

## See More Intelligence instantly with SeeMoreData

*By: Ferenc Mantfeld, Chief Technologist, ROI Data.*



There are numerous reporting facilities in the market today, ranging from the bare essential character-based Sql\*Plus / Microsoft Query Analyzer reporting type tools, to the very complex Business Intelligence and enterprise reporting suites. Many ERP, CRM and other business processing information systems do a great job capturing all sorts of information, but have difficulty extracting the information to mean something specific to a particular job or role within a corporation.

In many cases, installing a complex BI suite to create simple reports and to provide elementary-to-intermediate analysis and data mining capabilities is like using a Formula 1 racing car to do the grocery shopping. As a result the implementation:

- is badly planned
- is under-budgeted
- is difficult to set up
- is difficult to maintain
- needs specialist skills to drive the effort.

Ask 5 people what Business Intelligence means to them, and you will probably receive 10 different answers. For the most part, Business Intelligence solutions all have a fundamental principle and that is that the information presented in whatever form, is based on data.

For this presentation, we will assume a query of the sales data in the Microsoft Sql Server Northwind demo database that contains Orders, Order Details, Customers, Employees, Suppliers, Products tables.

Every time a sale is made, there are certain attributes to the sale, and some of those attributes are dimensions, others are measures. Let us see what is meant by this.

When a sale is made, the attributes of the sale for our example will be:

- The country it was sold in
- The city it was sold in
- The customer it was sold to
- The agent who made the sale
- The date on which the sale was made
- The amount of the sale
- The shipping / freight amount if applicable
- The Order ID
- The shipping date

This information can be relatively easily obtained from Sql Server with a normal query. The example of this query also calculates some derived information, like the year and month of the sale and the combination of these.

A developer with a fair knowledge of SQL and Sql Server specific SQL should be able to write this query in about 10 – 15 minutes using Query Analyzer.

### ***Initial Report Setup***

The query in Sql Server Northwind Database is as follows:

```
select Country, City, CompanyName, OrderID, OrderTotal, OrderDate, Freight,  
ShippedDate, ORDYY, ORDMM,  
convert(varchar,ORDYY) + ' ' + convert(varchar,ORDMM) as ORDYYMM, ORDMON, Soldby as Agent from
```



**See More Intelligence Instantly with SeeMoreData**

(select C.Country, C.City, C.CompanyName, O.OrderID, OD.OrderTotal, O.OrderDate, O.Freight, O.ShippedDate, DATEPART( year , OrderDate) as OrdYY, replicate('0',2 – len (DATEPART( month ,OrderDate ))) + convert( varchar , DATEPART( month , OrderDate)) as OrdMM /\* Sql Server equivalent of LPAD \*/ , datename (month ,OrderDate ) as ORDMON, E.FirstName + ' ' + E.LastName as SoldBy from Orders O, Customers C, Employees E, (select OrderID, sum ( UnitPrice \* Quantity \* (1 - Discount)) as OrderTotal from "Order Details" group by OrderID ) OD where O.CustomerID = C.CustomerID and O.OrderID = OD.OrderID and O.EmployeeID = E.EmployeeID) INQ

Viewed inside a regular SeeMoreData report, the information retrieved looks as follows:

Country	City	Customer Name	Order ID	Order Total	OrderDate	Freight Charge	ShippedDate	ORDYY
France	Reims	Vins et alcools Chevalier	10248	\$ 440.00	04-Jul-1996	\$ 32.38	16-Jul-1996	1996 07
Germany	Münster	Toms Spezialitäten	10249	\$ 1,863.40	05-Jul-1996	\$ 11.61	10-Jul-1996	1996 07
UK	Cowes	Island Trading	10321	\$ 144.00	03-Oct-1996	\$ 3.43	11-Oct-1996	1996 10
Mexico	México D.F.	Pericles Comidas clásicas	10322	\$ 112.00	04-Oct-1996	\$ 0.40	23-Oct-1996	1996 10
Germany	Brandenburg	Königlich Essen	10323	\$ 164.40	07-Oct-1996	\$ 4.88	14-Oct-1996	1996 10
USA	Boise	Save-a-lot Markets	10324	\$ 5,275.72	08-Oct-1996	\$ 214.27	10-Oct-1996	1996 10
Germany	Brandenburg	Königlich Essen	10325	\$ 1,497.00	09-Oct-1996	\$ 64.86	14-Oct-1996	1996 10
Spain	Madrid	Bólide Comidas preparadas	10326	\$ 982.00	10-Oct-1996	\$ 77.92	14-Oct-1996	1996 10
Sweden	Bräcke	Folk och få HB	10327	\$ 1,810.00	11-Oct-1996	\$ 63.36	14-Oct-1996	1996 10
Portugal	Lisboa	Furia Bacalhau e Frutos do Mar	10328	\$ 1,168.00	14-Oct-1996	\$ 87.03	17-Oct-1996	1996 10
USA	Lander	Split Rail Beer & Ale	10329	\$ 4,578.43	15-Oct-1996	\$ 191.67	23-Oct-1996	1996 10
Venezuela	Barquisimeto	LILA-Supermercado	10330	\$ 1,649.00	16-Oct-1996	\$ 12.75	28-Oct-1996	1996 10
France	Marseille	Bon app'	10331	\$ 88.50	16-Oct-1996	\$ 10.19	21-Oct-1996	1996 10
Canada	Montréal	Mère Paillarde	10332	\$ 1,786.88	17-Oct-1996	\$ 52.84	21-Oct-1996	1996 10
Finland	Oulu	Wartian Herkku	10333	\$ 877.20	18-Oct-1996	\$ 0.59	25-Oct-1996	1996 10
France	Lyon	Victuailles en stock	10334	\$ 144.80	21-Oct-1996	\$ 8.56	28-Oct-1996	1996 10
Ireland	Cork	Hungry Owl All-Night Grocers	10335	\$ 2,036.16	22-Oct-1996	\$ 42.11	24-Oct-1996	1996 10
Portugal	Lisboa	Princesa Isabel Vinhos	10336	\$ 285.12	23-Oct-1996	\$ 15.51	25-Oct-1996	1996 10

Figure 1: Report run against Sql Server displayed within regular SeeMoreData GUI

The above graphic illustrates a regular list / matrix (spreadsheet-like) report with a few formatting options and this does not show a lot of capability in its current form above and beyond what any reporting tool should be able to do.

Country	City	Customer Name	OrderID	Order Total	OrderDate	Freight Charge	ShippedDate	ORDYY	ORDMM	ORDYYMM	Order Month	Agent
France	Reims	Vins et alcools Chevalier	10248	\$ 440.00	04-Jul-1996	\$ 32.38	16-Jul-1996	1996 07	1996 07	1996 07	July	Steven Buchanan
Germany	Münster	Toms Spezialitäten	10249	\$ 1,863.40	05-Jul-1996	\$ 11.61	10-Jul-1996	1996 07	1996 07	1996 07	July	Michael Suyama
UK	Cowes	Island Trading	10321	\$ 144.00	03-Oct-1996	\$ 3.43	11-Oct-1996	1996 10	1996 10	1996 10	October	Janet Leverling
Mexico	México D.F.	Pericles Comidas clásicas	10322	\$ 112.00	04-Oct-1996	\$ 0.40	23-Oct-1996	1996 10	1996 10	1996 10	October	Robert King
Germany	Brandenburg	Königlich Essen	10323	\$ 164.40	07-Oct-1996	\$ 4.88	14-Oct-1996	1996 10	1996 10	1996 10	October	Margaret Peacock
USA	Boise	Save-a-lot Markets	10324	\$ 5,275.72	08-Oct-1996	\$ 214.27	10-Oct-1996	1996 10	1996 10	1996 10	October	Anne Dodsworth
Germany	Brandenburg	Königlich Essen	10325	\$ 1,497.00	09-Oct-1996	\$ 64.86	14-Oct-1996	1996 10	1996 10	1996 10	October	Nancy Davolio



*Figure 2: Logical analysis of report attributes*

We can see that certain of the attributes of the data for each row are raw attributes, some are measures (attributes with a numerical value to it that can be measured) and some attributes are derived from the base attribute. The following graphic serves to illustrate this purpose.

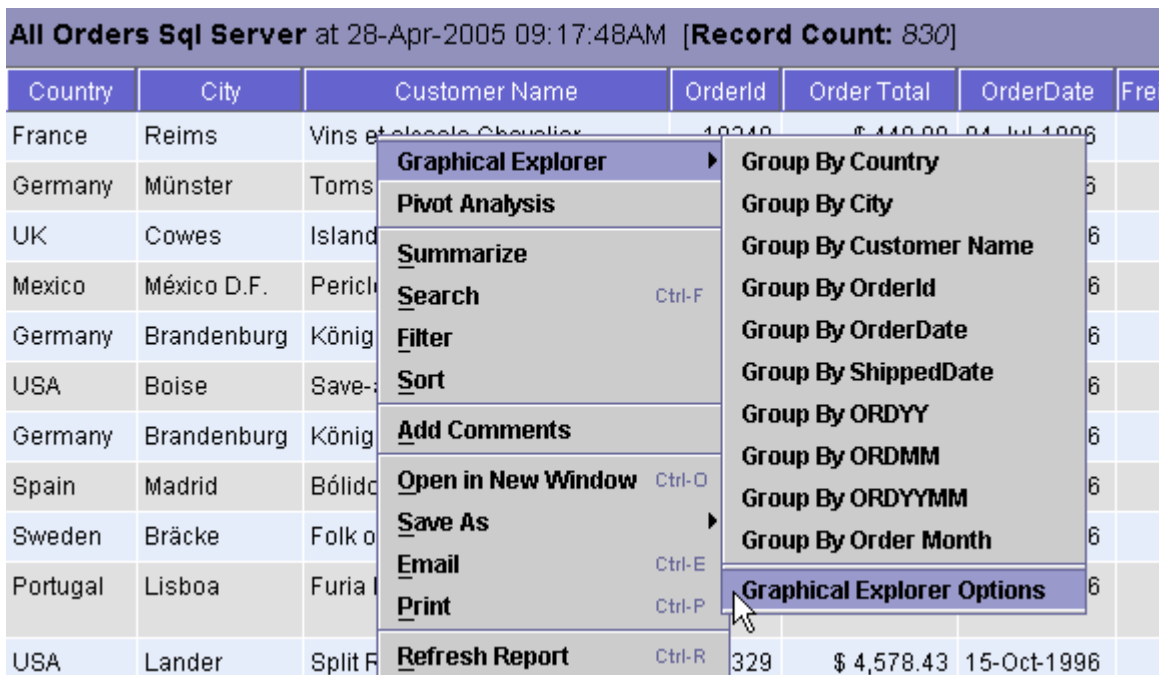
In its present form, the report itself does not give us a whole lot of meaningful analyzed data. This could just as easily be a report of the worldwide sales for the week, or if the report was parameterized, it could be the sales for given countries and cities between a certain date range, which is applied in the where clause of the query as a prepared query statement.

## **Dynamic Ad Hoc Analysis using SeeMoreData Graphical Explorer**

SeeMoreData Graphical Explorer is a tool that is built into the SeeMoreData applet that allows the end user to perform various dynamic groupings of summarized data so that exposing hidden trends and top-down analysis is easily achieved.

We will now explore the various paths of the graphical explorer and how it can provide meaningful analysis of the underlying data to the end user.

The initiation of the Graphical Explorer is done by right-clicking anywhere inside the report and choosing the 'Graphical Explorer' option of the right-click menu and then 'Graphical Explorer Options' from the sub-menu.



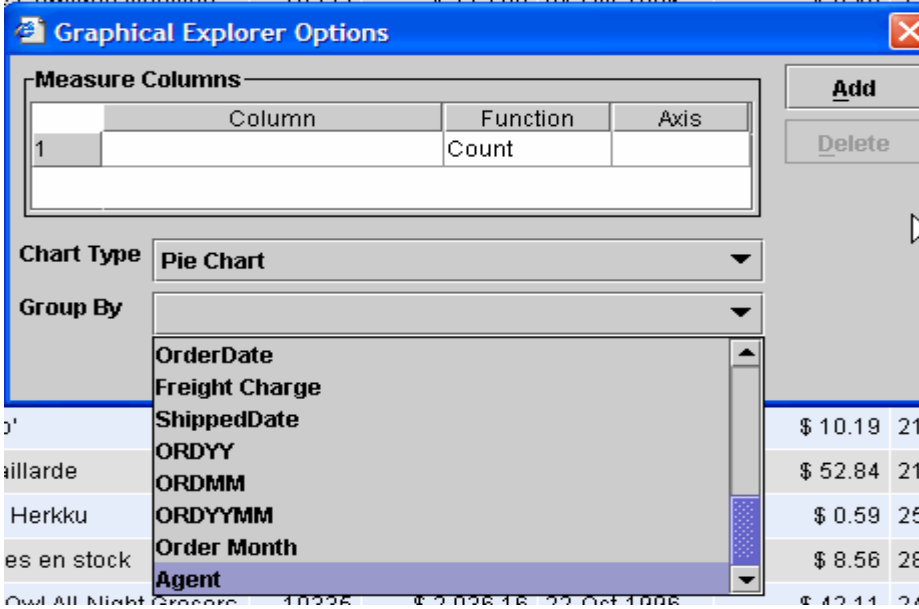
*Figure 3: Initially invoking SeeMoreData Graphical Explorer Ad Hoc analysis tool*

This invokes an end-user dialog where the user can specify which measures and attributes they wish to see grouping on.

The default presentation is a 3D pie chart.

Now we specify the columns for the grouping in the Group By picklist, which is populated from the formatted report column headings, and then we can specify the measure.

Suppose we would like to see total sales (Order Total) by Agent:



The Graphical Explorer dialog opens and the default Chart Type is a 3D Pie chart.

The list of all columns (attributes) from the report are available to choose from.

We have decided that we want to total the order amount by Agent, thus we choose Agent as the group by column.

Figure 4: Selecting the initial group by attribute

Next, we choose the measure to total and the aggregate function.

We want to add up the Order Total and group this by Agent, so pick the Order Total from the pick list of columns from the report.

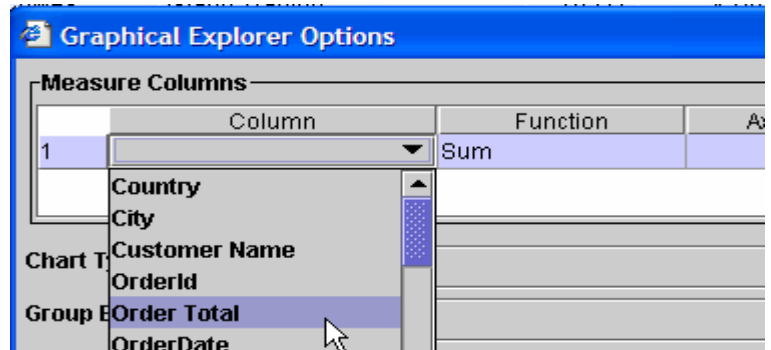


Figure 5: Selecting the initial measure from the pick list

As we click OK from the Graphical Explorer Options menu, the Ad Hoc Analysis engine is activated and analyzes all the records and groups the sum of Order Total by Agent and presents this in a pie chart output format.

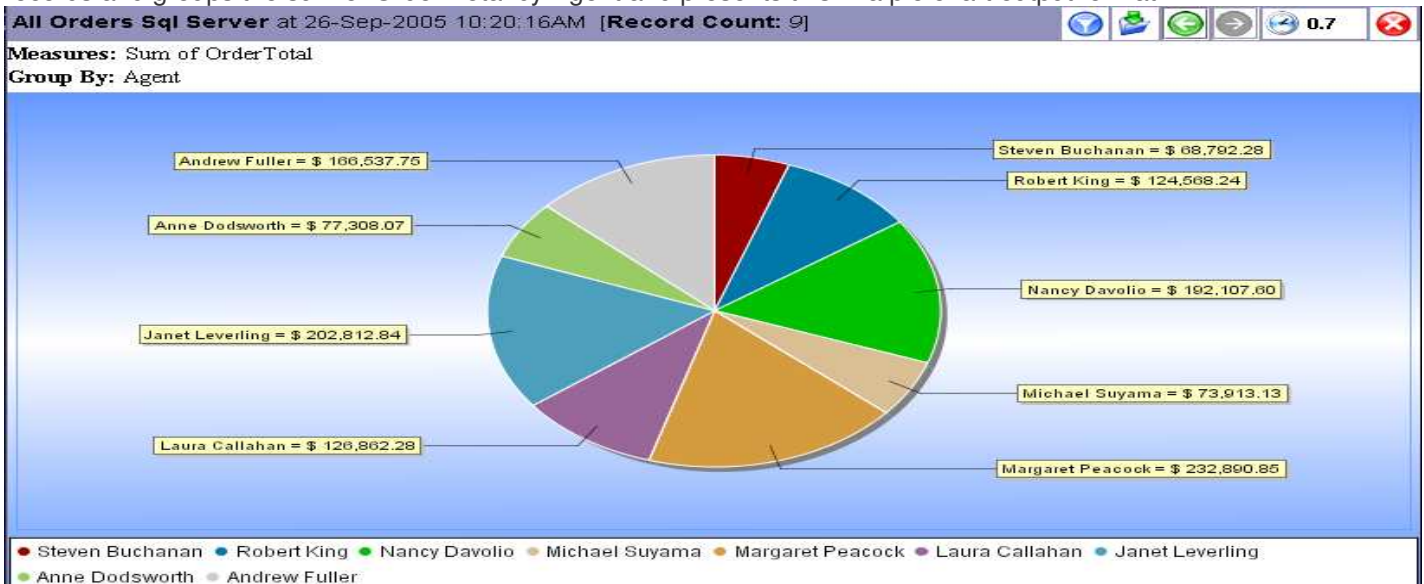


Figure 6: Initial Analysis showing grouping of sum of Order Totals by Agent.

**See More Intelligence Instantly with SeeMoreData**

Notice the Pie chart on the above graphic showing exactly this information. As you mouse over each slice, the legend and value for the slice is shown as a hover value.

If we now double-click any Agent's slice of the pie, the Graphical mini-menu will appear, and request the user to choose the next grouping level. All columns except those already being used by grouping, will be shown. For this example, we will now group the information by country. Note that this subsequent level of grouping will be a subset of the initial grouping, this for the particular agent.

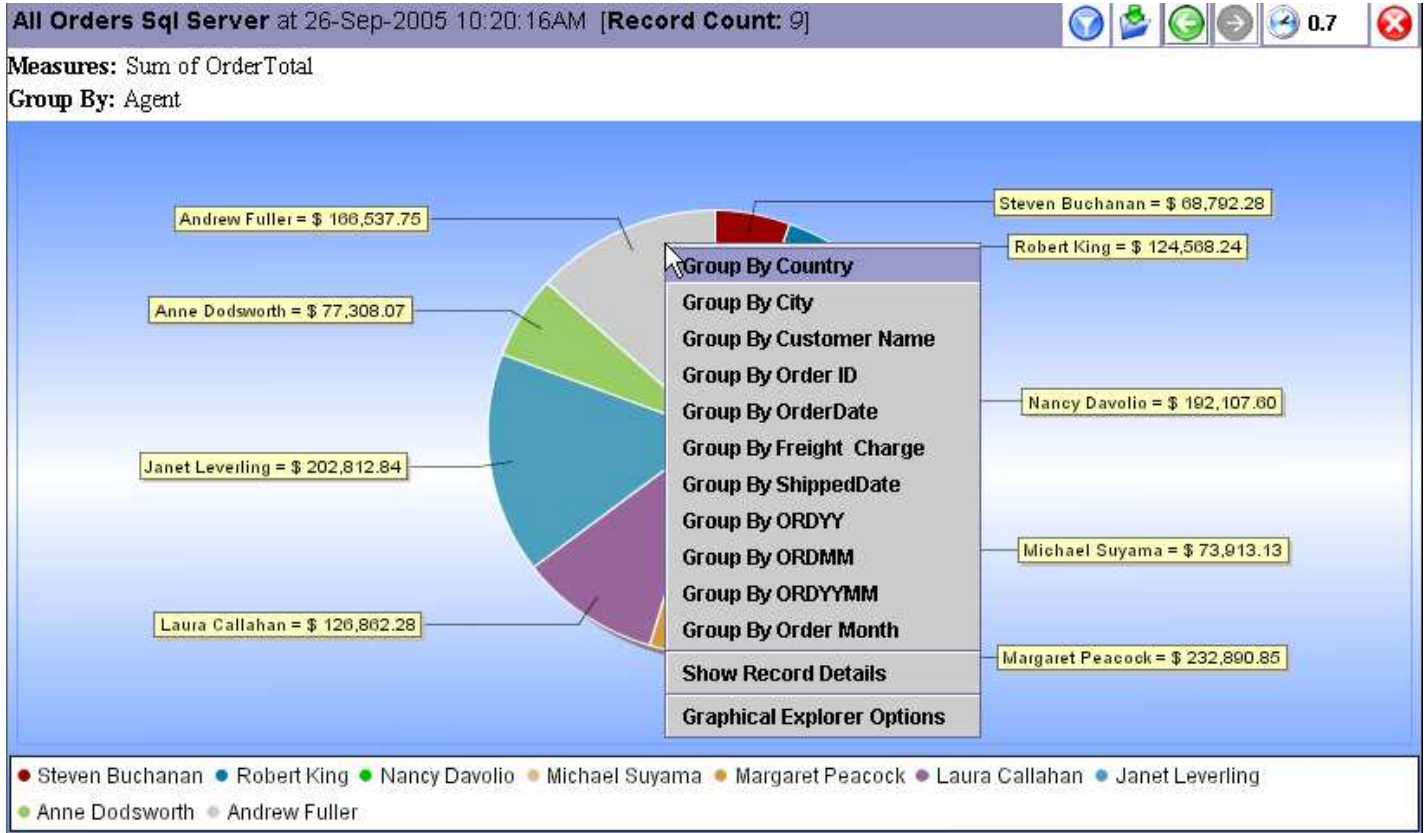


Figure 6: Initial Analysis showing grouping of sum of Order Totals by Agent.

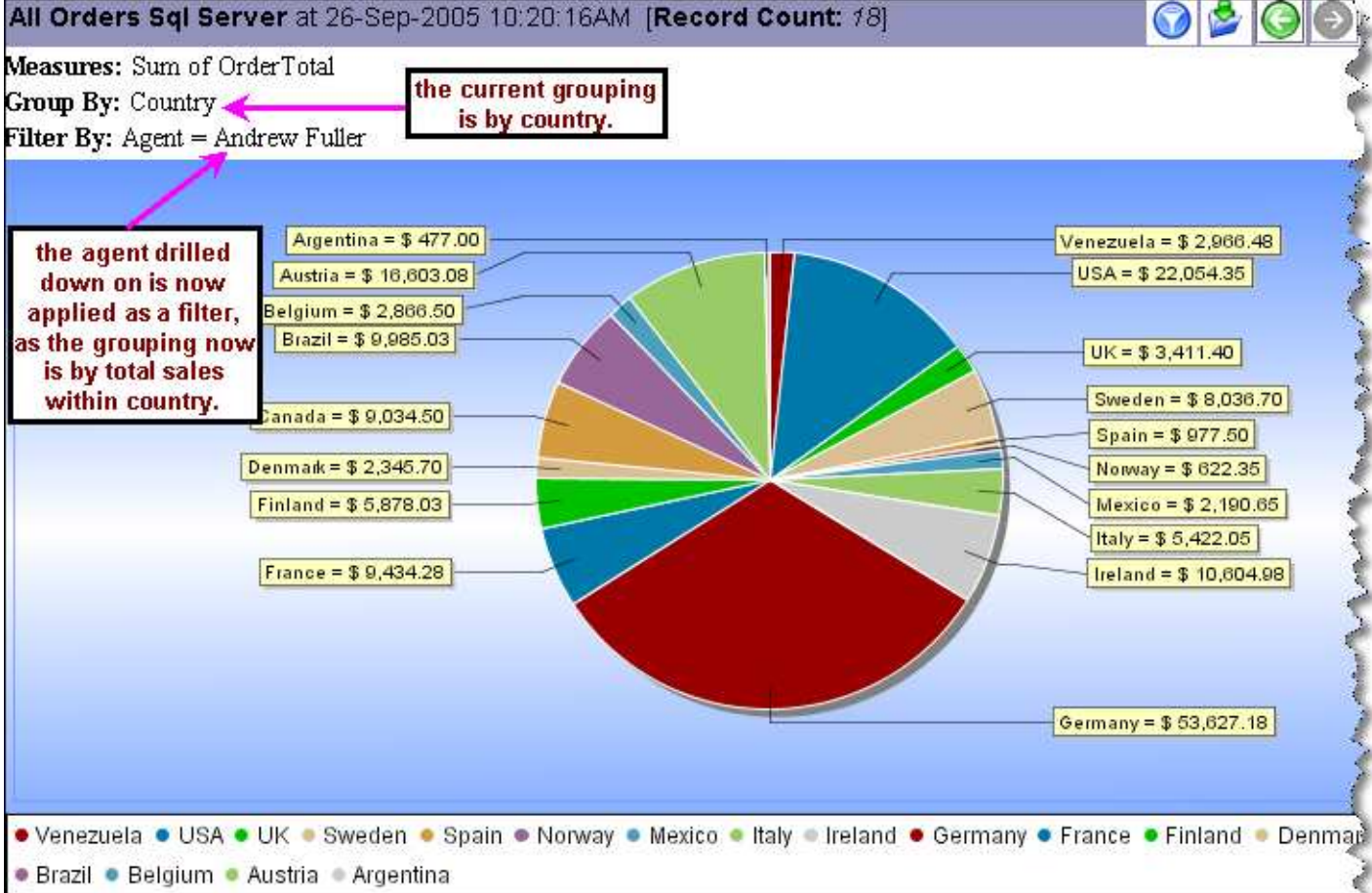


Figure 8: show the next level of grouping by attribute (country) within the first grouping attribute (Agent).

We will now drilldown for the country of Germany by double-clicking the Germany slice of the pie chart.

As we do so, note that the columns Agent and Country have been removed from the list of available grouping columns.

Now we will select the City and this will then give us another pie chart that shows us the total of sales by city within Germany for the Agent Andrew Fuller.

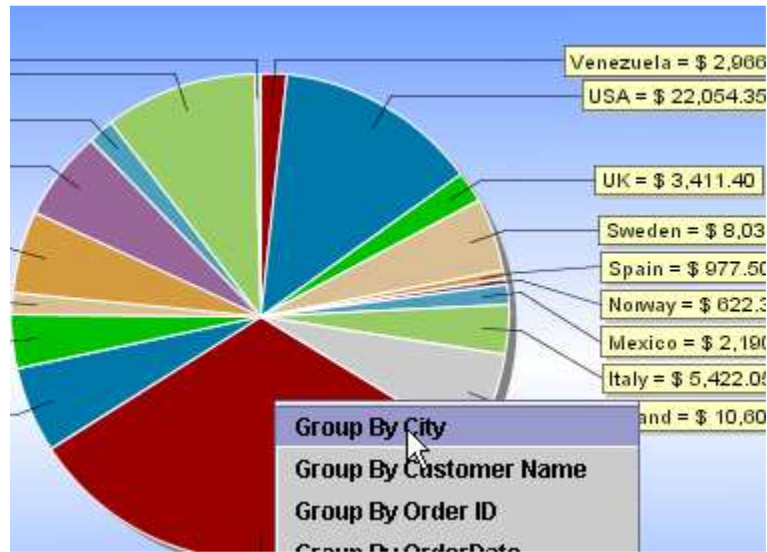


Figure 9: exposing the next level of drilldown grouping attributes for the user to choose.

We now get a pie chart which seems a little over crowded and it could be difficult to tell graphically, instantly and visually the ranking by total sales amount for each of the smaller cities.

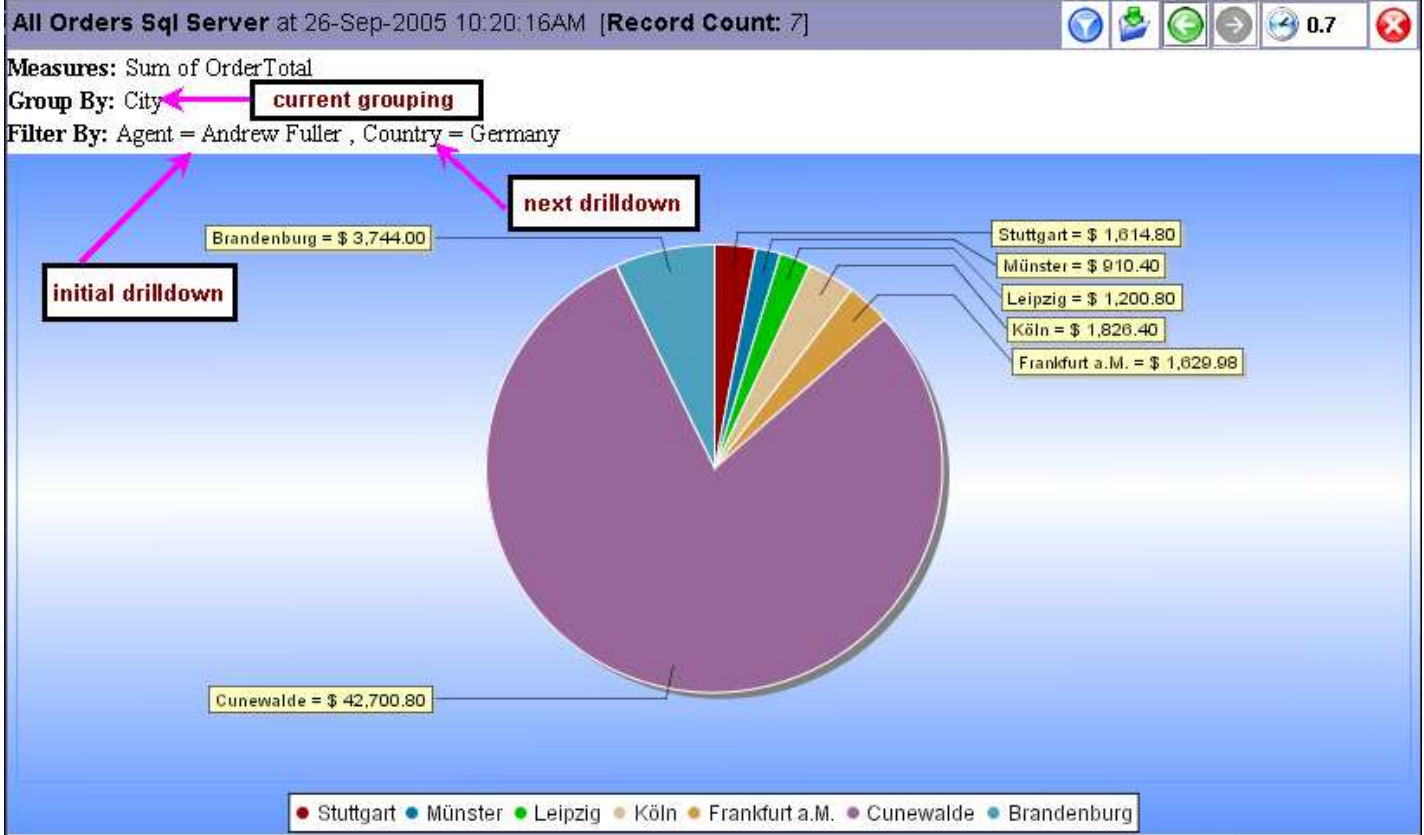


Figure 10: Total Sales by city for country=Germany and Agent='Andrew Fuller'

If the end user wanted to change the chart to a different type of chart, they can do this by invoking a right-click menu Layout option → Change display Type → New chart type.

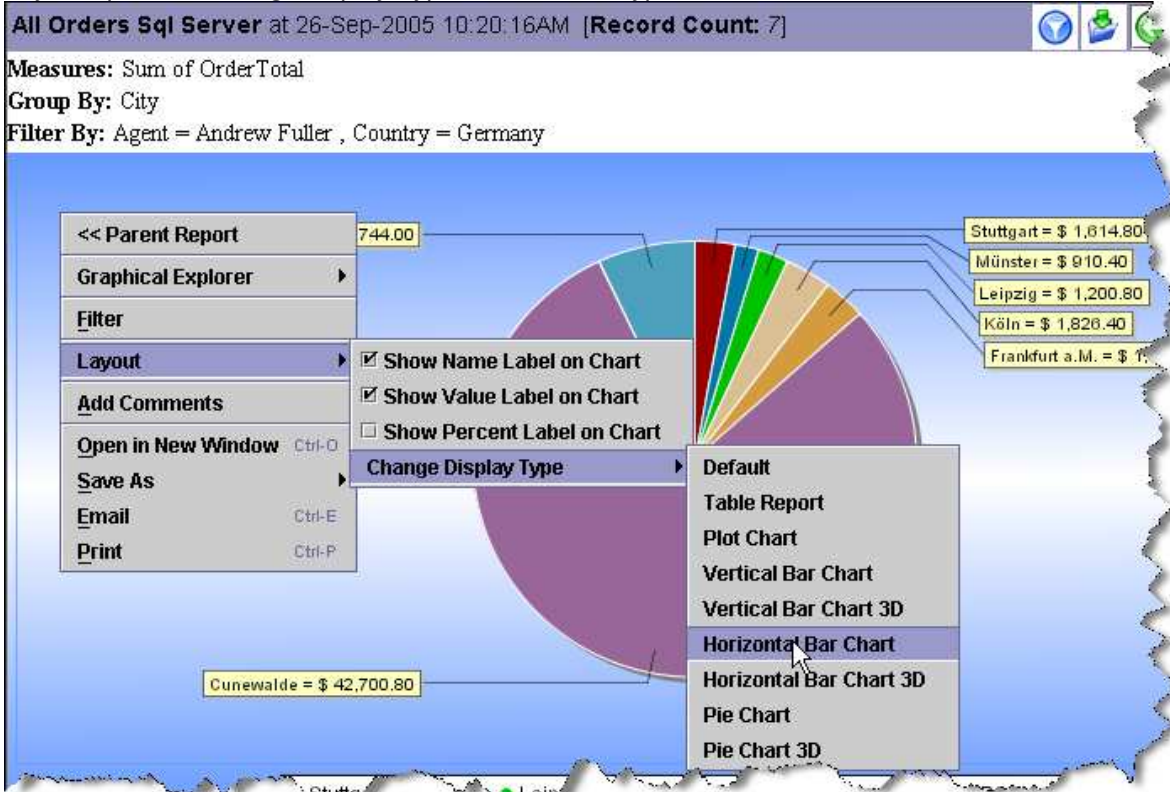


Figure 11: Changing the chart type

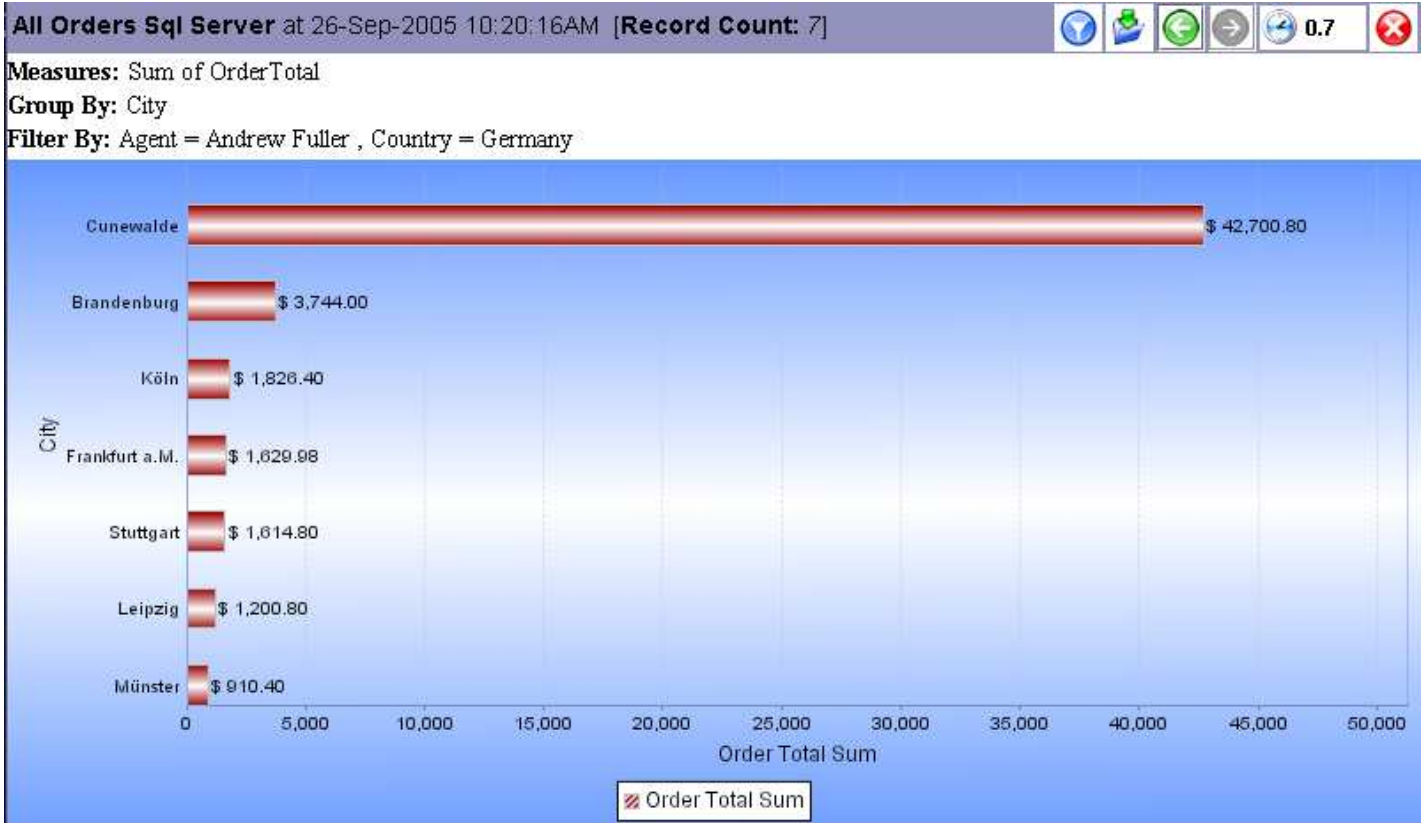


Figure 12: Chart type changed to horizontal bar chart.

This allows us now to better see the breakdown of sales by city for the chosen subgroup.

If we now wanted to further group any particular city's sales within the order year (ORDYY field), just double-click the city to drill into and choose ORDYY from the pick list of columns.

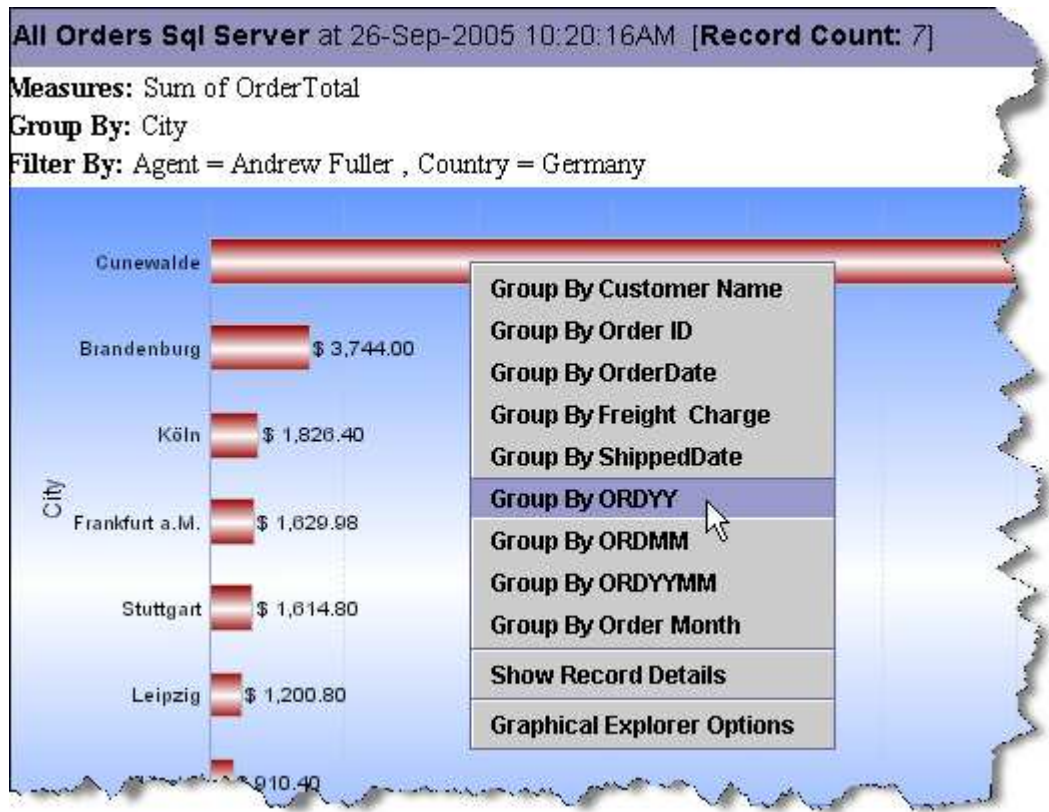


Figure 13: Group the City results by Year

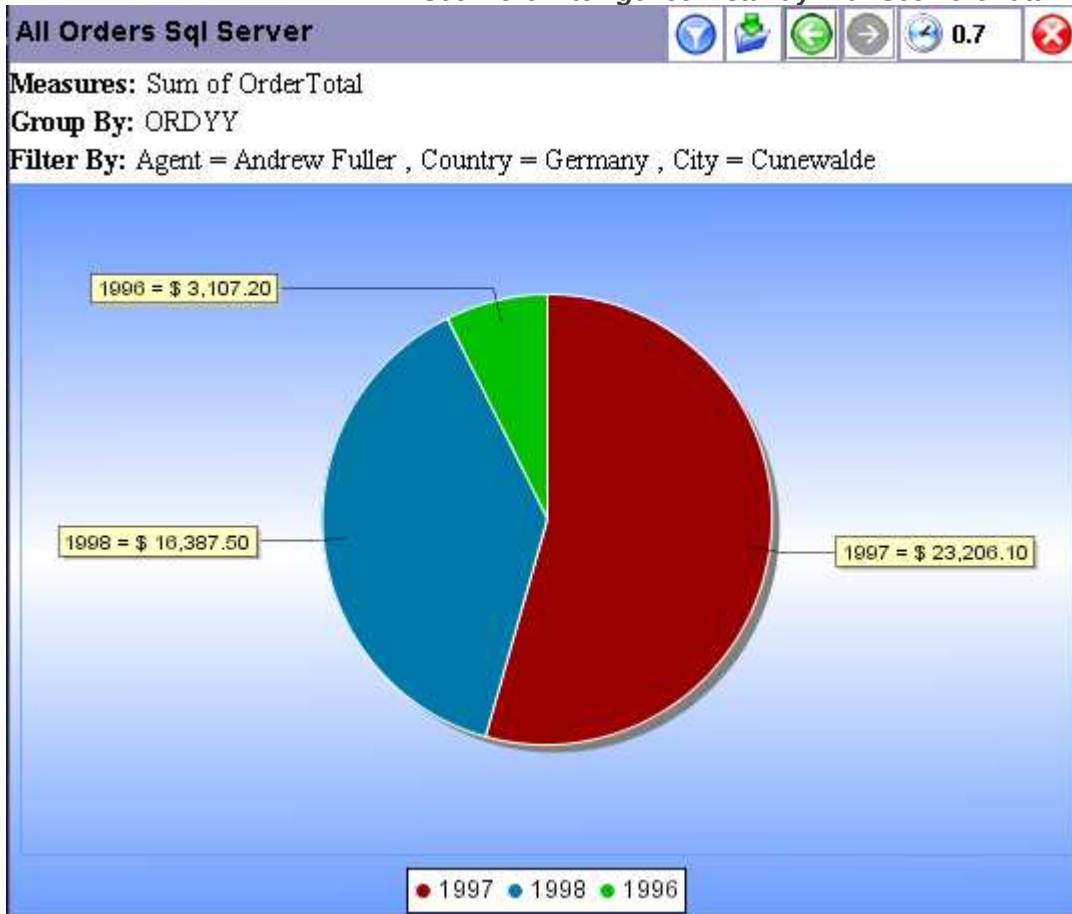


Figure 14: show the Total sales per year for city=Cunewalde AND country = Germany and agent = 'Andrew Fuller'

If we now wanted to expose the records that make up the 1998 slice of this pie chart, double-click the 1998 slice and then choose 'Show Record Details' option.

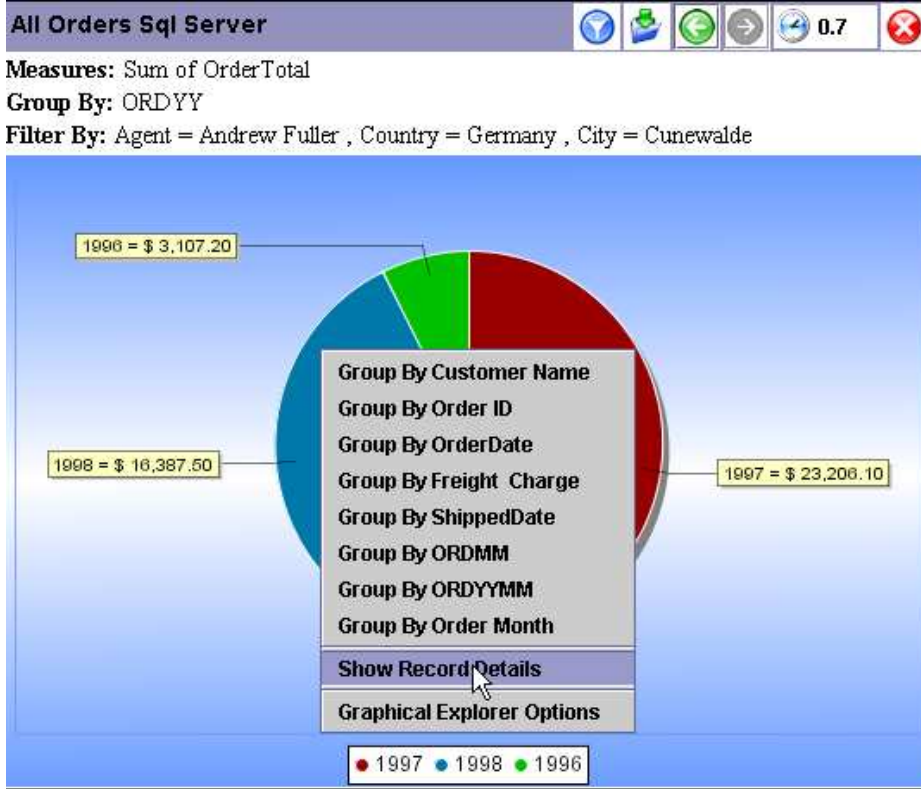


Figure 15: Exposing the data under a slice of the pie.

**All Orders Sql Server** [Navigation icons] 0.7 [Close]

**Filter By:** Agent = Andrew Fuller , Country = Germany , City = Cunewalde , ORDYY = 1998

Country	City	Customer Name	Order ID	Order Total	OrderDate	Freight Charge	ShippedDate	ORDYY	ORDMM	ORDYYMM
Germany	Cunewalde	QUICK-Stop	10865	\$ 16,387.50	02-Feb-1998	\$ 348.14	12-Feb-1998	1998	02	1998 02

Figure 16: drilldown to the detail record/s.

The actual record/s that comprise the grouping from the previous report are now shown.

**See More Intelligence Instantly with SeeMoreData**

Suppose we wanted to see all the records that made up the pie chart, thus, for all of the years (no year filter applied).

This is easily done in SeeMoreData. Simply use the green back button to take you back to the previous report, which was a grouping of total sales by year for the agent, country and city filters applied as a result of previous drilldown operations.

Now ensure that when you double-click to invoke the graphical explorer options, that you do not click any particular pie slice / bar, but click outside the borders of the pie chart / bar graph.

Choose 'Show Record Details'.

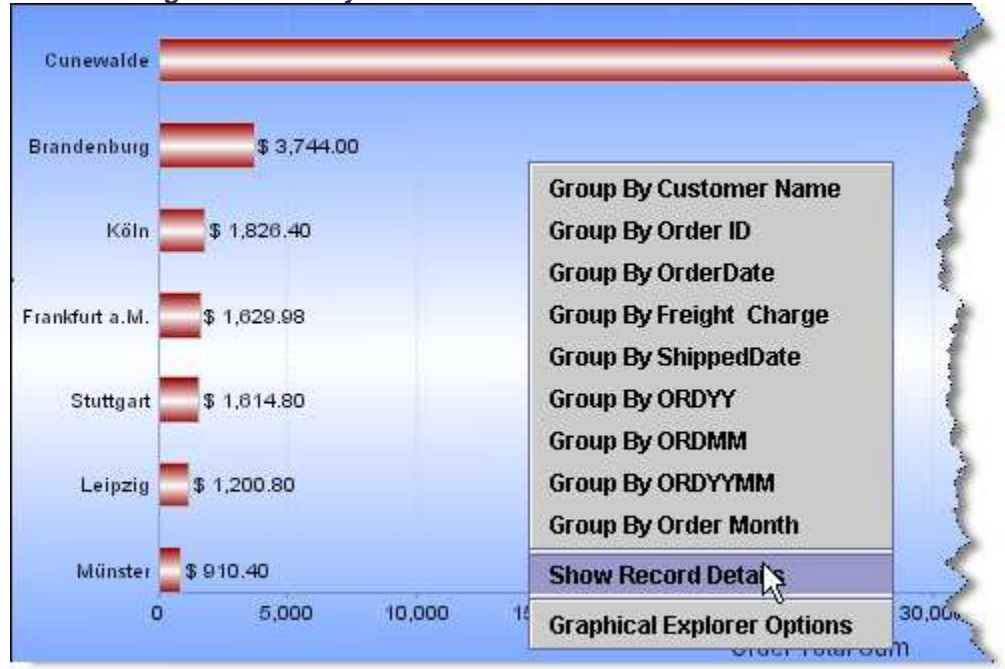


Figure 17: exposing all records for the chart

All Orders Sql Server at 29-Apr-2005 05:49:00AM [Record Count: 6]

Filter By: Agent = Andrew Fuller , Country = Germany , City = Cunewalde

Country	City	Customer Name	Orderid	Order Total	OrderDate	Freight Charge	ShippedDate	ORDYY	ORDM
Germany	Cunewalde	QUICK-Stop	10345	\$ 2,924.80	04-Nov-1996	\$ 249.06	11-Nov-1996	1996	11
Germany	Cunewalde	QUICK-Stop	10515	\$ 9,921.30	23-Apr-1997	\$ 204.47	23-May-1997	1997	04
Germany	Cunewalde	QUICK-Stop	10588	\$ 3,120.00	03-Jul-1997	\$ 194.67	10-Jul-1997	1997	07
Germany	Cunewalde	QUICK-Stop	10691	\$ 10,164.80	03-Oct-1997	\$ 810.05	22-Oct-1997	1997	10
Germany	Cunewalde	QUICK-Stop	10865	\$ 16,387.50	02-Feb-1998	\$ 348.14	12-Feb-1998	1998	02
Germany	Cunewalde	QUICK-Stop	10313	\$ 182.40	24-Sep-1996	\$ 1.96	04-Oct-1996	1996	09

Figure 18: Exposing records for the entire pie

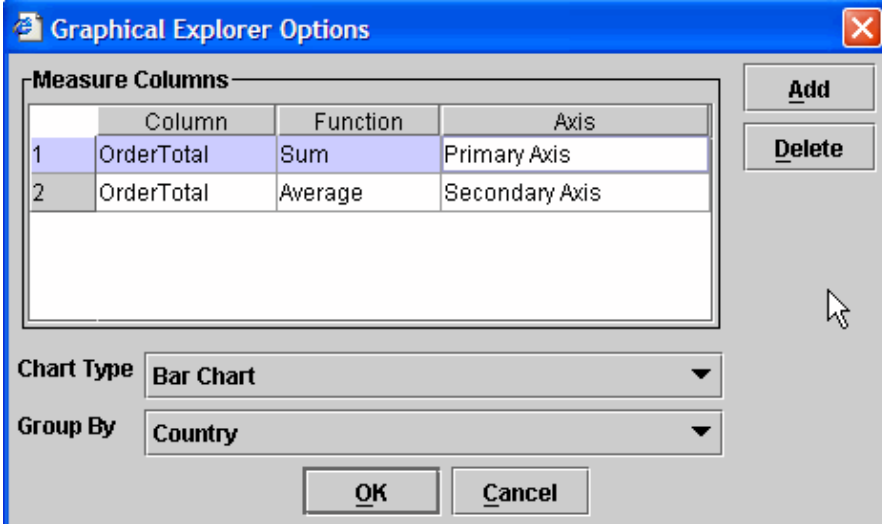
Notice from the highlight, how ALL years are included in the drilldown exposed records.

**Adding additional measures.**

SeeMoreData allows us to configure additional measures to show on the graphs.

If we used the green 'go back' arrow to take us all the way back to the original report (or if it is a quick report, simply just double-click it from the report tree on the left to run it again) and then invoke a different set of grouping attributes, we could show the country's total sales and also the average order per country.





Here we will use two axes to show the total sales per country and average sales per country.

The aggregations that can be used in the function part of the GREX (Graphical Explorer) options are: sum, min, max, count, count unique, count excluding null, average, average including null.

Figure 19: Specifying additional measures on separate chart axes

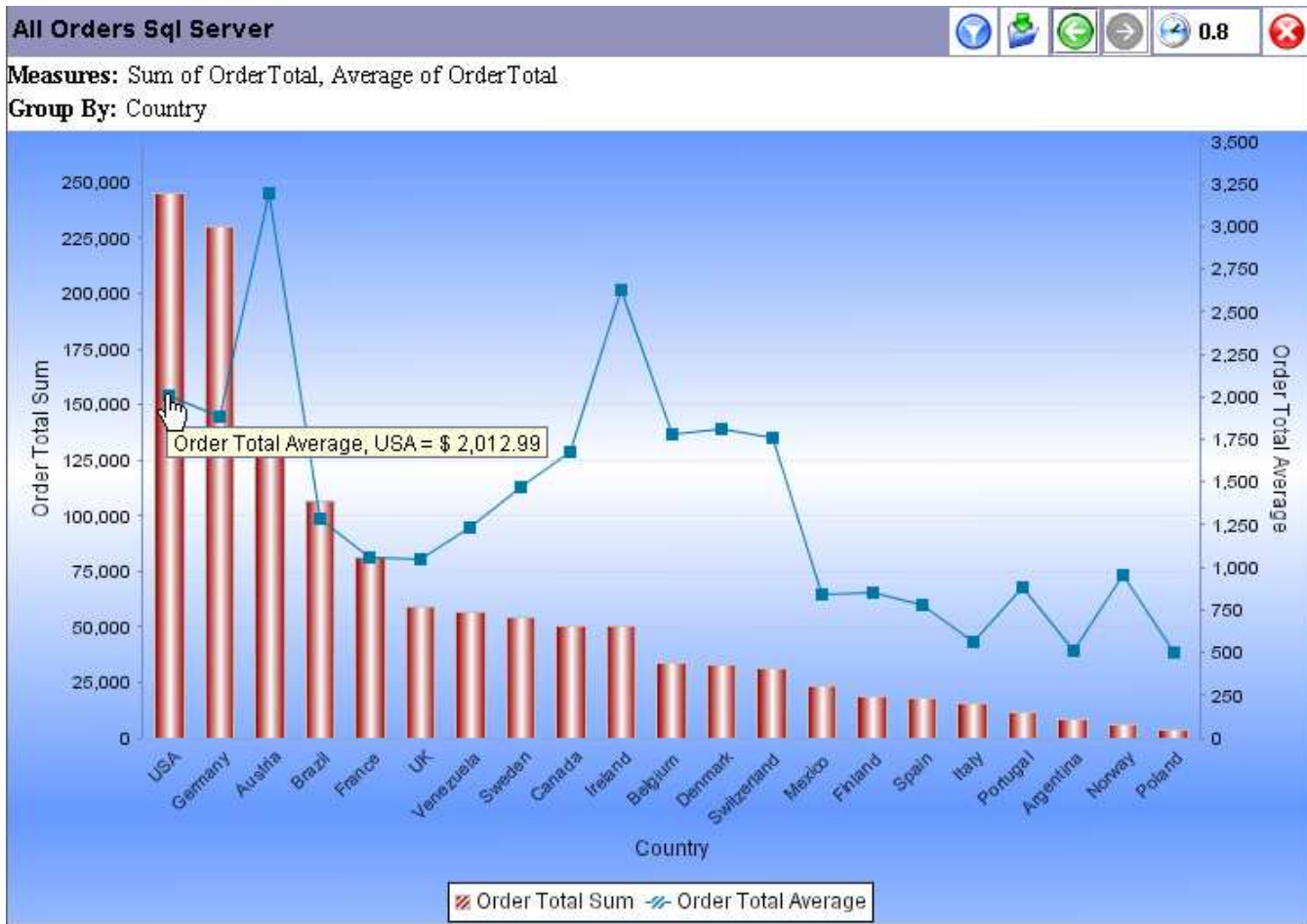


Figure 20: Total and average sales per country, instantly exposing the average order total per country

**See More Intelligence Instantly with SeeMoreData**

From the previous graphic (Figure 20) we can see that whilst USA and Germany may have the highest total sales, it is Austria and Ireland that have recorded the highest average sale per order. Also evident is that Brazil and France represent the median, regardless of total sales, since the plot point intersects the bar approximately at the peak of the bar.

Let us assume for a second that Marketing wanted to see the effects to profitability if shipping (freight) costs were to be included in the price of goods.

A quick changing of the measures would show the average freight costs per country compared to the average sales figure per country.

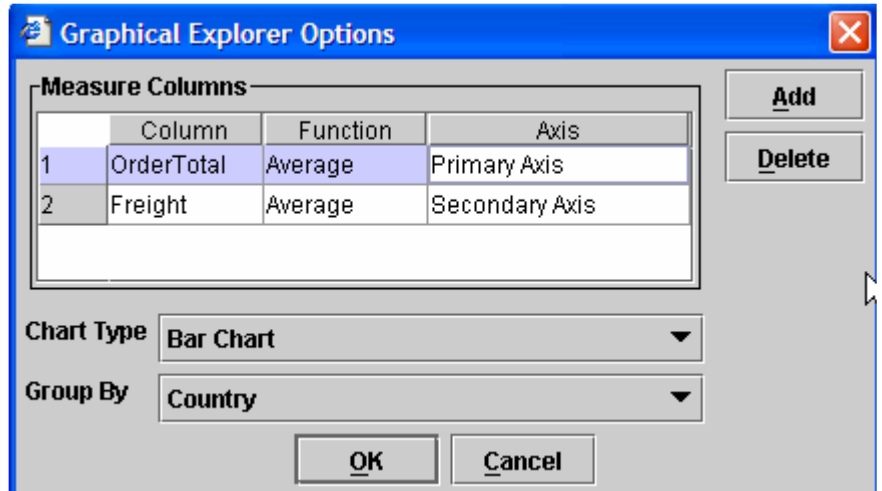


Figure 21: Changing the measure aggregates on the fly



Figure 22: Average Sales per country contrasted with average freight costs per country.

The graph in figure 22 shows that Belgium has the lowest average freight costs per revenue generated and that Freight charges for Austria are proportionately very high.

So if we were to change the measures again, and now track revenue per country against the number of orders per country, we might gain more insight into the trends of our sales data.

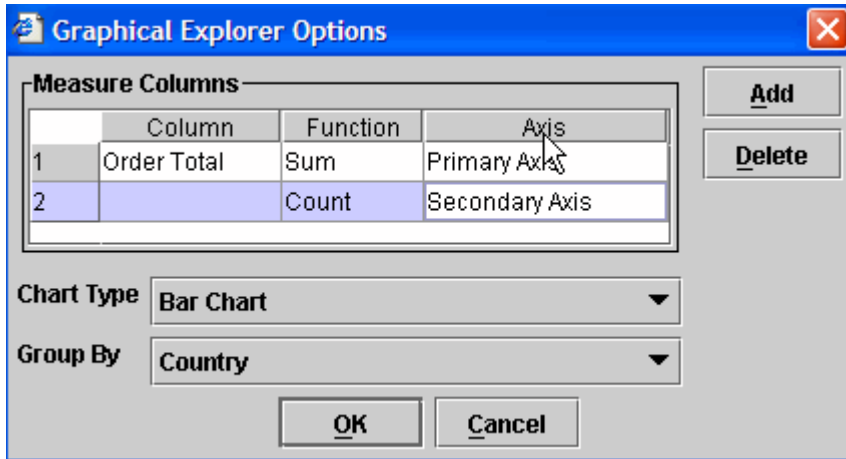


Figure 23: Changing the measures to plot the sum of Orders by country on 1 axis and the count of orders on another.

The graphic below shows a trend towards sales efficiency.



Figure 24: Analysis of Sales / Pricing efficiency per country.

**See More Intelligence Instantly with SeeMoreData**

From this we can see that those bars where the intersection of the “count of Orders” measure is way below the peak of the bar, shows that the sales total was achieved with relatively fewer transactions. Where the 2<sup>nd</sup> line is way above the peak of the bar, this shows that more orders were required to reach the level of sales. In an ideal world, the plot chart measure would intersect the bar at its peak, but this non-uniform pattern shows where sales efficiency is strong and where it is weak.

There may be other factors that contribute towards this picture. For instance, discount information has not been captured in the initial query, but if this were there (and it can be relatively easily added by just modifying the SQL query), then we could further analyze what the average discount is per country, and then drilldown into that country to expose who has been offering steep discounts, and the effect of the discounting policy in profitability.

We could also combine information like cost of sales, supplier information, etc, which for the sake of keeping this White Paper relatively straight forward, has been omitted.

If we now drilldown into Brazil to see the breakdown of total sales by Agent, and the number of transactions (orders) per agent, we can achieve this by double-clicking the Brazil bar and then choosing Agent from the pick list of grouping columns available.

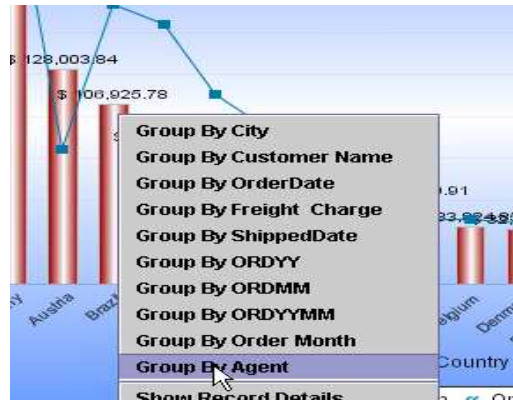


Figure 25: Drilldown on Brazil to group Brazil's sales by Agent.

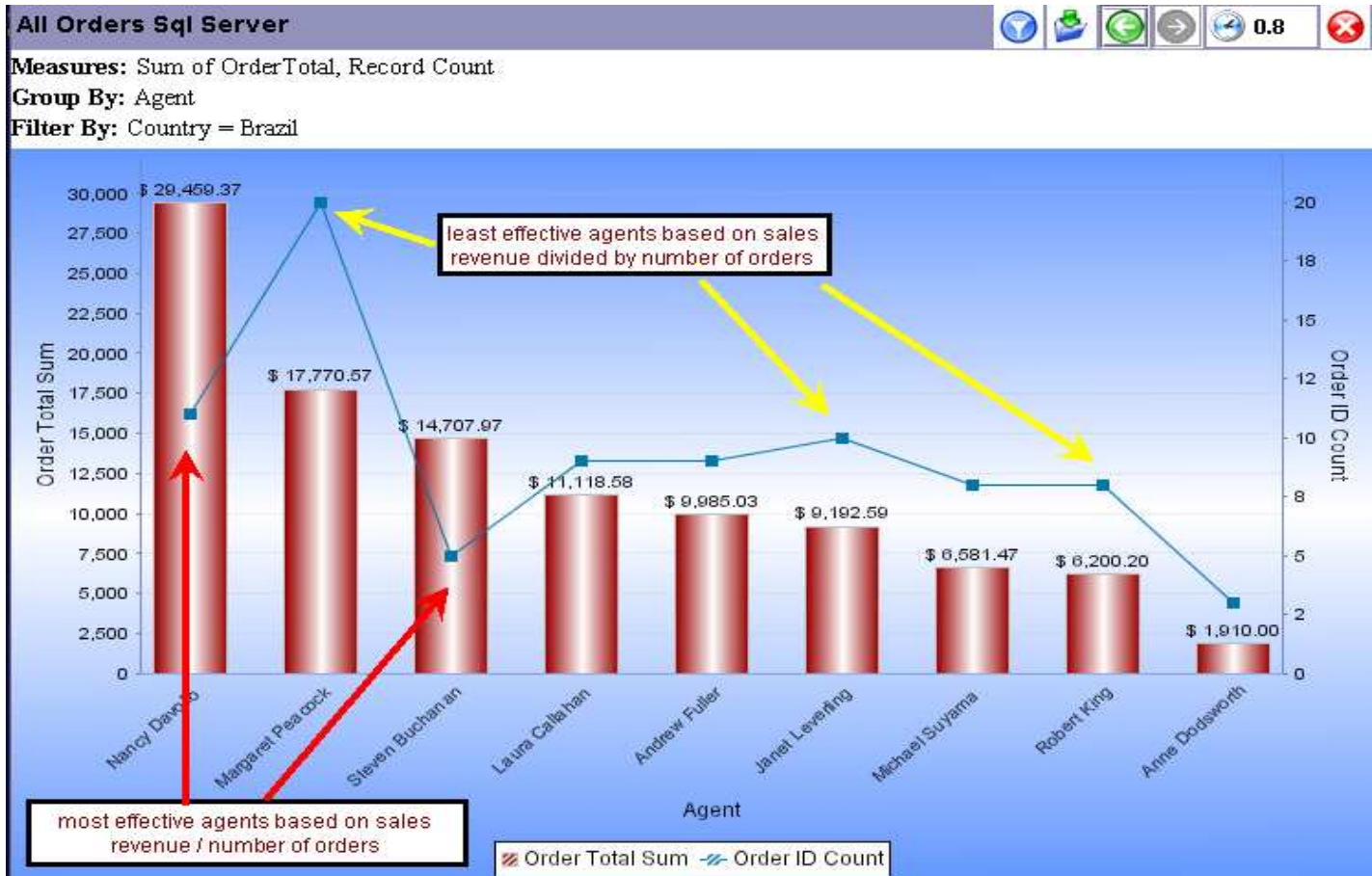


Figure 26: Showing Total Sales and number of orders per agent for Brazil

## ***See More Intelligence Instantly with SeeMoreData***

The above graphic gives us an immediate analytical insight into the sales in Brazil. From this insight, we can see that even though Margaret Peacock has the most number of orders, Nancy De Volio has the highest revenue. We also can see each agent's average sale by adding another measure (average of Order total).

### ***Graphical Explorer Wrap-Up***

So far, all this informational insight and end-user driven ad hoc analysis has been possible from just a reasonably straight forward SQL query producing a "dumb" report that runs against a database in real-time and has some output formatting applied against the resultant data. The query can be parameterized to only show data between certain date ranges or certain product types and then database table indexing will ensure optimal performance of the query.

This data could just as easily have resided in a spreadsheet or a CSV file.

SeeMoreData is not concerned with where the information resides as it supports native query extraction from Excel, CSV, Access, MS Sql Server, DB2 UDB, Oracle, Oracle OLAP, Sql Server OLAP, TeraData, Lotus Domino, Postgre, Informix, Sybase, Adabas D, IMS (with relational middleware), Postgre, ODBC and JDBC compliant relational data sources. What does make SeeMoreData powerful even in data extraction is that it is fully capable of retrieving information from each of these data sources in native format, using SQL, MDX, and vendor-specific stored procedures (Transact SQL, PL/SQL, etc) which can be invoked as easily as using the RDBMS vendor's native query tool.

## Pivot / Crosstab Analysis

While the Graphical Explorer option answers so many different business questions, from time to time, the best way to do comparative analysis is to have data grouped into like columns, creating a pivot on any particular attribute. An example of this might be to show the total sales by agent year on year in a cross-tabular fashion as shown below.

All Orders Sql Server at 29-Apr-2005 11:47:36AM [Record Count: 9]				0.6
Agent	1996	1997	1998	Grand Summary
Andrew Fuller	\$ 21,757.06	\$ 70,444.14	\$ 74,336.56	\$ 166,537.75
Anne Dodsworth	\$ 9,894.52	\$ 26,310.39	\$ 41,103.16	\$ 77,308.07
Janet Leverling	\$ 18,223.96	\$ 108,026.16	\$ 76,562.73	\$ 202,812.84
Laura Callahan	\$ 22,240.12	\$ 56,032.62	\$ 48,589.54	\$ 126,862.28
Margaret Peacock	\$ 49,945.11	\$ 128,809.79	\$ 54,135.94	\$ 232,890.85
Michael Suyama	\$ 16,642.60	\$ 43,126.37	\$ 14,144.15	\$ 73,913.13
Nancy Davolio	\$ 35,764.51	\$ 93,148.08	\$ 63,195.01	\$ 192,107.60
Robert King	\$ 15,232.16	\$ 60,471.20	\$ 48,864.88	\$ 124,568.24
Steven Buchanan	\$ 18,383.92	\$ 30,716.47	\$ 19,691.89	\$ 68,792.28
<b>Grand Sum</b>	<b>\$ 208,083.97</b>	<b>\$ 617,085.20</b>	<b>\$ 440,623.87</b>	<b>\$ 1,265,793.04</b>

Figure 27: Cross Tab Pivot Analysis shows side by side comparison of sales data

Another example might be to have a report showing the Top 3 agents for a given period together with their sales totals.

All Orders Sql Server at 29-Apr-2005 11:47:36AM [Record Count: 1]			0.6
Top 3 agents world wide for this year			
Margaret Peacock	Janet Leverling	Nancy Davolio	
\$ 232,890.85	\$ 202,812.84	\$ 192,107.60	
<b>\$ 232,890.85</b>	<b>\$ 202,812.84</b>	<b>\$ 192,107.60</b>	

Figure 28: Top N instant analysis

We refer to this as Pivot analysis, and the original data contents from the base query now becomes the column heading. Others have referred to these kinds of reports as cross-tab analysis.

What is so significant about SeeMoreData is that each user has the ability to create this kind of reports from the base “dumb” report that simply just churns out the data. By using Pivot Analysis, one can compare sales per agent month by month, easily find the top 3 selling products, find the top 10 customers for a given period without having to invoke very complex cubes or analytical engines, everything the end user needs to make this type of analytical processing is right there in front of them, as part of the standard SeeMoreData application, all using a browser based interface.

To start off, we have the standard “dumb” report as shown on page 2 (figure 1), which simply just extracts the information via the database query and returns the information in a matrix style spreadsheet-like format.

The right-click menu has an option called Pivot Analysis.

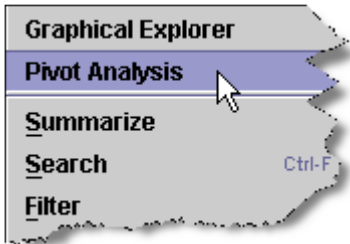


Figure 29: invoke Pivot Analysis from right-click

The Pivot analysis dialogue's basic configuration contains placeholders for the Pivot Column, the Row Fields and the measure fields (the Data Items).

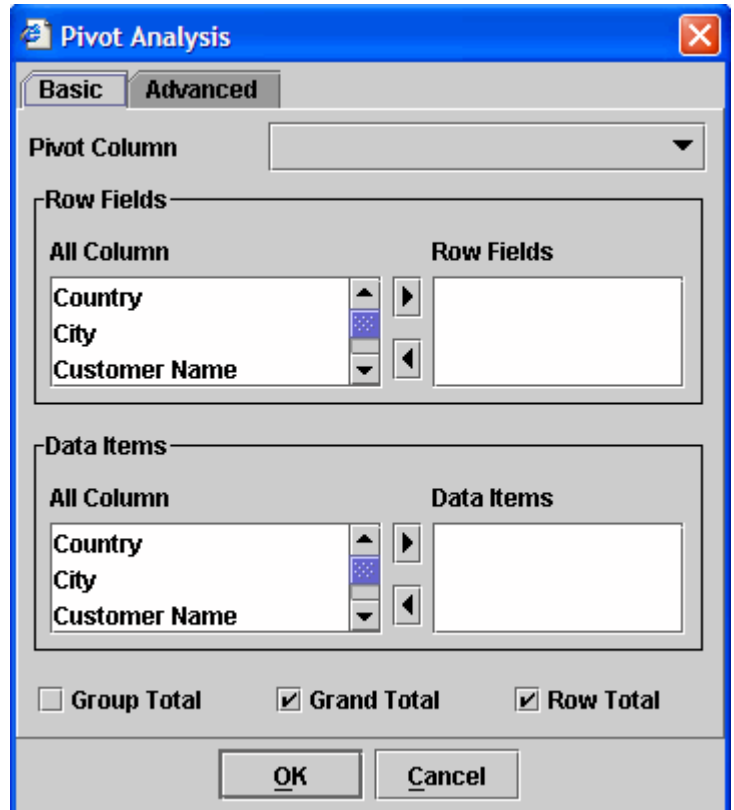


Figure 30: the Pivot Analysis Basic Dialog

For this example, we will create a cross tab report that shows the total of sales per agent by city within country.

The Agent will be the column being pivoted, thus the Agent's name / identified will become the column heading in the cross tab / Pivot section of the report.

The Data Items specify what we are measuring. The default aggregate is sum. Other aggregates can be used too, like count, min, max, average, average including null as zero, count unique, count excluding null.

For the Row Fields, care should be taken to have the correct relationship, as SeeMoreData Pivot Analysis engine will process the columns top down, thus city WITHIN country attribute.

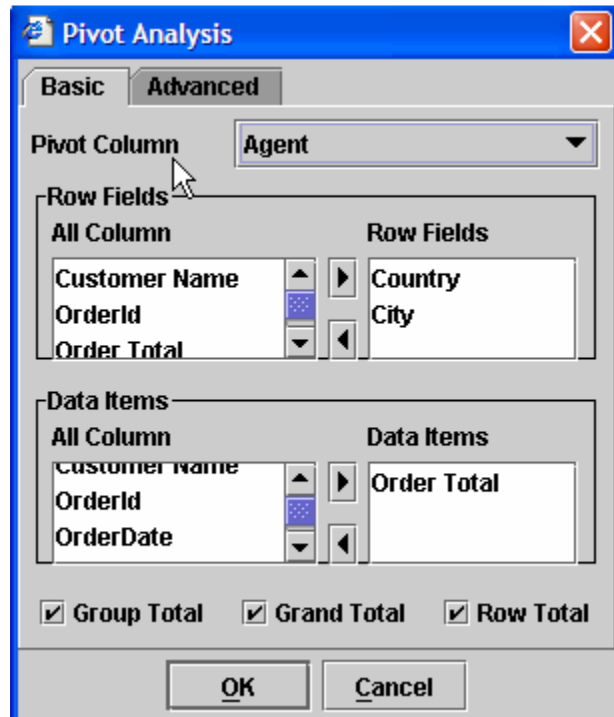


Figure 31: Specifying the Pivot Analysis columns to analyze

All Orders Sql Server at 29-Apr-2005 12:40:14PM [Record Count: 69]

Country	City	Andrew Fuller	Anne Dodsworth	Janet Leverling	Laura Callahan	Margaret Peacock	Michael
Argentina	Buenos Aires	\$ 477.00	\$ 944.50	\$ 319.20	\$ 2,750.50	\$ 1,329.40	
<b>Argentina Sum</b>		<b>\$ 477.00</b>	<b>\$ 944.50</b>	<b>\$ 319.20</b>	<b>\$ 2,750.50</b>	<b>\$ 1,329.40</b>	
Austria	Graz	\$ 10,703.63	\$ 8,967.80	\$ 19,761.35	\$ 10,235.59	\$ 17,308.66	
	Salzburg	\$ 5,899.45		\$ 4,180.00	\$ 735.00	\$ 651.00	
<b>Austria Sum</b>		<b>\$ 16,603.08</b>	<b>\$ 8,967.80</b>	<b>\$ 23,941.35</b>	<b>\$ 10,970.59</b>	<b>\$ 17,959.66</b>	
Belgium	Bruxelles		\$ 1,303.19	\$ 295.38		\$ 5,007.00	
	Charleroi	\$ 2,866.50	\$ 1,505.18			\$ 8,590.20	
<b>Belgium Sum</b>		<b>\$ 2,866.50</b>	<b>\$ 1,505.18</b>			<b>\$ 13,597.20</b>	
Brazil	Campinas	\$ 1,498.35				\$ 1,119.93	
	Resende					\$ 1,812.51	
	Rio de Janeiro	\$ 6,966.40	\$ 1,402.00	\$ 4,184.00	\$ 4,271.52	\$ 7,438.00	
	Sao Paulo	\$ 1,520.28	\$ 166.00	\$ 1,137.55	\$ 6,847.06	\$ 7,400.12	
<b>Brazil Sum</b>		<b>\$ 9,985.03</b>	<b>\$ 1,910.00</b>	<b>\$ 9,192.59</b>	<b>\$ 11,118.58</b>	<b>\$ 17,770.57</b>	
Canada	Montréal	\$ 3,354.00		\$ 5,660.03	\$ 1,278.40	\$ 1,101.00	
	Tsawassen	\$ 5,400.00		\$ 218.70		\$ 3,725.05	
	Vancouver	\$ 2,000.00		\$ 278.00			
<b>Canada Sum</b>		<b>\$ 9,034.00</b>		<b>\$ 6,156.73</b>	<b>\$ 1,278.40</b>	<b>\$ 4,826.05</b>	
Denmark	Kobenhavn	\$ 1,405.20				\$ 14,131.21	
	Århus	\$ 940.50		\$ 1,684.27	\$ 1,814.35	\$ 3,160.60	
<b>Denmark Sum</b>		<b>\$ 2,345.70</b>	<b>\$ 0.00</b>	<b>\$ 1,684.27</b>	<b>\$ 1,814.35</b>	<b>\$ 17,291.81</b>	

Pivot Column is Agent.  
Agent Name is now the header, as it has been pivoted into a cross-tab report.  
The measure is the sum of the Data item/s chosen (in this case 'Order Total').

The Row Fields chosen are displayed in the relationship specified, thus city within country.

Figure 32: Standard Pivot / Cross Tab Report showing total sales per agent by city within country.

The report above shows the total sales per agent for each city, alongside each other for comparative purposes.

But there is more. To expose the sales records for the city of Graz for Andrew Fuller, simply double-click inside the relevant cell to expose the records that comprise the summary aggregation.

All Orders Sql Server at 29-Apr-2005 03:22:33PM [Record Count: 69]

Country	City	Andrew Fuller	Anne Dodsworth	Janet Leverling
Argentina	Buenos Aires	\$ 477.00	\$ 944.50	\$ 319.20
<b>Argentina Sum</b>		<b>\$ 477.00</b>	<b>\$ 944.50</b>	<b>\$ 319.20</b>
Austria	Graz	\$ 10,703.63	\$ 8,967.80	\$ 19,761.35
	Salzburg	\$ 5,899.45		\$ 4,180.00
<b>Austria Sum</b>		<b>\$ 16,603.08</b>	<b>\$ 8,967.80</b>	<b>\$ 23,941.35</b>
Belgium	Bruxelles		\$ 1,303.19	\$ 295.38
	Charleroi	\$ 2,866.50	\$ 1,505.18	

double-click here!

Figure 33: Drilldown into the detail for Andrew Fuller for the city of Graz



All Orders Sql Server at 29-Apr-2005 03:26:20PM [Record Count: 3]						
Country	City	Customer Name	Agent	OrderId	Order Total	OrderDate
Austria	Graz	Ernst Handel	Andrew Fuller	10368	\$ 1,689.78	29-Nov-1996
Austria	Graz	Ernst Handel	Andrew Fuller	10595	\$ 4,725.00	10-Jul-1997
Austria	Graz	Ernst Handel	Andrew Fuller	10990	\$ 4,288.85	01-Apr-1998

Figure 34: All orders for Andrew Fuller for the city of Graz.

## Finding the Top 5 Customers (by revenue) for a period

SeeMoreData Pivot Analysis provides a very fast way to find the top N customers within a given dataset.

Starting again from the standard “All Orders” “dumb” report, we invoke a filter and select only those transactions that occurred in 1997.

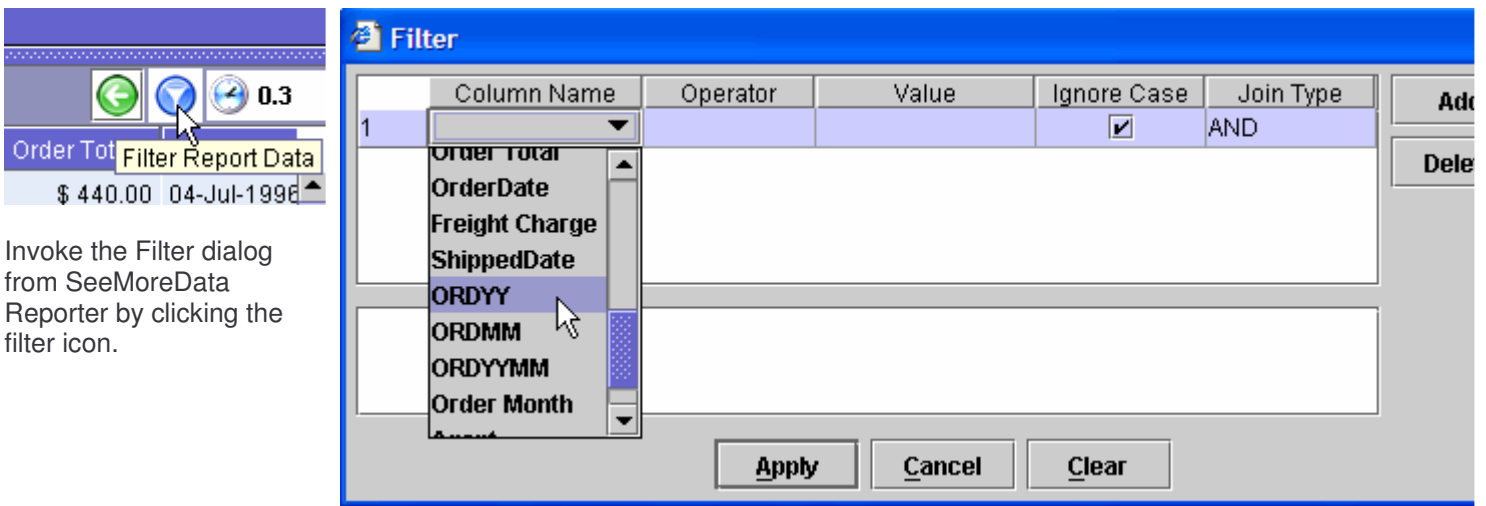


Figure 35: Choose the column name from the filter dialog column pick list

Apply the value of 1997 in the Value field, and then click the ‘Apply’ button. The filter is applied and only records with ORDYY = ‘1997’ are shown.

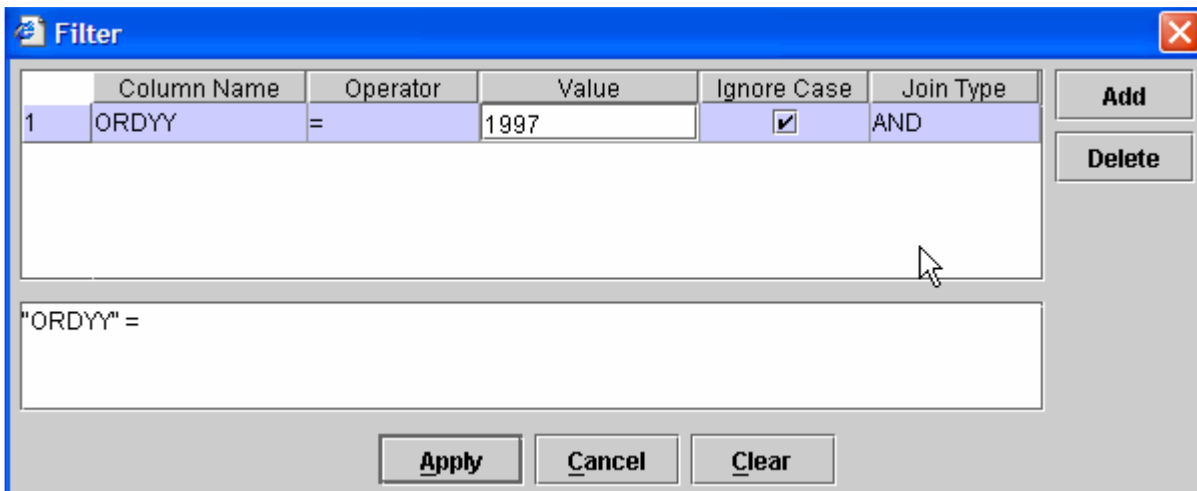


Figure 36: Specifying the filter condition/s

Now we invoke the Pivot Analysis and on the Basic tab of the Pivot Analysis dialog.

We specify that we want to pivot the sum of values (Order Total) by Customer Name, thus there will be a total for each customer. Of course there could be thousands and thousands of customers, so we will limit this to the top 5 in the next step, from the Advanced tab of the Pivot Analysis dialog.

For now, we do not want any row fields; we simply want to display the total of Order total by customer, so no row fields are necessary.

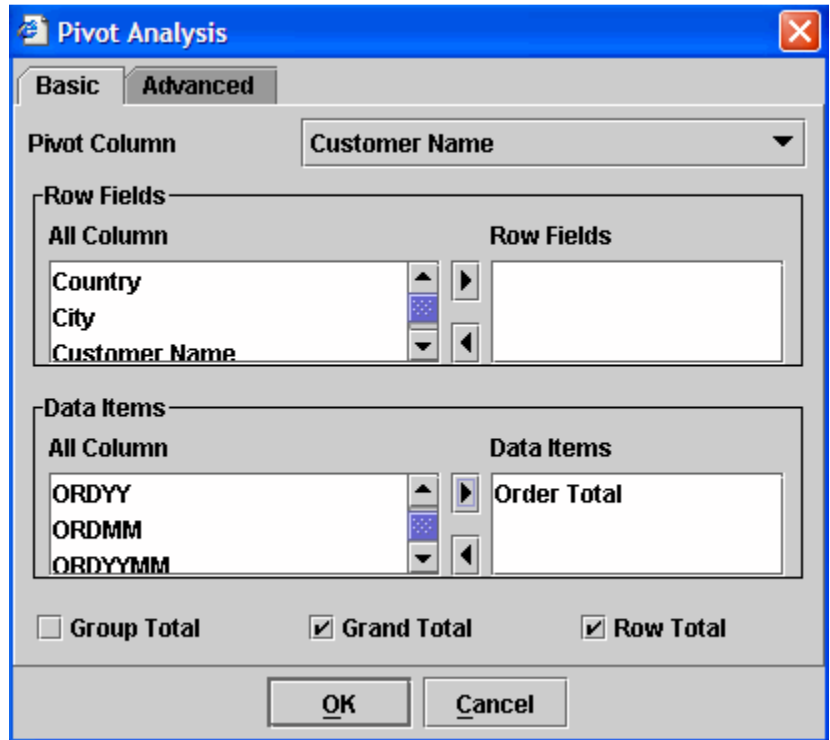


Figure 37: Basic Pivot Analysis run-time parameters

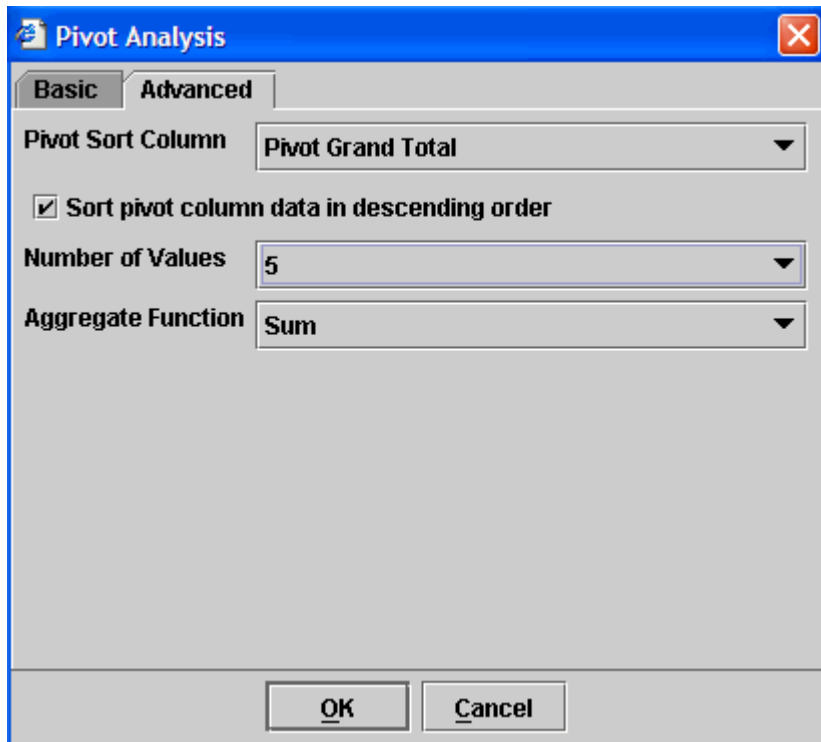


Figure 38: Specifying the sort order, the direction and number of values.

Here on the Advanced Tab of the Pivot Analysis dialog, we specify that we would like to sort the pivoted values in descending order, by the grand total. There is only 1 line of totals, since no row Fields were specified.

We then change the Number of Values to be displayed, from the default of 'All' to 5. If we do not specify 5 here, we will get values for every single customer, and depending on the dataset, there could be thousands. We are only interested in the top 5 customers based on revenue (Order Totals selection from the Data Item of the Basic tab) for the reporting period.

Leave the aggregation of sum alone, we want to add the total orders per customer. Other available aggregations are min, max, count, count unique, count excluding null, average, average including null (treat null as zero).

All Orders Sql Server at 29-Apr-2005 03:51:30PM [Record Count: 1]				
QUICK-Stop	Save-a-lot Markets	Ernst Handel	Mère Paillarde	Hungry Owl All-Night Grocers
\$ 61,109.91	\$ 57,713.58	\$ 48,096.26	\$ 23,332.31	\$ 20,454.41
<b>\$ 61,109.91</b>	<b>\$ 57,713.58</b>	<b>\$ 48,096.26</b>	<b>\$ 23,332.31</b>	<b>\$ 20,454.41</b>

Figure 39: Top 5 customers for 1997

The report above is very simplistic but to the point and shows that from an ordinary tabular report that could span possibly thousands and thousands of rows, within 60 seconds or less, using SeeMoreData, we were able to identify the top 5 (or Top N) customers for the period being reported on. Note how we specified the sort order is to be on the Grand total in descending order and limited to 5 pivot values. That is exactly what is seen above.

If we now double-clicked the data cell for 'QUICK-Stop', we would drilldown to expose the orders from the original "All Orders" report that comprised the aggregate (sum of) value.

All Orders Sql Server at 29-Apr-2005 03:51:30PM [Record Count: 14]						
Country	City	Customer Name	OrderId	Order Total	OrderDate	Freight Charge
Germany	Cunewalde	QUICK-Stop	10418	\$ 1,814.80	17-Jan-1997	\$ 17.55
Germany	Cunewalde	QUICK-Stop	10451	\$ 3,849.66	19-Feb-1997	\$ 189.09
Germany	Cunewalde	QUICK-Stop	10515	\$ 9,921.30	23-Apr-1997	\$ 204.47
Germany	Cunewalde	QUICK-Stop	10527	\$ 1,503.00	05-May-1997	\$ 41.90
Germany	Cunewalde	QUICK-Stop	10540	\$ 10,191.70	19-May-1997	\$ 1,007.64
Germany	Cunewalde	QUICK-Stop	10549	\$ 3,554.28	27-May-1997	\$ 171.24
Germany	Cunewalde	QUICK-Stop	10588	\$ 3,120.00	03-Jul-1997	\$ 194.63
Germany	Cunewalde	QUICK-Stop	10658	\$ 4,464.60	05-Sep-1997	\$ 364.15
Germany	Cunewalde	QUICK-Stop	10691	\$ 10,164.80	03-Oct-1997	\$ 810.05
Germany	Cunewalde	QUICK-Stop	10694	\$ 4,825.00	06-Oct-1997	\$ 398.36
Germany	Cunewalde	QUICK-Stop	10721	\$ 923.88	29-Oct-1997	\$ 48.92
Germany	Cunewalde	QUICK-Stop	10745	\$ 4,529.80	18-Nov-1997	\$ 3.52
Germany	Cunewalde	QUICK-Stop	10765	\$ 1,515.60	04-Dec-1997	\$ 42.74

Figure 40: The details of the Orders for QUICK-Stop customer.

## Conclusion

This white paper has illustrated the use of two of the analytic functions available to all SeeMoreData users at run-time, which can be learned relatively easily. These powerful analytic functions are:

- SeeMoreData Graphical Explorer
- SeeMoreData Pivot Analysis

The examples shown above are from a single report that was created in about 10 minutes running against data from Microsoft's Sql Server Northwind demo database,

The built-in intelligence that is described herein is available to all SeeMoreData reports.