ANNUAL USERS CONFERENCE

January 12-14, 2015 | Coronado Bay, CA

SQL Reporting Services

Presented By:

Jim Mayes

Table of Contents

Overview.....	4
Definitions & Terms	4
Data Mining	4
Data Warehouse	4
GUI (Graphical User Interface)	4
SSL Encryption (Secure Sockets Layer)	4
Conventions Used:.....	4
Developing and Accessing Business Intelligence	5
So Much Data	5
Seeing Things Your Way	5
Introducing SQL Server Reporting Services	5
Reporting Services vs. Crystal Reports	5
Easy to Use	5
Safe	5
Secure	5
100% Compatible.....	6
Improved Access to Information	6
Installing Reporting Services.....	6
Practical Applications of Reporting Services	8
The Service Ticket Listing.....	8
Tables Used	8
Loading Visual Studio 2008 and Creating a New Project.....	8
Start Reporting Services	8
Run the Report Wizard	9
Generate the base query for the Service Ticket List.....	10
Adding Tables:	11
Format the Report	15
Review and Adjust the Report:.....	16
Service Ticket Costing Report	17
Tables Used	17
Create the Costing Query	18
Creating the Report	19
Improving the Format.....	20
Adding Service Company and Service Level	21
Enabling Drill Down	23

Reporting on Totals Only..... 25

The Final Analysis..... 26

 What Have You Learned? 26

 Where Do You Go From Here? 26

 Extending This to Other Tables..... 26

 Books and Learning Materials 26

 Online Assistance..... 26

 Getting Reporting Services Up and Running..... 27

 Training..... 27

Final Word 27

Appendix..... 28

 Formatting Conventions for Dates 28

 Service Ticket Costing Query 29

Overview

The purpose of this guide is to provide the user with an introduction to Microsoft SQL Server Reporting Services and demonstrate how it can be used to extract meaningful data from the SedonaOffice database. This guide will review the many features of Reporting Services and then use a number of examples to illustrate the value of this tool used in conjunction with SedonaOffice.

Definitions & Terms

Data Mining

The automatic extraction of useful, often previously unknown information from large databases or data sets.

Data Warehouse

A system for storing, retrieving and managing large amounts of data without worrying about slowing down day-to-day operations of the production database. This can include sophisticated techniques for fast searches, as well as advanced filtering.

GUI (Graphical User Interface)

Designed to standardize and simplify the use of computer programs, as by using a mouse to manipulate text and images on a display screen featuring icons, windows, and menus.

SSL Encryption (Secure Sockets Layer)

A protocol to provide secure communications over the Internet using asymmetric key encryption. SSL is layered beneath application protocols such as HTTP, SMTP, Telnet, FTP, Gopher and NNTP and is layered above the connection protocol TCP/IP. It is used by the HTTPS access method.

Conventions Used:

Terms enclosed by "" indicate literal expressions.

Type "test" means to type T E S T, not " T E S T "

A word in [BLUE](#) indicates a mouse button or selection.

Click [New](#) indicates to click on a button labeled New or select New from a drop-down.

Developing and Accessing Business Intelligence

So Much Data

SedonaOffice stores an enormous amount of information about your business. Even though there are hundreds of reports with myriad options and filters each, you only have access to a fraction of that information.

Seeing Things Your Way

Of those hundreds of reports, do all of them fit your organization? SedonaOffice is installed in hundreds of companies with thousands of users, and cannot be tailored to each company or user. Perhaps the information is spot on, but the formatting is too generic. Maybe you would like to add a calculation or some additional data. You may even want an entirely new report completely tailored to your specific needs. You could commission custom reporting through SedonaOffice's highly trained development team, contract with a third party, or build reports yourself using Reporting Services.

Introducing SQL Server Reporting Services

You can design and develop your own reports tailored to your needs using Reporting Services. When you buy SQL Server, Microsoft throws in Reporting Services free of charge! Do not let the price fool you. Reporting Services is a fully featured, well developed product that could easily stand on its own.

Reporting Services vs. Crystal Reports

The web contains numerous comparisons of Reporting Services to Crystal Reports and I would encourage prospective users to visit some of those sites. In general Reporting Services 2005 is considered weaker than Crystal, but Reporting Services 2008 compares favorably and many users/critics prefer it!

Easy to Use

Reporting Services is designed to be intuitive and offers report design through a reporting wizard as well as a nuts and bolts approach. Most of the input is through a GUI and you do not need to understand SQL language to design and produce effective reports.

Safe

All data access in Reporting Services is "Read-Only" which means you cannot change or corrupt your data. Try any combination of tables and design any report without fear.

Secure

Access to the report designer and to the reports themselves is controlled through Windows User Group Security. In addition, all data communications including via web can be secured via SSL encryption.

100% Compatible

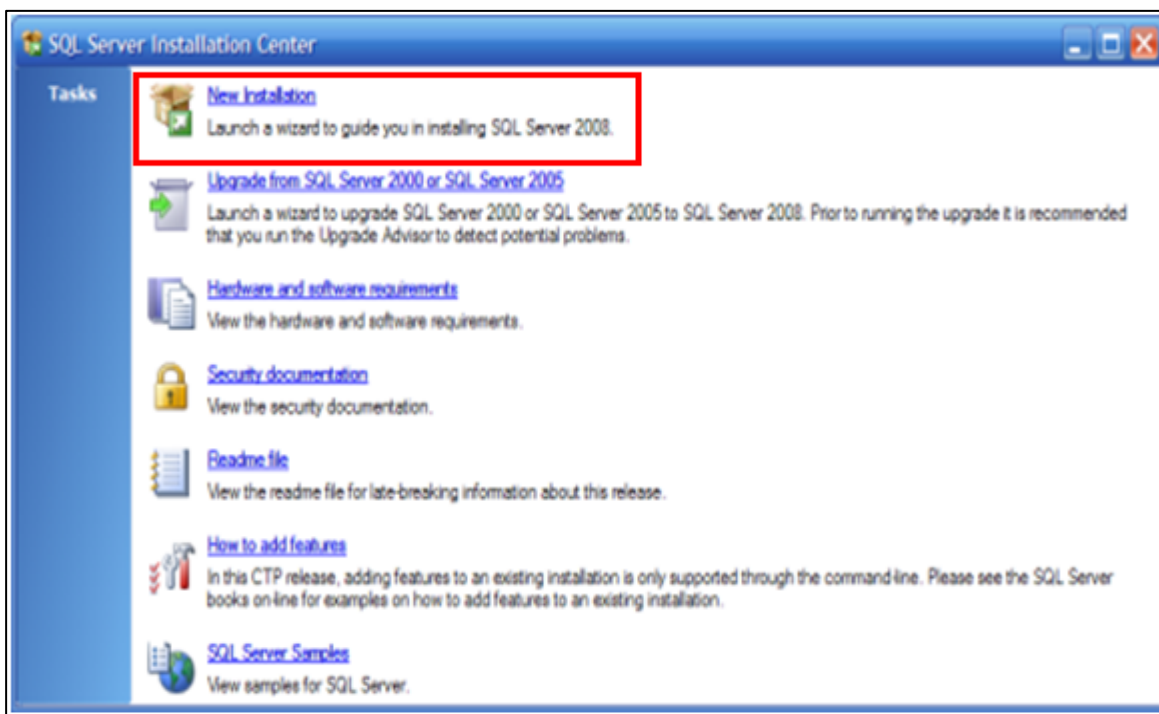
Reporting Services is 100% compatible with both SQL and SedonaOffice. As Reporting Services is native to SQL Server you can be confident that it will always be supported within SQL Server. SedonaOffice is rooted in SQL Server and therefore you can also be confident that Reporting Services will always work with SedonaOffice's database.

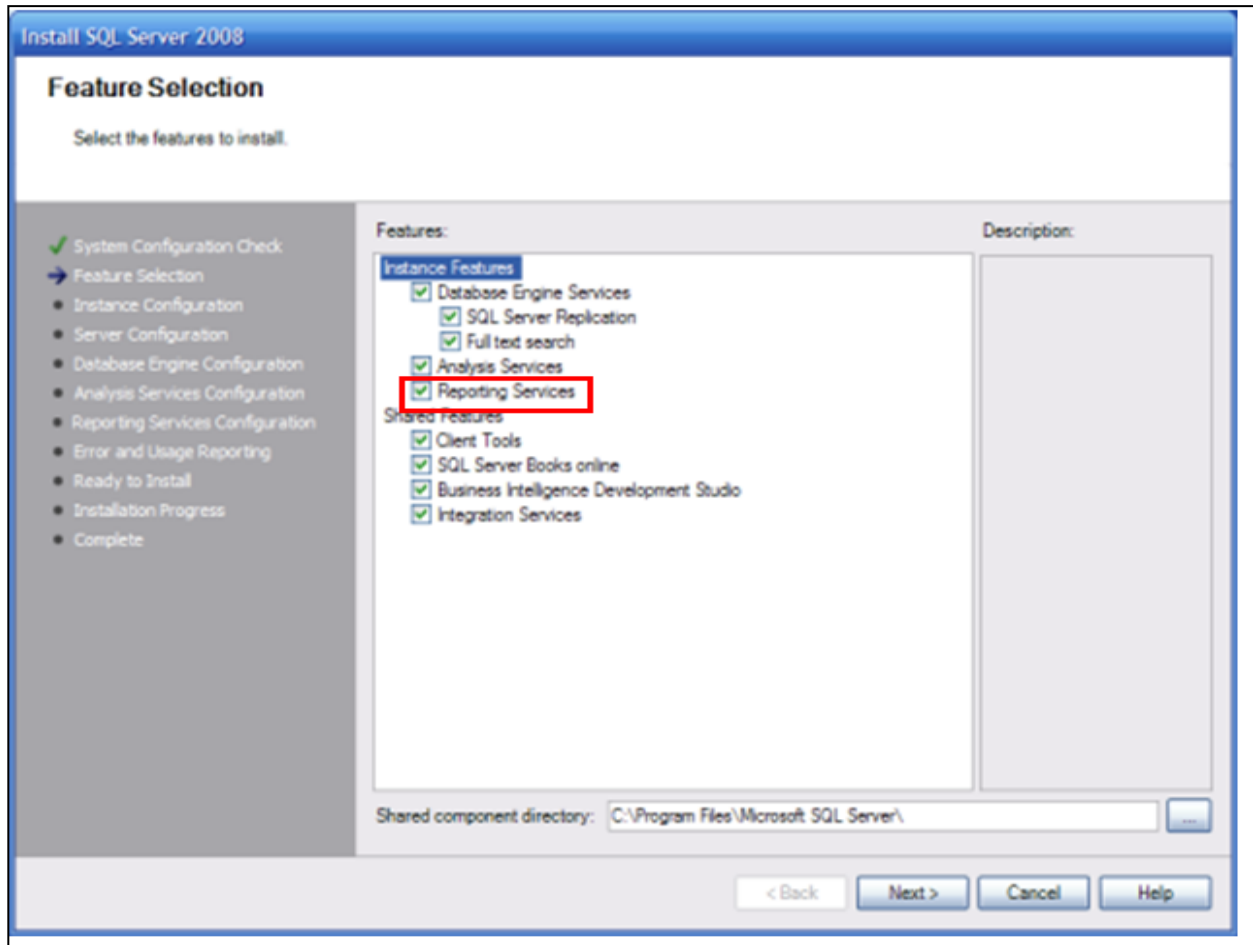
Improved Access to Information

Reporting Services gives you numerous options for making reports available to your users as well as actively distributing reports via email. Once a report is published, you can make it available for access throughout your company to anyone connected to the report server and who has the appropriate authorization. You can attach the report to a Web Page for display or download. Finally, you can configure reports to run automatically and then send the results to any number of people via an email list.

Installing Reporting Services

You already own Reporting Services, but that is not the same as having it installed. As with many Windows components, you had to select to install reporting services during the installation of SQL Server. The installation and configuration of reporting services as well as the set-up of a report server is beyond the scope of this class. However, you can begin the process by determining if Reporting Services is already installed. On your SQL Server Machine, navigate to "Add/Remove Programs" in the Control Panel: [Start → Control Panel → Add or Remove Programs → Microsoft SQL Server 2008 → Change/Remove → Add]. You will need to have your SQL Server Installation Disk available.





Notes

Practical Applications of Reporting Services

The Service Ticket Listing

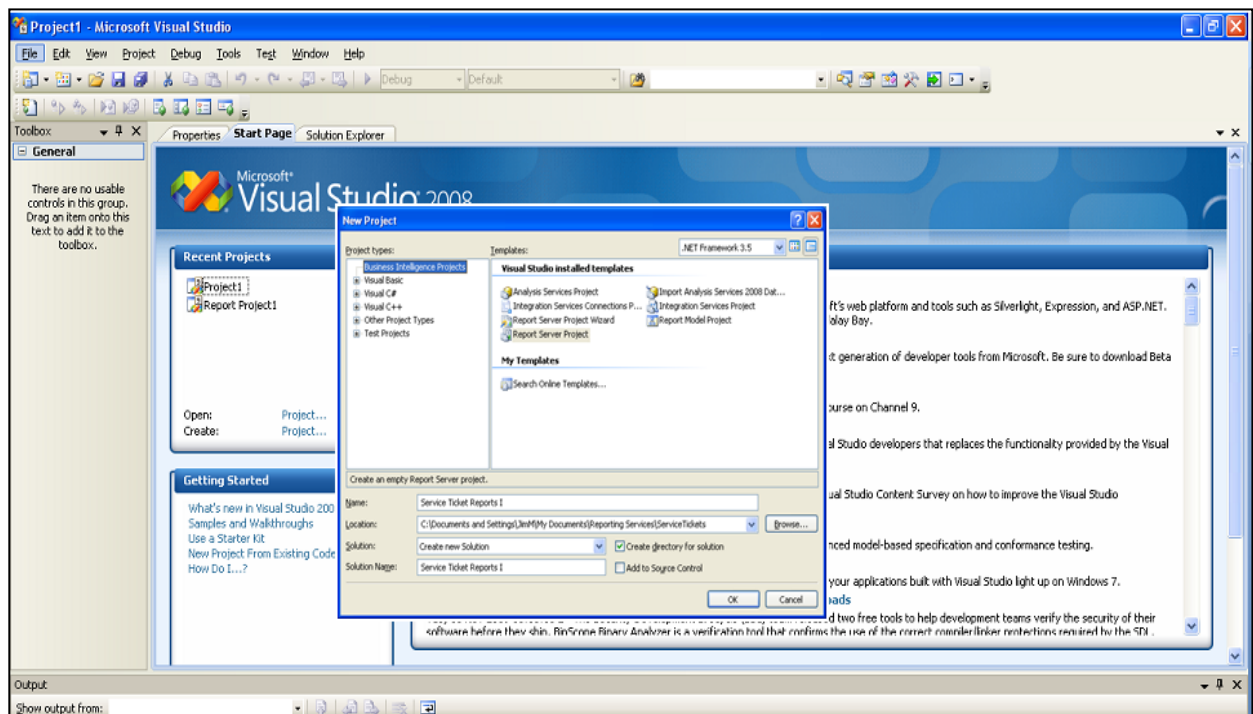
Tables Used

SV_Service_Ticket *SV_Service_Tech* *SY_Employee* *SV_Service_Company*
AR_Customer *AR_Customer_Site* *SV_Problem* *AR_Customer_System*
SV_Resolution *SV_Service_Level*

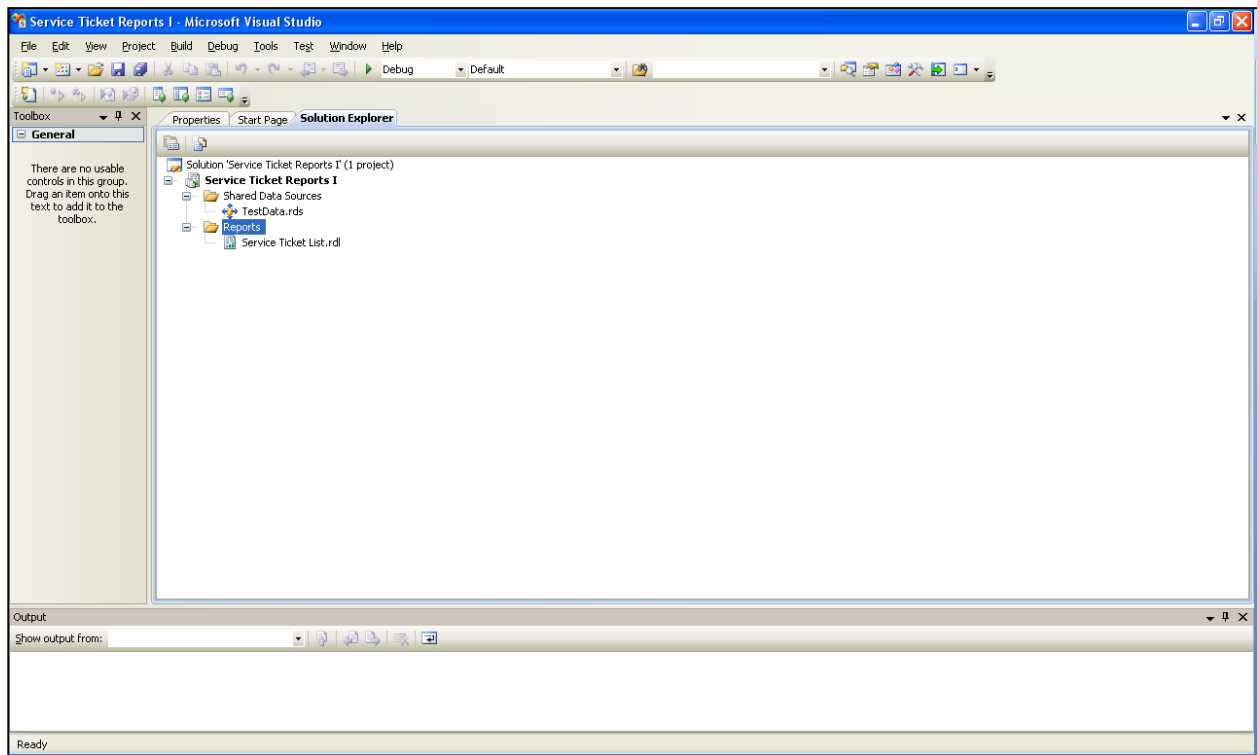
Loading Visual Studio 2008 and Creating a New Project

Start Reporting Services

- 1) Start
- 2) All Programs
- 3) Microsoft Visual Studio 2008
- 4) File
- 5) New
- 6) Project

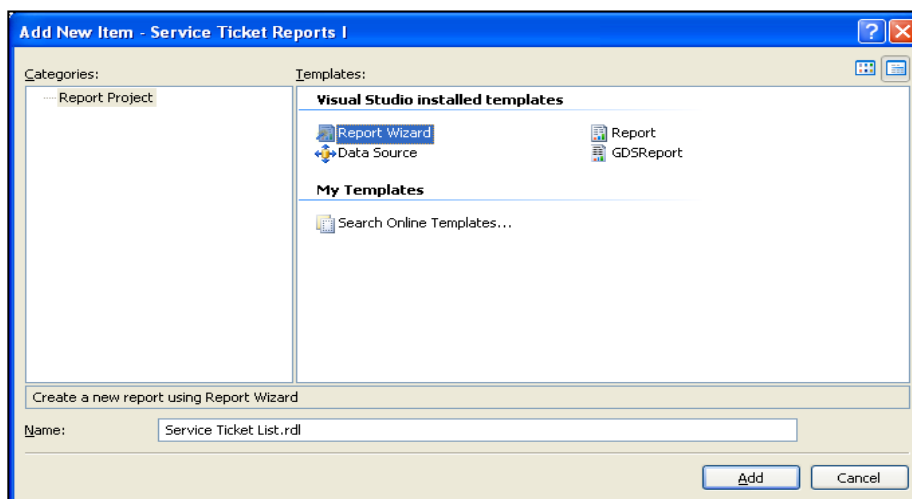


- 7) Type Name: Service Ticket Reports 1
- 8) Click OK
- 9) Click on Tab for Solution Explorer

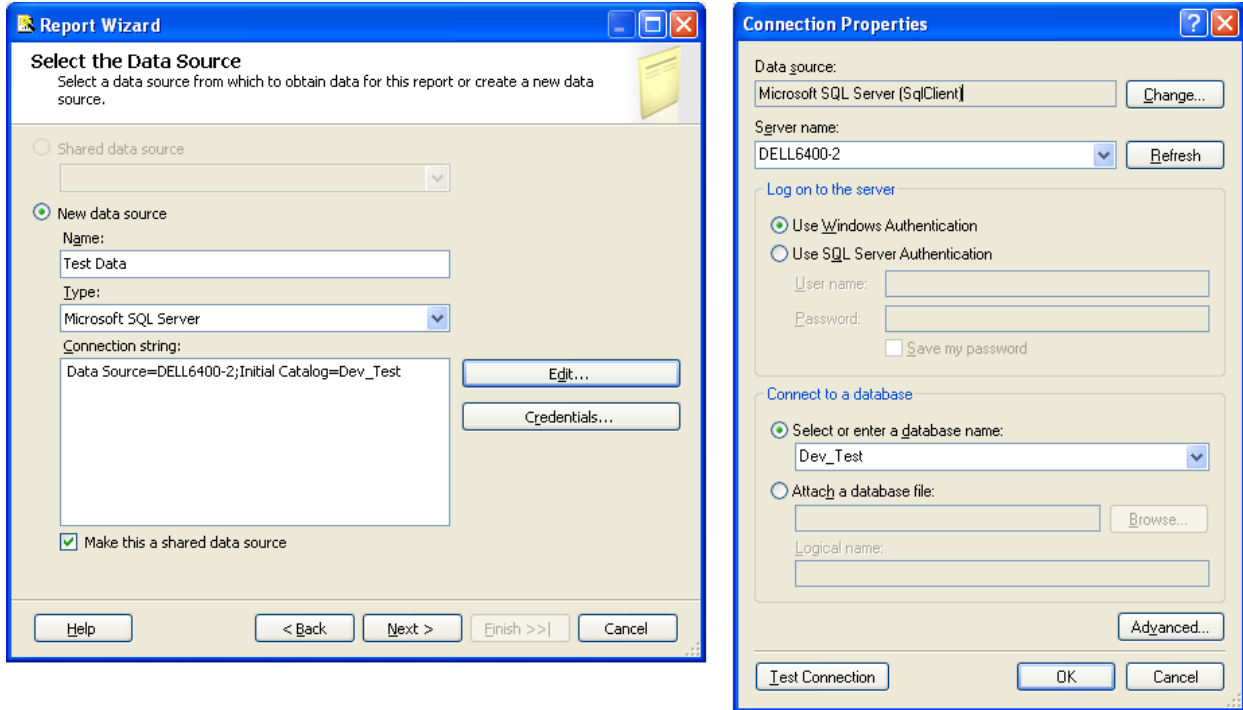


Run the Report Wizard

- 1) Right Click on "Service Ticket Reports I"
- 2) Select Add
- 3) New Item
- 4) Choose Report Wizard
- 5) At Name: type Service Ticket List



- 6) Click **ADD**
- 7) Establish the connection to the server. Note: server and database name are installation dependent, so yours will be different.

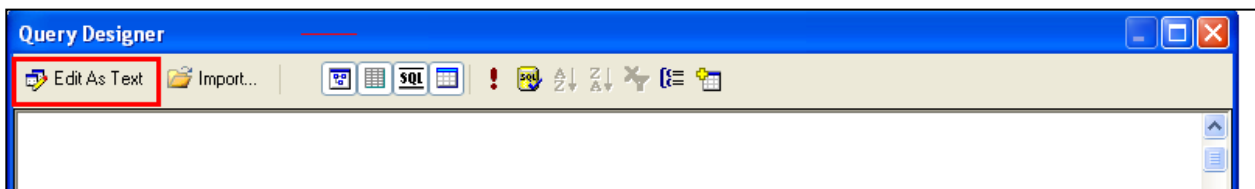


Key Points: Be sure “Make this a shared data source” is checked. You can use “Test Connection” to make sure your settings are correct.

- 8) Click **Next**
- 9) Select Query Builder

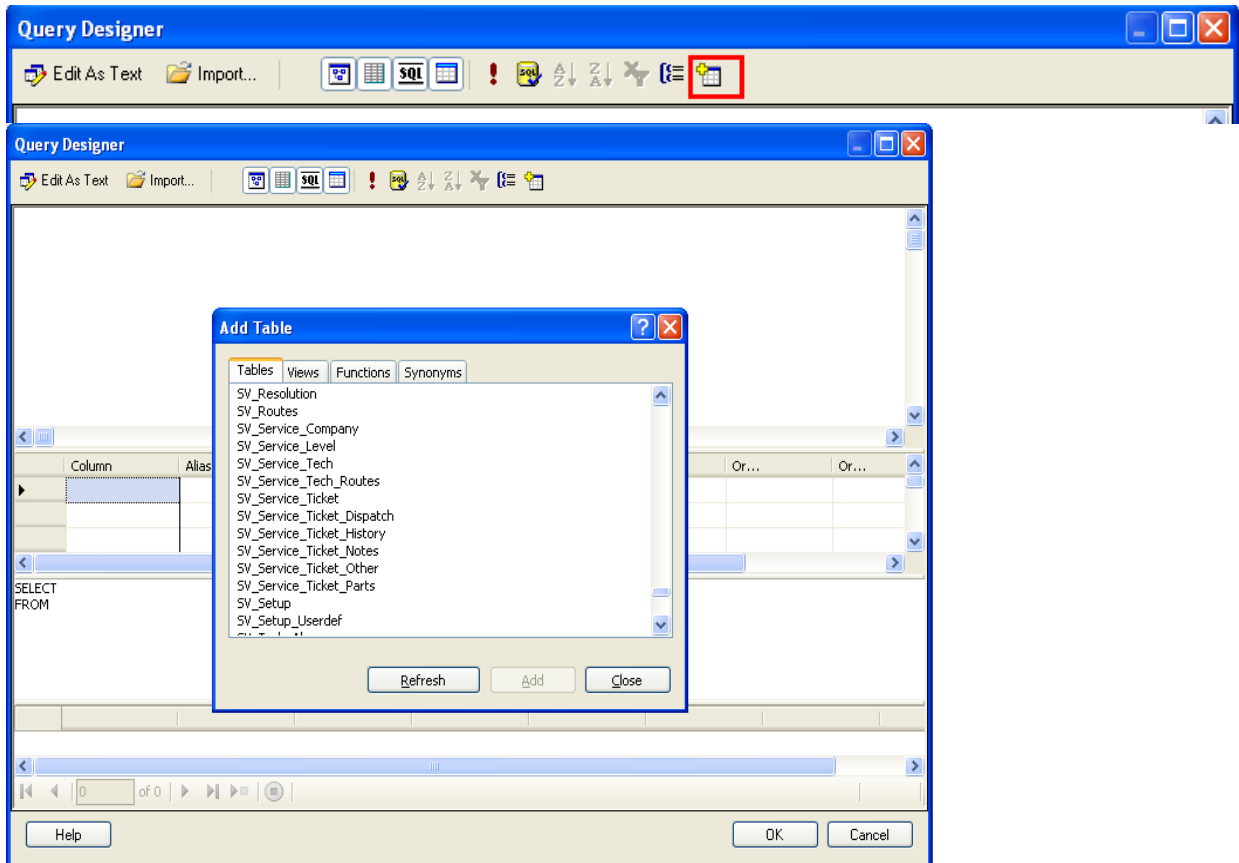
Generate the base query for the Service Ticket List

Be sure that “Edit as Text” is not highlighted.

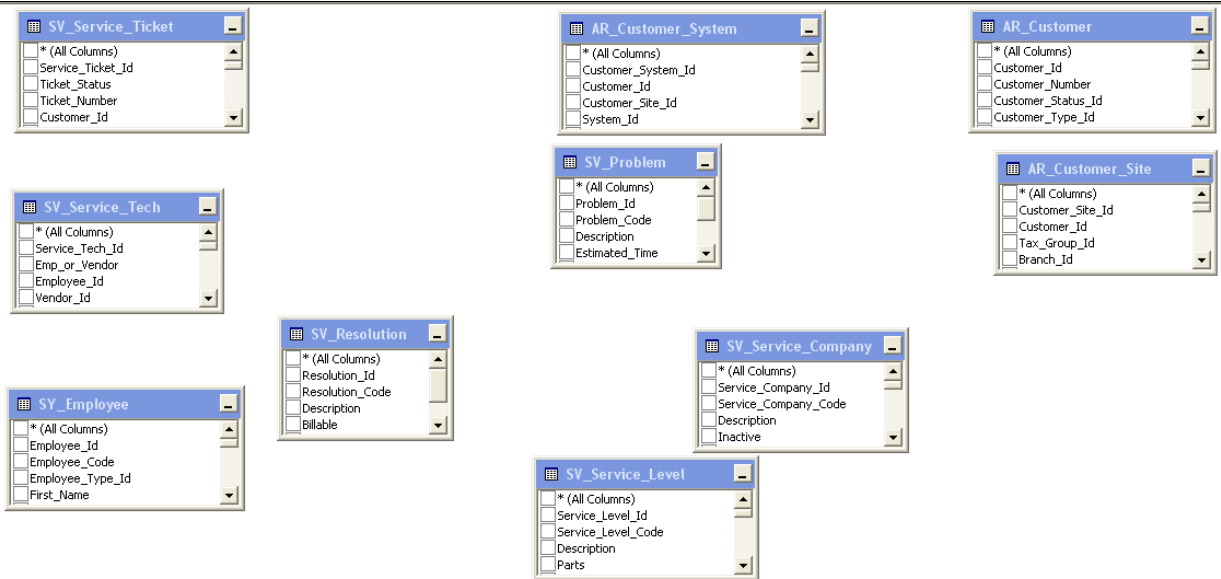


Adding Tables:

- 1) Select Add Table



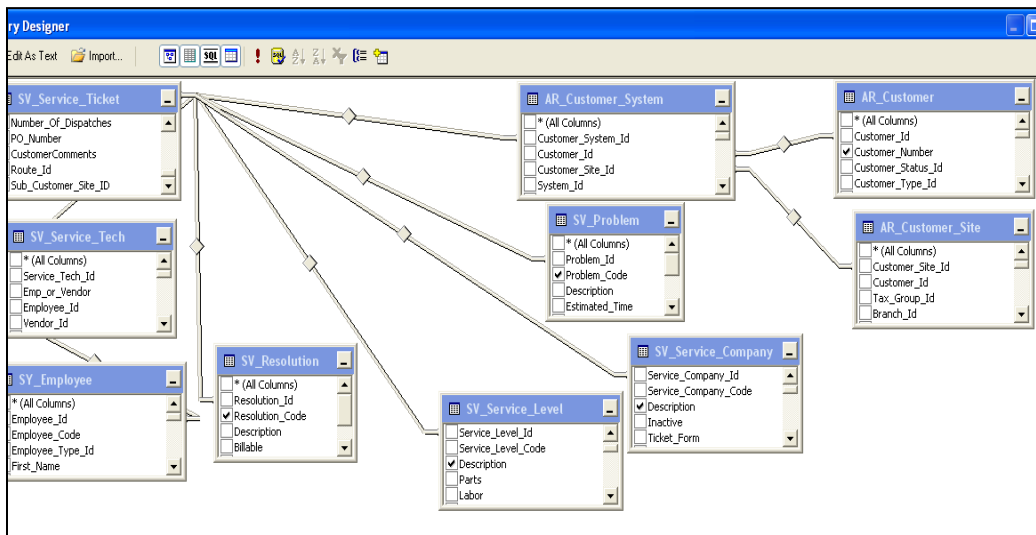
- 2) From the list that appears, select: SV_Service_Ticket, AR_Customer, AR_Customer_System, AR_Customer_Site, SV_Problem, SV_Resolution, SV_Service_Level, SV_Service_Company, SV_Service_Tech, SY_Employee
- 3) Click [Close](#)
- 4) Click and drag the tables to match the following diagram



5) Join (connect) the tables by clicking on a column in one table and dragging that column to match a column in another table as follows:

Table	Column	<i>match to</i>	Table 2	Column
SV_Service_Ticket	Customer_System_ID		AR_Customer_System	Customer_System_ID
	Problem_ID		SV_Problem	Problem_ID
	Resolution_ID		SV_Resolution	Resolution_ID
	Service_Company_ID		SV_Service_Company	Service_Company_ID
	Service_Level_ID		SV_Service_Level	Service_Level_ID
	Last_Service_Tech_ID		SV_Service_Tech	Service_Tech_ID
AR_Customer_System	Customer_ID		AR_Customer	Customer_ID
	Customer_Site_ID		AR_Customer_Site	Customer_Site_ID
SV_Service_Tech	Employee_ID		SY_Employee	Employee_ID

6) The table connections should look as follows:



7) Select columns to display by going into each table and clicking the box next to the column name as follows:

SV_Service_Ticket: Ticket_Number, Ticket_Status, Creation_Date, ClosedDate

AR_Customer: Customer_Number, Customer_Name

AR_Customer_Site: Business_Name

AR_Customer_System: Alarm_Account

SV_Resolution: Resolution_Code

SV_Problem: Problem_Code

SV_Service_Level: Description

SV_Service_Company: Description

SV_Employee: Employee_Code

8) In the column listing (middle section of window), fill out the Alias columns as indicated below.

9) Select sort type "Ascending" from the drop down in the column "Sort Order".

10) In the "Filter" column for Ticket_Number, type "> 1".

11) After steps 8, 9 and 10 the middle section should look as follows:

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...
Ticket_Number	[Ticket Number]	SV_Service_Ticket	<input checked="" type="checkbox"/>	Ascending	1	> 1	
Ticket_Status	Status	SV_Service_Ticket	<input checked="" type="checkbox"/>				
Customer_Nu...	[Customer Number]	AR_Customer	<input checked="" type="checkbox"/>				
Customer_Name	[Customer Name]	AR_Customer	<input checked="" type="checkbox"/>				
Employee_Code	Employee	SV_Employee	<input checked="" type="checkbox"/>				
Description	[Service Company]	SV_Service_Company	<input checked="" type="checkbox"/>				
Description	[Service Level]	SV_Service_Level	<input checked="" type="checkbox"/>				
Alarm_Account	System	AR_Customer_System	<input checked="" type="checkbox"/>				
Business_Name	[Site Name]	AR_Customer_Site	<input checked="" type="checkbox"/>				
Creation_Date	Opened	SV_Service_Ticket	<input checked="" type="checkbox"/>				

12) The query in the bottom section should look like this:

```

SELECT  SV_Service_Ticket.Ticket_Number AS [Ticket Number], SV_Service_Ticket.Ticket_Status AS Status,
        AR_Customer.Customer_Number AS [Customer Number], AR_Customer.Customer_Name AS [Customer Name],
        SV_Employee.Employee_Code AS Employee, SV_Service_Company.Description AS [Service Company],
        SV_Service_Level.Description AS [Service Level], AR_Customer_System.Alarm_Account AS System,
        AR_Customer_Site.Business_Name AS [Site Name], SV_Service_Ticket.Creation_Date AS Opened, SV_Resolution.Resolution_Code AS Resolution,
        SV_Problem.Problem_Code AS Problem, SV_Service_Ticket.ClosedDate AS Closed
FROM    SV_Service_Level INNER JOIN
        SV_Service_Ticket INNER JOIN
        AR_Customer_System ON SV_Service_Ticket.Customer_System_Id = AR_Customer_System.Customer_System_Id INNER JOIN
        AR_Customer ON AR_Customer_System.Customer_Id = AR_Customer.Customer_Id INNER JOIN
        AR_Customer_Site ON AR_Customer_System.Customer_Site_Id = AR_Customer_Site.Customer_Site_Id INNER JOIN
        SV_Problem ON SV_Service_Ticket.Problem_Id = SV_Problem.Problem_Id INNER JOIN
        SV_Service_Tech ON SV_Service_Ticket.Last_Service_Tech_Id = SV_Service_Tech.Service_Tech_Id INNER JOIN
        SV_Resolution ON SV_Service_Ticket.Resolution_Id = SV_Resolution.Resolution_Id INNER JOIN
        SV_Service_Company ON SV_Service_Ticket.Service_Company_Id = SV_Service_Company.Service_Company_Id ON
        SV_Service_Level.Service_Level_Id = SV_Service_Ticket.Service_Level_Id INNER JOIN
        SV_Employee ON SV_Service_Ticket.Employee_Id = SV_Employee.Employee_Id
WHERE   (SV_Service_Ticket.Ticket_Number > 1)
ORDER BY [Ticket Number]
    
```

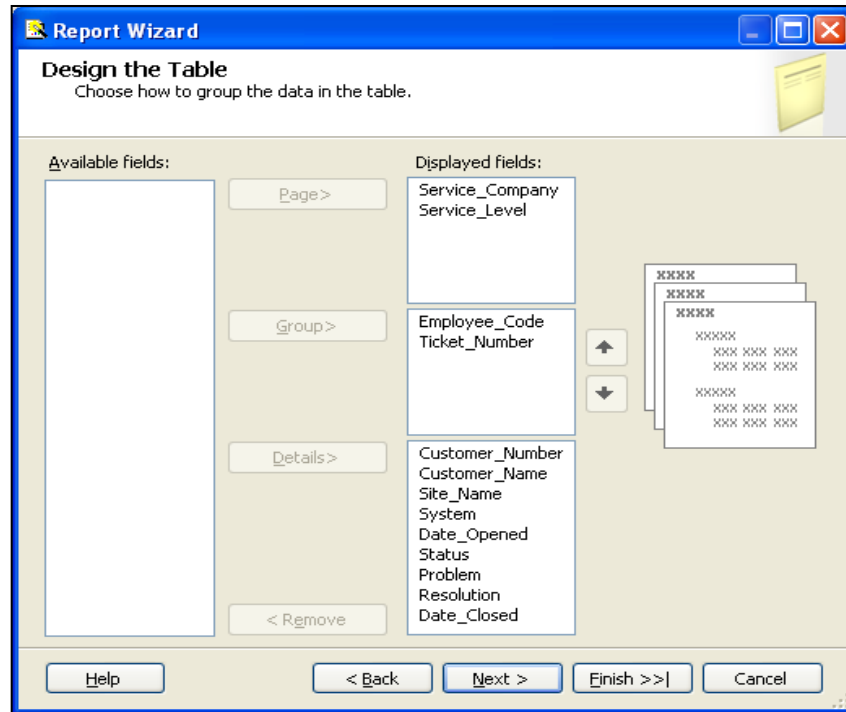
13) You can test the results of your query by clicking the **!** at the top of the screen.

14) Click [OK](#)

15) Click [Next](#)

Format the Report

- 1) Select a “Tabular Report”
- 2) Click [Next](#)
- 3) Move Service Company and Service Level to the Page Category
- 4) Move Employee and Ticket Number to the group category
- 5) Move the remaining fields to the Details Category in order as indicated below.



Note: the order of the fields in this step will determine the initial order for the report, but you can easily change that order later if needed!

- 6) Click [Next](#)
- 7) At Table Layout, choose stepped
- 8) Click [Next](#)
- 9) Choose Corporate for Style
- 10) Click [Next](#)
- 11) Name the report “Service Ticket List”
- 12) Click [Finish](#)

Review and Adjust the Report:

Service Ticket List										
[Service Company]										
[Service Level]										
Employee C	Ticket Num	Customer N	Customer N	Site Name	Svstem	Date Opene	Status	Problem	Resolution	Date Closed
[Employee_Cod]	[Ticket_Number]									
		[Customer_Nurr]	[Customer_Narr]	[Site_Name]	[System]	[Date_Opened]	[Status]	[Problem]	[Resolution]	[Date_Closed]

- 1) Preview the report by clicking on the Preview Tab
- 2) Return to the design page
- 3) Repeat the following until you are satisfied
 - a. Adjust date format
 - b. Adjust column size and headers
 - c. Adjust text color, font, weight and alignment
- 4) View the finished Report

Service Ticket List										
Alarm Co - Detroit										
Monroe Com Non Contract										
Emp	Tkt #	Cust #	Customer Name	Site Name	System	Date Opened	O/C	Problem	Resolution	Date Closed
Alex	2290									
		1001687	BLAZE SCHOOLS	BLAZE SCHOOLS Site	LS200379	6/3/2009	CL	A ADJUSTPRGRM	Re Programmed System	6/12/2009 11:47 AM
	3146									
		4013340	PORT YACHT CLUB	PORT YACHT CLUB Site	LS200294	6/24/2009	CL	TV CAMERA	Needs Additional Service	7/6/2009 4:51 PM
	3174									
		8000624	HUDSON, TOM	HUDSON, TOM Site	LS200388	6/24/2009	CL	A ARM	Fixed Trouble	7/1/2009 12:25 PM
	3960									
		4002606	PDQ CORPORATION	PDQ CORPORATION Site	720280	7/13/2009	CL	SH MISC	Needs Additional Service	8/3/2009 9:33 AM
	4716									
		8000780	ATLANTIC PLAZA	ATLANTIC PLAZA Site	P24409	7/28/2009	CL	SH MISC	Fixed Trouble	8/2/2009 3:23 PM

- 5) Save

Service Ticket Costing Report

Tables Used

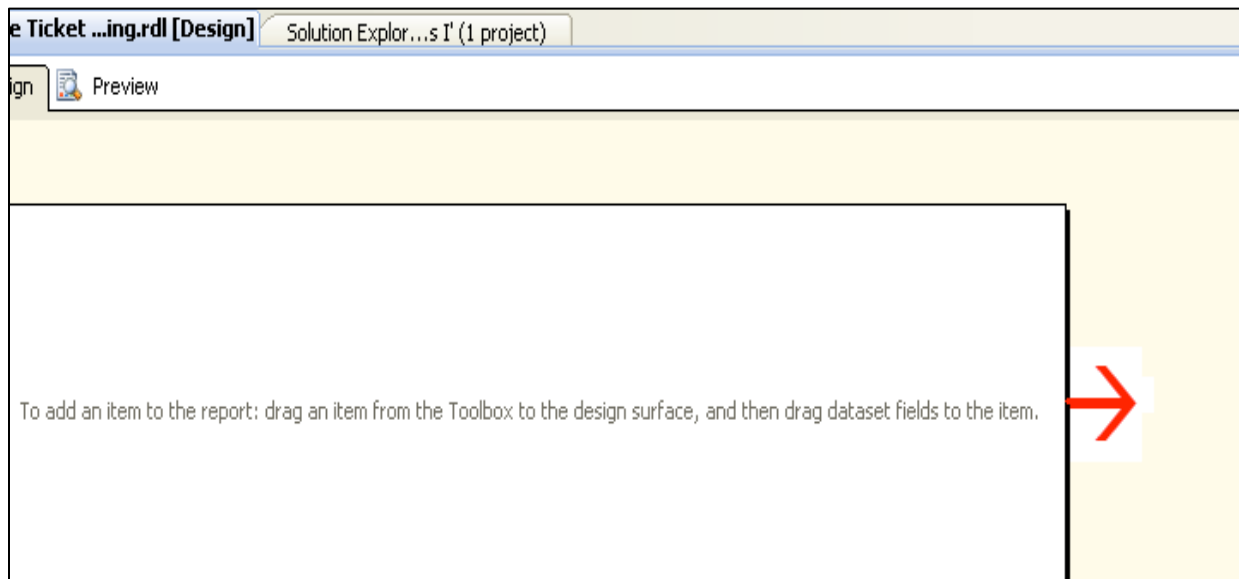
<i>SV_Service_Ticket</i>	<i>SV_Service_Tech</i>	<i>SY_Employee</i>	<i>SV_Service_Company</i>
AR_Customer_System	SV_Resolution	SV_Service_Level	
SV_Service_Ticket_Dispatch	IN_Journal	SV_Service_Ticket_Parts	IN_Receipt
AP_Invoice	AP_Credit	GL_Register	GL_Account
		AR_Invoice	AR_Credit

Create the Costing Report

- 1) Right Click "Reports"
- 2) Select [Add](#) → [New Item](#)
- 3) Change the name to "Service Ticket Costing"
- 4) Select "Report"

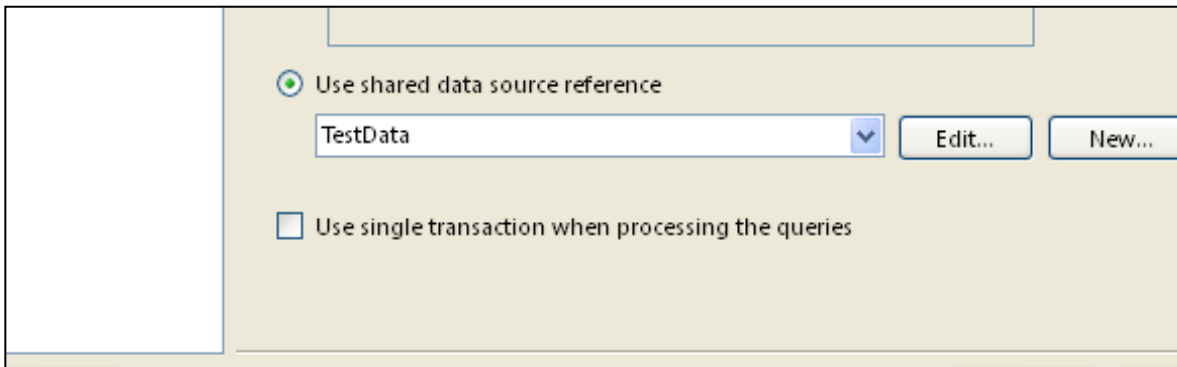
Note: Do not select "Report Wizard"!

- 5) Click [Add](#)
- 6) Expand the work area by dragging out the boundaries



- 7) Click once on "Matrix" to highlight
- 8) Start in the upper left hand corner of the work area and drag the cursor to the lower right and release to create a "Matrix Shell"

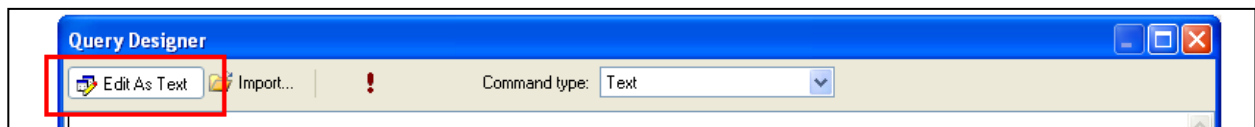
- 9) Select "Use Shared Data Source" from the window that appears



- 10) Click [Next](#)

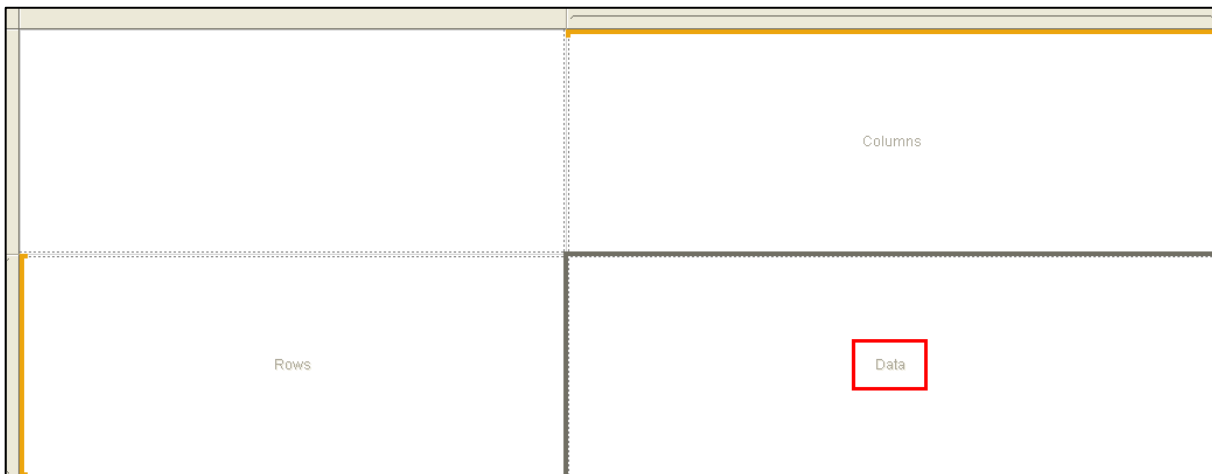
Create the Costing Query


- 1) Be sure that "Edit as Text" is highlighted



- 2) Type (or copy and paste) the query titled "Service Ticket Costing Query" into the text window. You will find it in the appendix.
- 3) Click the **!** to confirm that the query was typed in correctly. The query should run without error.
- 4) Click [Finish](#)

Creating the Report



- 1) Insert six columns using:
 - a. Right Click Data Column
 - b. Insert Column
 - c. Inside Group Right
- 2) Resize the original data column to match the new columns
- 3) Starting from the left most column:
 - a. Left click the column
 - b. Click on the small drop down box in the corner 
 - c. Fill the eight columns in this order:

Ticket Number, *Leave Blank*, Direct Revenue, *Leave Blank*, Labor Cost, *Leave Blank*, *Leave Blank*, Journal Entries

Note: A number of Columns will add "sum" to the entry. Ignore that while selecting.

- 4) Edit the entries with *sum*. Remove "Sum(" and ")". For example:
"[Sum(Direct_Revenue)]" → "[Direct_Revenue]"
- 5) Right click on the blank column to the right of "Labor_Cost"
- 6) Select "Expression"
- 7) Left click on "Fields"
- 8) Select "Parts_Cost_Issued" add a "+" and then select "Parts_Cost_DE"
- 9) Click **OK**
- 10) Right click on the column next to the previous and repeat steps 6..9, substituting:
"Expense" - "Credits_to_Expense"

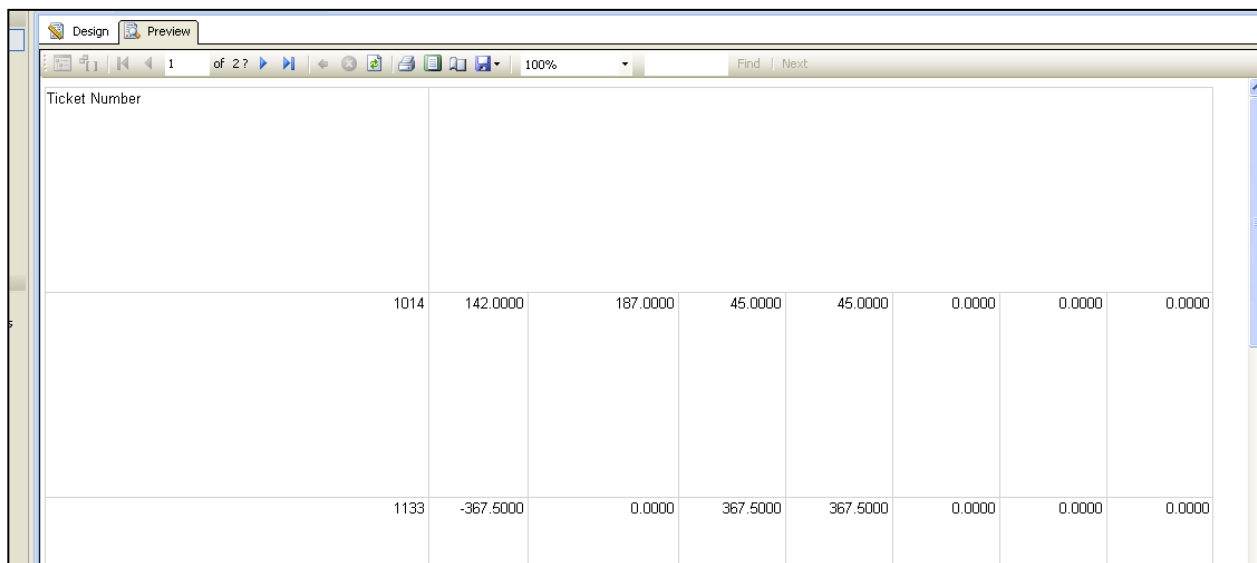
- 11) Repeat this same procedure on the column to the right of “Direct_Revenue” using:
“Labor_Cost” + “Parts_Cost_Issued” + “Parts_Cost_DE” + “Expense” –
“Credits_to_Expense” + “Journal Entries”

Note: This is the total cost for the service ticket.

- 12) Before exiting the window, select and copy the entry you just made
13) Repeat the procedure again for the remaining blank column except
a. Select “Direct_Revenue”
b. Type “- ()”
c. Position the cursor between the parenthesis and paste

Note: This is the Net profit of the service ticket!

- 14) Click “Preview”. Notice that only one column has a header and the format and spacing are large and unimpressive



The screenshot shows a software window with a 'Design' and 'Preview' tab. The 'Preview' tab is active, displaying a table with the following data:

Ticket Number								
1014	142.0000	187.0000	45.0000	45.0000	0.0000	0.0000	0.0000	
1133	-367.5000	0.0000	367.5000	367.5000	0.0000	0.0000	0.0000	

Improving the Format

- 15) Return to the Design Section
16) Shrink the row height (click the lower left corner and drag up)
17) Double click each column and fill in the “Label” field accordingly. You can also change the alignment, font, and other characteristics of the column!
18) You can add column headers to the data by right clicking the open box and selecting [Split Cells](#)
19) You can add a header and footer to the page using the icons in the upper left hand corner
20) Preview the final version

<u>Service Ticket Costing Report</u>							
Service Ticket Number	Net Profit	Gross Profit	Total Cost	Labor	Parts	Expenses	JE
1014	\$142.00	\$187.00	\$45.00	\$45.00	\$0.00	\$0.00	\$0.00
1133	(\$367.50)	\$0.00	\$367.50	\$367.50	\$0.00	\$0.00	\$0.00
1138	(\$78.75)	\$0.00	\$78.75	\$78.75	\$0.00	\$0.00	\$0.00
1179	\$102.50	\$155.00	\$52.50	\$52.50	\$0.00	\$0.00	\$0.00
1180	(\$182.05)	\$132.95	\$315.00	\$315.00	\$0.00	\$0.00	\$0.00
1207	(\$290.50)	\$94.50	\$385.00	\$385.00	\$0.00	\$0.00	\$0.00
1225	(\$265.00)	\$120.00	\$385.00	\$385.00	\$0.00	\$0.00	\$0.00
1244	\$0.00	\$140.00	\$140.00	\$140.00	\$0.00	\$0.00	\$0.00
1258	(\$180.49)	\$144.50	\$324.99	\$315.00	\$9.99	\$0.00	\$0.00

Adding Service Company and Service Level


- 1) Return to the Design Section
- 2) Right click on the Section for "Ticket Number"
 - a. Add Group
 - b. Parent Group
 - c. At "Group By" select "Service Level" from the drop down
 - d. Check "Add Group Header"
 - e. Check "Add Group Footer"
 - f. Click [OK](#)
- 3) Right click in the new data column ("Service Level")
 - a. Add Group
 - b. Parent Group
 - c. At "Group By" select "Service Company" from the drop down
 - d. Check "Add Group Header"
 - e. Check "Add Group Footer"
 - f. Click [OK](#)
- 4) Click on the Preview tab. You now have the Service Tickets grouped by Service Company and Service Level
- 5) Return to the Design Section
- 6) Format the New Columns including the fonts, labels and headers
- 7) Right Click on "Service Company"
- 8) Select "Add Total" → "After"
- 9) Right Click on "Service Level"

10) Select "Add Total" → "After"

Note: Reporting Services inserts the "Sum" for the columns "Gross Profit", "Labor" and "JE"

11) Create the Sum expressions for "Net Profit", Total Cost, Parts and Expenses by:

a. Double click the line item

b. Next to the "Value" section Select 

c. Copy the entire expression except for the "="

d. Click Cancel

e. Right click in the cell crossing that column and the lower most "Total"

f. Select "Expression"

g. Next to the "=", type "Sum ()"

h. Position the cursor between the parenthesis and paste

i. Click OK

12) You now should have each of the lowest data columns filled with an expression or sum.

Net Profit	Gross Profit	Total Cost	Labor	Parts	Expenses	JE
[Net Profit]	[Gross Profit]	[Total Cost]	[Labor]	[Parts]	[Expense]	[JE]
	i(Direct_Reve		sum(Labor_Cost			m(Journal_Entri
«EXPR»	i(Direct_Reve	«Expr»	sum(Labor_Cost	«Expr»	«Expr»	m(Journal_Entri

13) Preview the report and make necessary adjustments to the format on the final totals

14) Copy the lowest seven cells starting at "Net Profit" moving across to "JE" (shift select)

15) Paste these values into the two rows above (yes, you will overwrite some expressions)

16) Edit and Format the total labels (under "Ticket Number", "Service Level", and "Service Company")

Service Ticket Costing Report									
Service Company	Service Level	Service Ticket Number	Net Profit	Gross Profit	Total Cost	Labor	Parts	Expenses	JE
[Service_Company]									
	[Service_Level]	[Ticket Number]	[Net Profit]	[Gross Profit]	[Total Cost]	[Labor]	[Parts]	[Expense]	[JE]
		Total Level	«EXPR»	i(Direct_Reve	«Expr»	sum(Labor_Cost	«E xpr»	«E xpr»	m(Journal_Entri
	Total Companv		«EXPR»	i(Direct_Reve	«Expr»	sum(Labor_Cost	«E xpr»	«E xpr»	m(Journal_Entri
Total			«EXPR»	i(Direct_Reve	«Expr»	sum(Labor_Cost	«E xpr»	«E xpr»	m(Journal_Entri

17) Preview the Report

Enabling Drill Down

- 1) Insert a row above and inside the "Service Level"
- 2) Right click on the blank cell under the "Total Cost" heading
- 3) Select "Text Box Properties"
- 4) Change the "Name" to "CostToggle"
- 5) Change the Value to "Show Costs"
- 6) Click [OK](#)
- 7) Right click on the "Labor" Column Header
- 8) Select "Text Box Properties"
- 9) Click Visibility
- 10) Toggle [Hide](#)
- 11) Check "Display can be toggled by this report item"
- 12) Select "CostToggle"
- 13) Click [OK](#)
- 14) Repeat steps 7 through 13 for each cell in the itemized costing section including the "Service Level" total cells, but not the "Service Company" cells
- 15) Using the same method as above, use the "Total Cost" by "Service Company" to toggle the itemized total cost cells by "Service Company"
- 16) Repeat the process in 15 at the Grand Total level
- 17) Preview the report. You can now switch the itemized costing on and off!

<u>Service Ticket Costing Report</u>					
Service Company	Service Level	Service Ticket Number	Net Profit	Gross Profit	Total Cost
Alarm Co - Detroit	Monroe Com Non Contract				<input type="checkbox"/> Shows Costs
		2290	\$107.50	\$130.00	\$22.50
		2702	(\$45.00)	\$0.00	\$45.00
		3140	\$13.68	\$20.00	\$6.32
		3146	(\$180.00)	\$0.00	\$180.00
		3174	\$182.25	\$238.50	\$56.25
		3486	(\$534.50)	\$0.00	\$534.50

Service Ticket Costing Report										
Service Company	Service Level	Service Ticket Number	Net Profit	Gross Profit	Total Cost					
Alarm Co - Detroit	Monroe Com Non Contract					≡ Shows Costs	Labor	Parts	Expenses	JE
		2290	\$107.50	\$130.00	\$22.50	\$22.50	\$0.00	\$0.00	\$0.00	
		2702	(\$45.00)	\$0.00	\$45.00	\$45.00	\$0.00	\$0.00	\$0.00	
		3140	\$13.68	\$20.00	\$6.32	\$0.00	\$6.32	\$0.00	\$0.00	
		3146	(\$180.00)	\$0.00	\$180.00	\$180.00	\$0.00	\$0.00	\$0.00	
		3174	\$182.25	\$238.50	\$56.25	\$56.25	\$0.00	\$0.00	\$0.00	
		3486	(\$534.50)	\$0.00	\$534.50	\$270.00	\$242.00	\$22.50	\$0.00	
		3960	(\$56.25)	\$0.00	\$56.25	\$56.25	\$0.00	\$0.00	\$0.00	
		4716	(\$112.50)	\$0.00	\$112.50	\$112.50	\$0.00	\$0.00	\$0.00	
		4764	(\$392.50)	\$0.00	\$392.50	\$157.50	\$235.00	\$0.00	\$0.00	
		4798	\$48.75	\$150.00	\$101.25	\$101.25	\$0.00	\$0.00	\$0.00	
		5120	\$62.50	\$130.00	\$67.50	\$67.50	\$0.00	\$0.00	\$0.00	
		5181	\$39.38	\$55.55	\$16.17	\$0.00	\$16.17	\$0.00	\$0.00	
		5211	\$7.50	\$30.00	\$22.50	\$22.50	\$0.00	\$0.00	\$0.00	

Notes

Reporting on Totals Only

- 1) Click the "Solution Explorer" Tab
- 2) Right click "Service Ticket Costing.rdl"
- 3) Select **Copy**
- 4) Right click "Service Ticket Reports I"
- 5) Select **Paste**
- 6) Change the name of "Copy of Service Ticket Costing.rdl" to "Service Ticket Costing Totals.rdl"

Note: You must include the .rdl for the report to function correctly!

- 7) Double click the new report
- 8) Delete the row for "Ticket Number", the blank rows above and below, and the column for "Ticket Number"
- 9) Verify and adjust the visibility for the itemized costing cells to point to the corresponding "Total Cost" (the expression, not the column header!)
- 10) Center the Report Header
- 11) Insert Column Headers for the toggle columns
- 12) Insert a row between companies
- 13) Adjust cell and header formatting as needed
- 14) Preview the Report

<u>Service Ticket Costing Report</u>				
Service Company	Service Level	Net Profit	Gross Profit	Total Cost
Alarm Co - Detroit	Monroe Com Non Contract	(\$1209.96)	\$1292.03	⊕ \$2501.99
	Monroe Resi Non Contract	\$2984.38	\$4716.50	⊕ \$1732.12
	Company Total	\$1774.42	\$6008.53	⊕ \$4234.11
Alarm Co - Grand Rapids	Grand Rapids Com Contract	\$248.67	\$956.00	⊕ \$707.33
	Grand Rapids Com Non Contract	\$889.89	\$10783.10	⊕ \$9893.21
	Company Total	\$1138.56	\$11739.10	⊕ \$10600.54

- 15) Save

The Final Analysis

What Have You Learned?

SQL Server Reporting Services can be used to create valuable reports from your SedonaOffice database. We have just created the framework for a number of reports on service costing that many customers have requested. You get it for free.

Where Do You Go From Here?

The reports we created today still need work to reach production grade. You still need the ability to:

- Including Service Contract Revenues
- Have versions of this report targeted to levels within your company

Select date ranges

- Select by service level and company separately and individually
- Select costing by service tech
- Test pricing variations and how they would affect profitability
- Test parts costing and how it would affect profitability
- Publish the report
- Configure for emailing and internal distribution

Extending This to Other Tables

Just as we were able to create this report from the service side, you can create reports that cover any aspect of SedonaOffice's data. Using our database guides and your own ingenuity, there isn't any report you can't create.

Books and Learning Materials

There are numerous books available on using Reporting Services. Microsoft Press has released a number of very informative and well written manuals on the topic. These guides often include CD's with sample reports and training aids. If you are going to write your own reports I advise that you invest in at least one of these.

Online Assistance

The web also has plenty of guidance for those who want to write reports. You can search on Reporting Services Help. Note that you can also get lots of help on writing SQL queries as well.

Getting Reporting Services Up and Running

If it isn't already installed and configured correctly, you will need to do that before you can begin using the product. You can learn how to do this yourself, or tap an IT professional to do the heavy lifting for you. There are three things you should remember. First, you can design and generate reports using report services as long as the application is installed. You do not need a report server until you are going to publish reports. Second, the same books and websites that guide you through using Reporting Services often have sections on how to install and configure the product. Finally, your database is not tied to your Reporting Services installation. Reporting Services can be installed and uninstalled without any affect on your precious data!

Training

If you aren't into the whole self-taught method, then you can sign up for training on Reporting Services either locally or through classes that may be offered through SedonaOffice. Whether we sponsor classes depends largely on user demand, so if you are interested let us know!

Final Word

You have attended this class and are incredibly excited about the kinds of reports you can get from Reporting Services. Some of you will return to your servers and get started right away. Those of you who are willing but lack the available time can still take advantage of these wonderful reports and all they have to offer. You can contract with 3rd parties, or you can contract with SedonaOffice. We are available to design and build reports based on your specifications! No matter what path you choose, we have demonstrated the value of this resource and you now have sufficient knowledge to decide on how to best utilize it.

Notes

Appendix

Formatting Conventions for Dates

d	08/17/2000
D	Thursday, August 17, 2000
f	Thursday, August 17, 2000 16:32
F	Thursday, August 17, 2000 16:32:32
g	08/17/2000 16:32
G	08/17/2000 16:32:32
m	August 17
r	Thu, 17 Aug 2000 23:32:32 GMT
s	2000-08-17T16:32:32
t	16:32
T	16:32:32
u	2000-08-17 23:32:32Z
U	Thursday, August 17, 2000 23:32:32
y	August, 2000
dddd, MMMM dd yyyy	Thursday, August 17 2000
ddd, MMM d ""yy	Thu, Aug 17 '00
dddd, MMMM dd	Thursday, August 17
M/yy	8/00
dd-MM-yy	17-08-00

Notes

Service Ticket Costing Query

Select

```
Ticket_Number as "Ticket Number",
Company.Description as "Service Company",
SLevel.Description as "Service Level",
Employee.Employee_Code as "Employee",
ClosedDate,
"Labor Cost" =
  ISNULL ((select Cost = Sum ((Tech.RegularPayRate *
    DATEDIFF (minute, dispatch.Dispatch_Time, Departure_Time))/ 60)
  from SV_Service_Ticket_Dispatch dispatch
  inner join SV_Service_Tech Tech on Tech.Service_Tech_Id = dispatch.Service_Tech_Id
  Where (Dispatch_Time > '1900-1-1' and Departure_Time > '1900-1-1') and
    Dispatch.Service_Ticket_Id = Ticket.Service_Ticket_Id), 0),
"Parts Cost Issued" =
  ISNULL ((select Sum (Jrn.Extended_Cost * Jrn.Multiplier * -1)
  From SV_Service_Ticket_Parts Parts
  inner join IN_Journal Jrn on Jrn.Journal_Id = Parts.Journal_Id
  Where Parts.Service_Ticket_Id = Ticket.Service_Ticket_Id), 0),
"Parts Cost DE" =
  ISNULL ((SELECT Sum(rp.extended_cost * case r.ls_return when 'N' then 1 else -1 end)
  From IN_Receipt r
  Inner Join IN_Receipt_Parts rp ON rp.Receipt_Id = r.Receipt_Id
  Where r.Service_Id = Ticket.Service_Ticket_Id and r.Direct_Expense = 'Y'),0),
"Expense" =
  ISNULL ((select Sum (Expense_Total)
  From AP_Invoice
  Where Service_Ticket_Id = Ticket.Service_Ticket_Id), 0) +
  ISNULL ((Select Sum (Expense_Cost)
  From IN_Receipt
  Where Service_Id = Ticket.Service_Ticket_Id and Invoice_Id < 2), 0),
"Credits to Expense" =
  ISNULL ((Select SUM (Expense_Total * -1)
  From AP_Credit
  Where Service_Ticket_Id = Ticket.Service_Ticket_ID), 0),
"Journal Entries" =
  ISNULL ((Select SUM (r.Amount * CASE WHEN Credit_Or_Debit = 'C' THEN -1
    WHEN Credit_Or_Debit = 'D' THEN 1 END)
  From GL_Register r
  INNER JOIN GL_Account gl ON gl.Account_id = r.Account_id
  Where r.Service_Ticket_Id = Ticket.Service_Ticket_Id
  AND r.Register_Type_Id = 4
```

```
AND gl.Account_Type_Id IN (3,13,14,15,16)), 0) ,
"Direct Revenue" =
ISNULL ((Select SUM (Inv.Amount - Inv.Tax_Amount - IsNull (Credit.Amount,0)
+ IsNull (Credit.Tax_Amount, 0))
From AR_Invoice Inv
Left Outer join AR_Credit Credit On Credit.Service_Ticket_Id =
Ticket.Service_Ticket_Id
Where Inv.Service_Ticket_Id = Ticket.Service_Ticket_Id), 0)
From SV_Service_Ticket Ticket
Inner Join SV_Service_Company Company on Company.Service_Company_Id =
Ticket.Service_Company_Id
Inner Join SV_Service_Level SLevel on SLevel.Service_Level_Id =
Ticket.Service_Level_Id
Inner Join SV_Service_Tech Tech on Tech.Service_Tech_Id =
Ticket.Last_Service_Tech_Id
Inner Join SY_Employee Employee on Employee.Employee_Id = Tech.Employee_Id
Where Ticket.Service_Ticket_Id > 1 and Resolution_Id <> 12 and Ticket_Status = 'CL'
Order by Ticket_Number
```

Notes