dun & bradstreet

Market Insight

Base Advanced Module

Training Manual v3.2

D&B Market Insight

Base Advanced Module

Manual Version: 3.2

Software Version: 2016 Q4

System: Training (US)

D&B Market Insight is powered by
FastStats Technology from Apteco Ltd

Contents

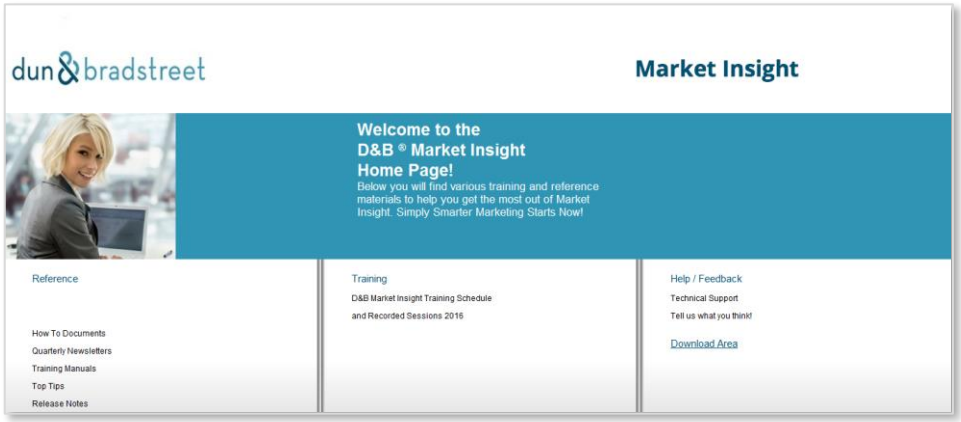
Introduction

Market Insight provides powerful and interactive marketing analysis of customer data overlaid on a D&B data universe. The system is web based with a truly easy to use Windows interface. Using a consistent and intuitive "drag and drop" approach throughout, every action automatically results in a query that can be saved and reused with ease. With a wide range of descriptive and predictive analytical tools, Market Insight's analysis options are virtually unlimited as any technique can be applied to any results in any order. Market Insight provides a unique combination of speed, power and accessibility for data exploration and understanding.

Market Insight holds your data overlaid on a D&B universe. This enables you to accurately measure your customer data in proportion to the opportunities in the market place. Hence the product's name: it enables insight of your activities in comparison to the market place rather than just within your business.

The D&B data universe in your Market Insight system will be adjusted to suit your licensing and measurement requirements. Your customer data is loaded from extract file(s) you provide and although this process allows for some cleaning and manipulation of the data, what you see within Market Insight is a reflection of the data you provide.

The Market Insight view of the data is a snapshot at the time that the data was loaded. Market Insight is an analytical system able to provide insight and understanding but it can also provide data feeds to your operational marketing systems to implement your targeting decisions.



Market Insight Homepage

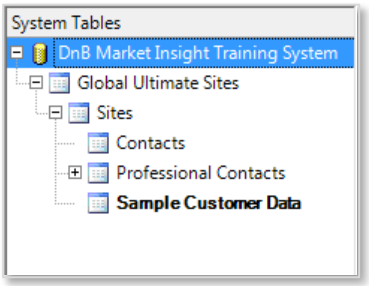
N.B. The counts and figures in this manual may differ to those seen when you use this system as the data changes over time. Not all the functionality shown in this manual may be available in the system you are using.

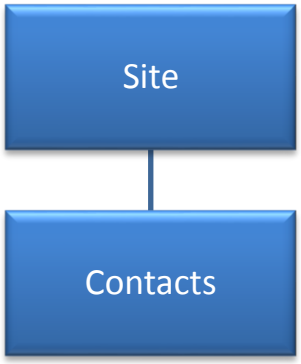
Data Structure

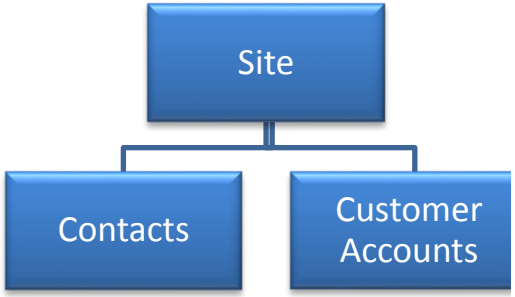
The structure of your Market Insight system can vary. The elements shown here are typical – each Site may be simply flagged with Customer data or can have many related Contacts. A Site may also have many matched Customer Accounts, and / or many other types of related data (such as one or more URLs, UCC Filings and Corporate Family Linkage Plus entries, etc.) The data loaded for each matched Customer Account is configurable – for example you may have multiple Transactions or Divisional Summaries or Product Summaries etc.

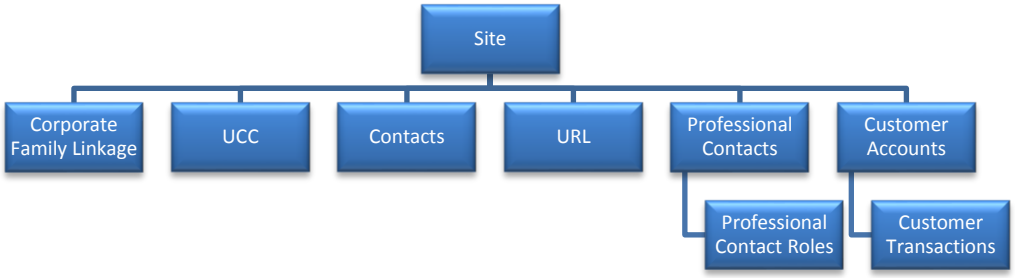
The detail present on each table of data depends on the Market Insight administrator. The data is arranged into folders to assist the user to navigate and find data items.

The structure used in the Training System, illustrated in this manual, uses a simple structure that has sample customer data and contact tables that are linked to a Sites table and these sites are in turn linked to the Global Ultimate Sites table (e.g. worldwide headquarters).









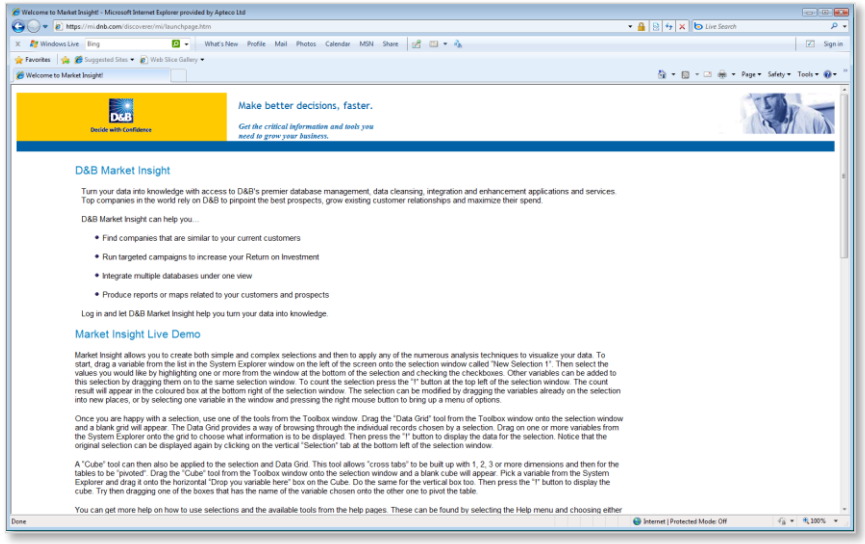
Accessing Market Insight

The Market Insight software is downloaded automatically to your PC when you click a link to launch the system. Once the software has been downloaded, it will automatically update from the server whenever necessary.

To access Market Insight you need:

- Windows PC – Market Insight is a Windows.NET application that combines the best of the Windows interface with web based systems. Market Insight is not available on Mac or UNIX computers

- The latest Windows.NET framework version installed. This can be obtained by visiting www.windowsupdate.com or from your IT team

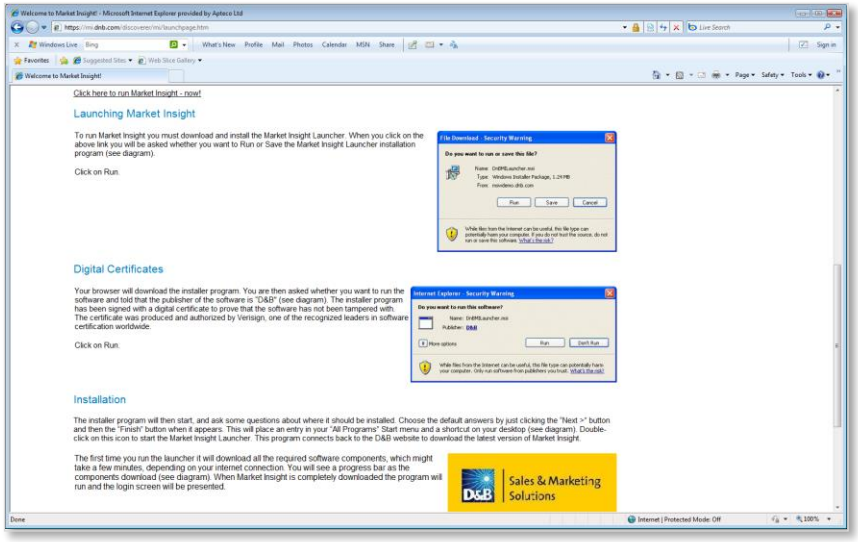


To launch your Market Insight system, use a browser to view:

https://mi.dnb.com/discoverer/mi/launchpage.htm

N.B. The “https” prefix, which establishes a secure connection between your browser and the D&B Server.

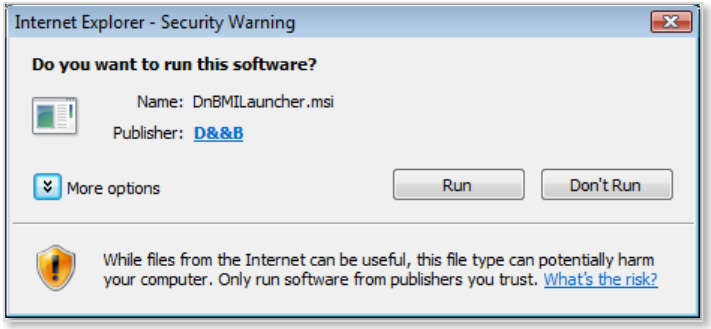
- Click on the **“Click here to run Market Insight”** link to launch the software installer



Instructions to Download Launcher

You can elect to save the file or run it directly from the web site.

- If you elected to save the file, navigate to where you saved and double click it. Agree to run when prompted, and then follow the on screen instructions



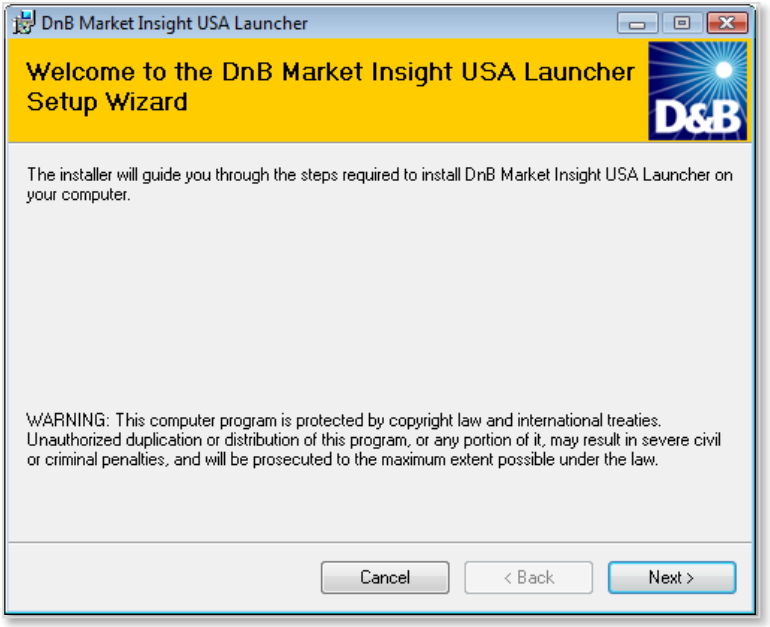
Security Warning Prompt

- The installation process will result in an icon on your desktop and in a D&B Start Menu folder

- On subsequent uses of Market Insight, you can simply double click this icon. The software will automatically update from the D&B server

whenever new releases are made available

- You can install Market Insight on as many computers as you wish – it is your user id that controls your access. This means, for example, you can use Market Insight when working from home



Launcher Setup Wizard

How to Login

To use Market Insight, you need to have an Internet connection.

Start Market Insight by:

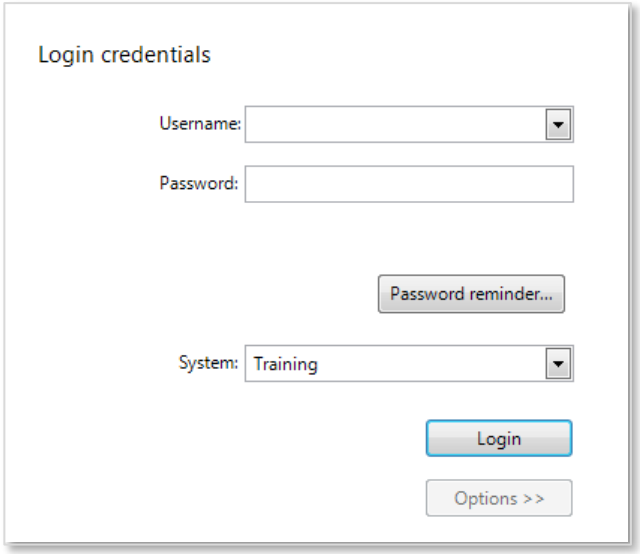
- Clicking on the **Market Insight** icon on your desktop, or
- Navigating via:

  Start → All Programs → DnB → Market Insight

In the upper left hand corner of the screen you will see a Login window that gives you the opportunity to connect to a Market Insight system containing data available to you for analysis.

Enterprise Tab

Your Market Insight system operates on a secure and resilient web connected server enabling you to access the system from any location with an Internet connection. A number of users may access the system at the same time, each of whom is authorized by a user account and password. Your Market Insight Administrator will provide you with a Username and Password.

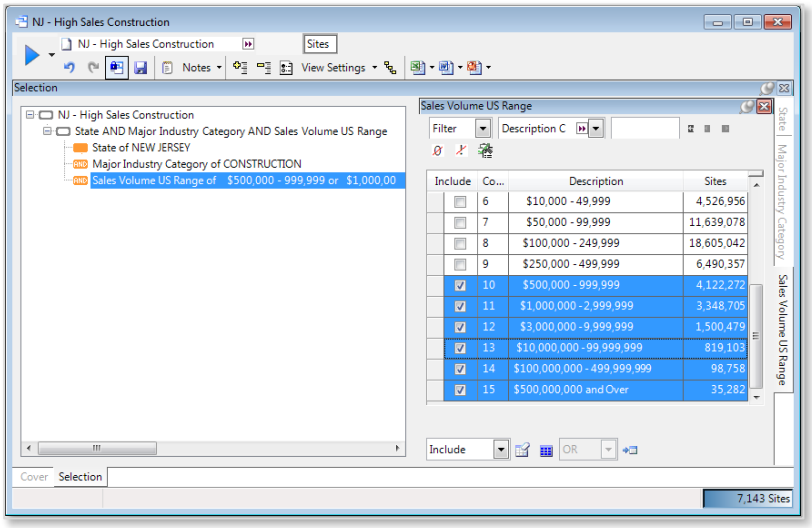


Login Window

Selection: Create and Save

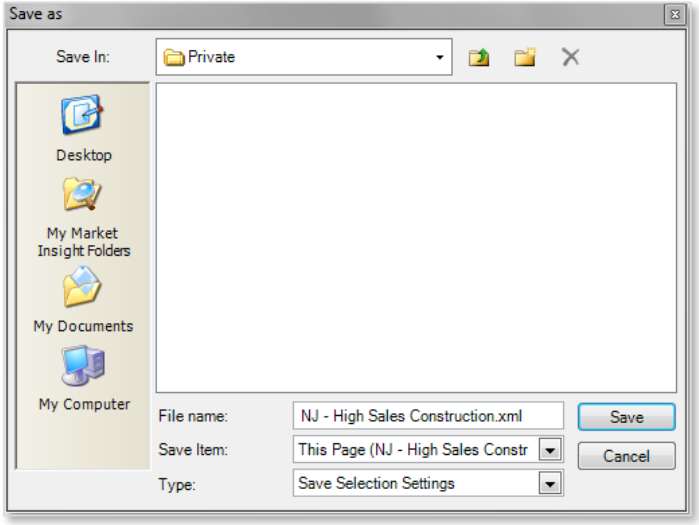
Having worked through the Base Standard manual or as part of your work activities you will be familiar with the creation of selection queries. The following example will be one we can use throughout this course.

- Display a new selection window set to the **Sites** table level
- Within the **System Explorer** search for the following variables and add them to the selection window as shown opposite:
  - **State** – New Jersey
  - **Major Industry Category** – Construction
  - **Sales Volume US Range** – Select all ranges from $500,000-999,999 to $500,000,000 and Over
- Click the **Build** button to count the result



To use this example as the basis for a number of activities in this manual it will be useful to have it saved.

- Rename the selection from **New Selection 1** to **NJ – High Sales Construction**
- Click the **Save** button and navigate to the **Private** folder to set it as the save destination
- Click the **Save** button



This selection query can now be found under the Private folder of the My Market Insight Folders within the File Explorer.

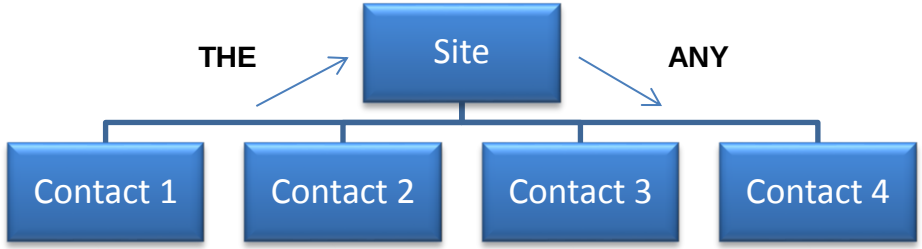
Selection Logic Between Tables

The Structure

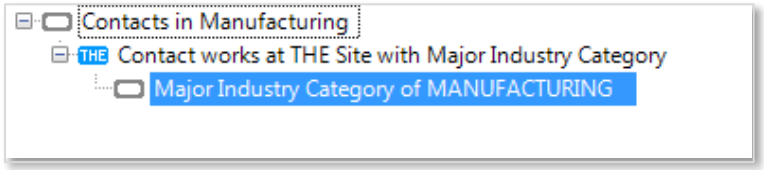
You will already be familiar with the logic operators **AND** & **OR** which are used between variables when building a selection query.

Remember, each selection query has a **Resolve Table** – the query is resolved to select a set of rows from this table.

However, you will have noticed that when you use a variable from a different table to the one set for the window a **THE** or **ANY** precedes the variable. The word used indicates the direction in which the query has to travel, up or down, the table structure.

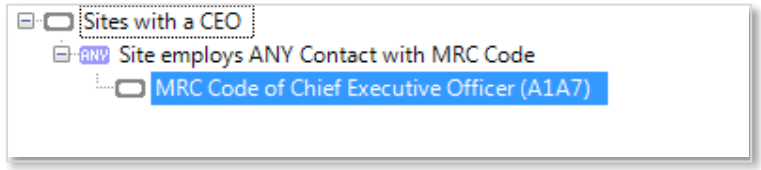


- Open a Selection window set to the **Contacts** table
- Drag on the **Major Industry Category** variable, select **Manufacturing**



You will notice in this example a THE indicator is used. This is because we are moving up the table structure selecting Contacts where THE Site they relate to is manufacturing.

- Open a Selection window set to the **Sites** table
- Drag on the **MRC Code** variable, select **Chief Executive Officer**



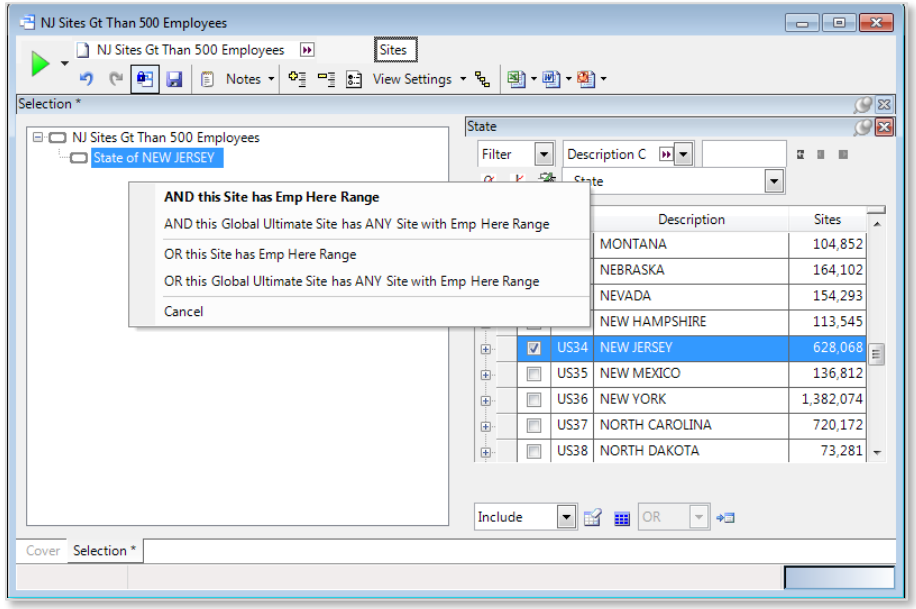
You will notice in this example an ANY indicator is used. This is because we are moving down the table structure, selecting sites that have ANY contact with responsibility code of CEO.

Selection Options

In the examples above you made your selections with a normal left mouse drag. However, if you make a right mouse drag you will be presented with more options than the default logic join of an **AND**.

- Open a Selection window, set to the **Sites** table, then drag on the **State** variable & select **New Jersey** then right drag the **Emp Here Range** variable beneath **State**

The popup menu now presents you with all the possible options based around the logic operators and any tables above and including the one you are counting at. For illustration, assume we pick 500 and over:



Option 1 – AND this Site has Emp Here Range (Default)

This will select all New Jersey Sites **AND** have 500 and over employees.

Option 2 – AND this GUS has ANY Site with Emp Here Range

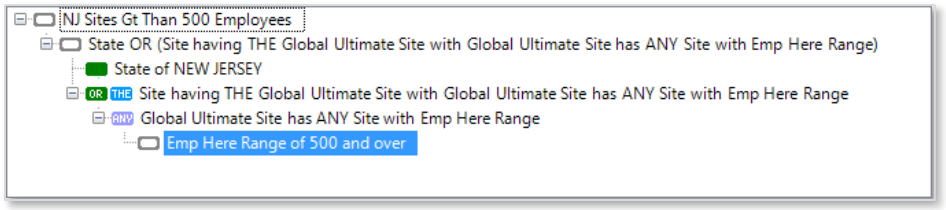
This will select all New Jersey Sites **AND** where it's GUS has any Sites with 500 or over employees. Note that the same Site could meet both these criteria and satisfy the query.

Option 3 - OR this Site has Emp Here Range

This will select all New Jersey Sites **OR** Sites with 500 or over employees. A Site could meet both of these criteria and satisfy the query, but will still only be counted once.

Option 4 – OR this GUS has ANY Site with Emp Here Range

This will select all New Jersey Sites **OR** all Sites where its GUS have any Sites with 500 or over employees.

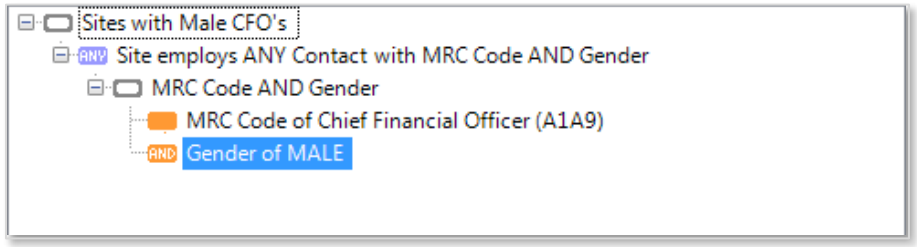


OR this GUS has ANY Site with Emp Here Range

Variable Position

It is important to be aware of the effect that the positioning of a variable will have on the final selection.

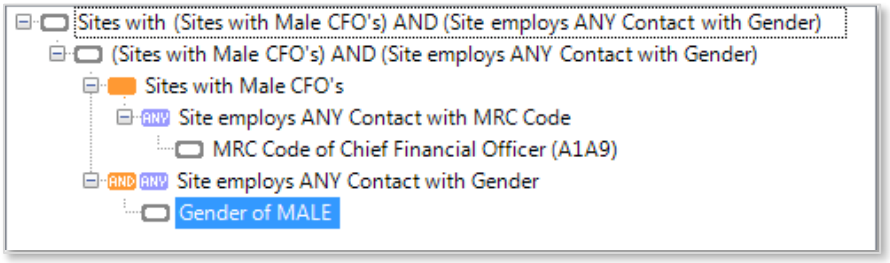
- Open a new selection window set to the **Site** table
- Drag on the **MRC Code** variable and select **Chief Financial Officer**
- Drag on the **Gender** variable and drop it underneath the **MRC Code** variable. Select **Male**
- Click the **Build** button



In this example the MRC and Gender variable are tightly bound together. The query will therefore return all sites that have a contact that meets both criteria i.e. a Male CFO.

Let's see what happens when you reposition one of the variables.

- Click and drag the **Gender** variable above the row that contains the **ANY**
- When the black line appears extend it to the left and let go of the mouse button
- Click the **Build** button



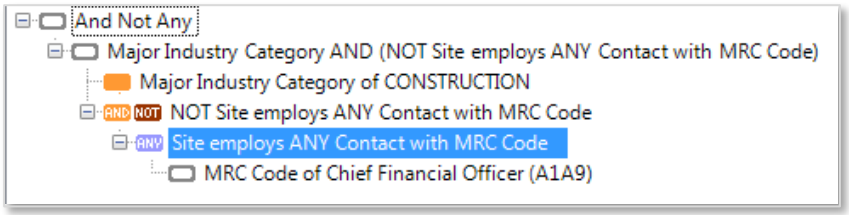
This new example has a more loose association between the MRC and Gender variables. The query will therefore return all Sites where there is at least one CFO contact and at least one Male contact, but they no longer need to be the same contact.

NOT Operator Position

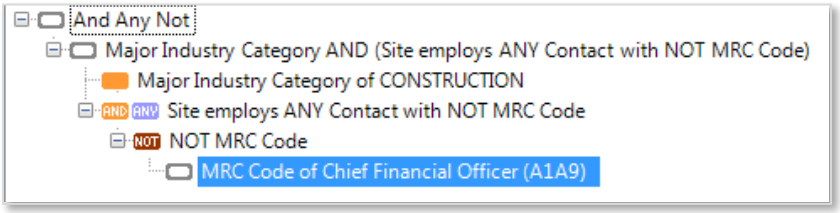
It is important to understand that where you place your NOT operator in the logic structure can affect the question asked and subsequently the results given.

- Create a selection of **Major Industry Category** of **Construction** and **MRC Code** of **Chief Financial Officer**
- Right click on the line with the **AND** operator and select **NOT**
- Click the **Build** button

With the combination of the **AND NOT ANY** this selection will return Construction Sites who have not got a CFO contact.

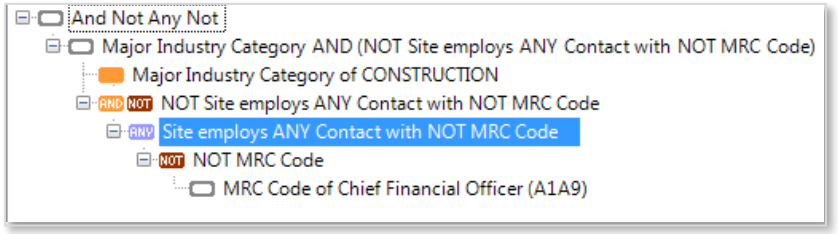


- Recreate the selection of **Major Industry Category** of **Construction** and **MRC Code** of **Chief Financial Officer**
- Right click on the line **MRC Code** of **Chief Financial Officer** and select **NOT**
- Click the **Build** button



With the combination of the **AND ANY NOT** this selection will return Construction Sites which have Contacts other than a CFO. This means some of these Sites will also have a CFO as well as other contacts, hence a higher result with this selection than the previous one.

If you place a **NOT** on both the lines mentioned above this will return Construction Sites which only have a CFO contact!



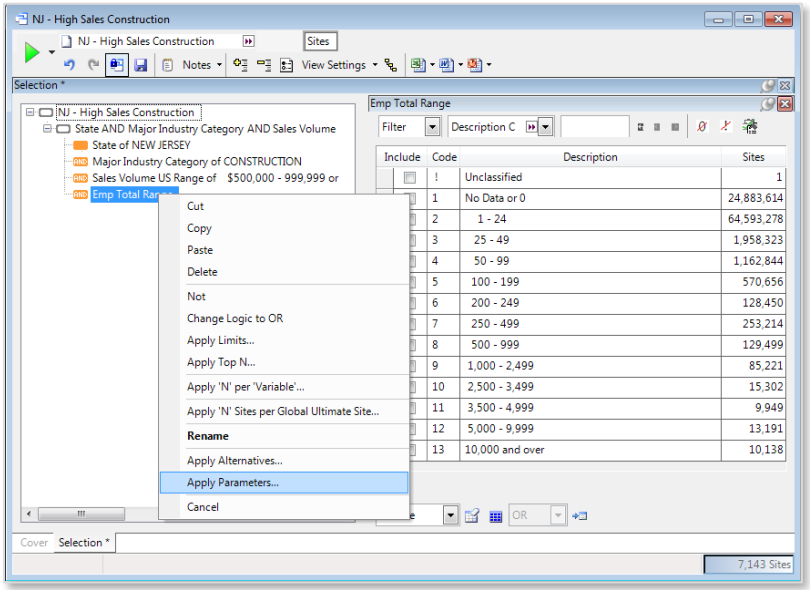
Parameterised Selections

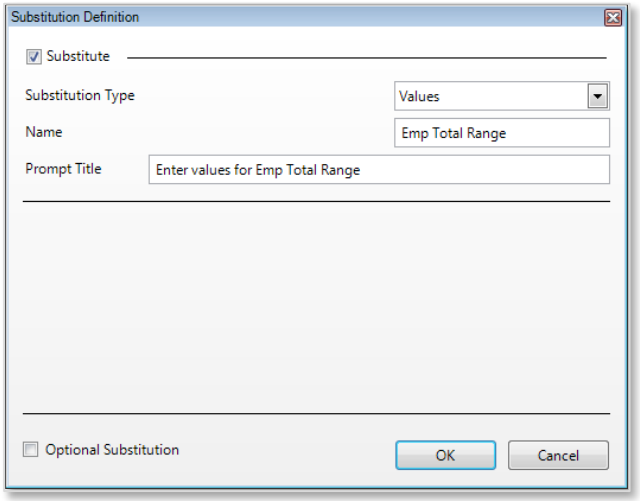
When creating a selection you may wish to mark certain parts of it that allow for User input. This could involve creating a complex selection with one or more elements that give the User the chance to determine the values used. For example a complex selection to determine a group of sites might be further examined by their total employees. The User would then be prompted to choose the employee range without being concerned with the rest of the pre created selection.

In the example opposite the selection of NJ – High Sales Construction will have the Employee Total Range element parameterised to allow other users to determine which range they want to apply to this selection.

- Display the saved selection **NJ – High Sales Construction** and add the **Emp Total Range** variable

- Right click on the variable to be parameterised (Destination) and click on the option **Apply Parameters...** from the popup menu





The **Substitution Definition** window allows you to firstly select the **Substitution Type** which could be to choose new **Values** or a new **Variable & Values**. The other options are:

Name	This will be the description that the User will see. This can be amended here.
Prompt	This will be the prompt description the User will see. This can be amended here.
Optional Substitution	Ticking this box will activate the Remove from the selection tick box on the Substitution Values window. This allows the User to ignore the parameterised part of the selection query.

In this example the default options will be used.

- Click **OK**. Now run the selection by clicking on the **Build** button. Select **500 - 999** and then **OK**

- From the **Cover** page click on the **Substitutions** tab

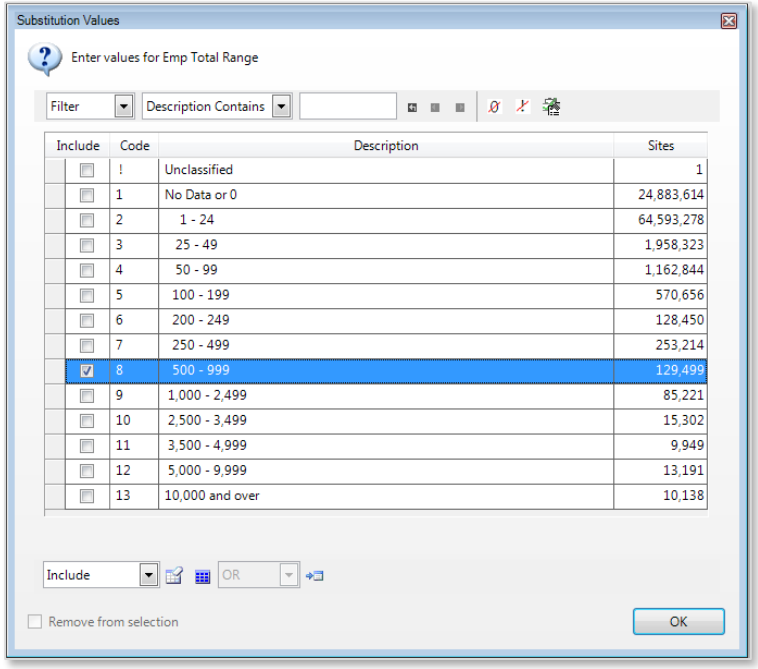
If this selection is to be saved and used again you may want to set the Behaviour of how the Users interact with the selection.

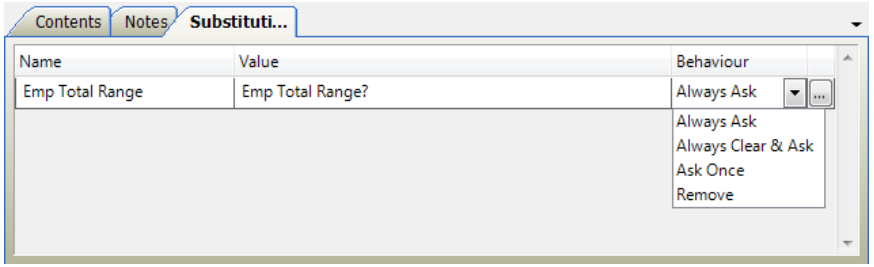
Always Ask	Every time the selection is used the User will be prompted to see if a parameter needs to be changed.
Always Clear & Ask	As above but with any existing parameter cleared.
Ask Once	After the first prompt to set or change the parameter that choice will be set as the default.
Remove	This option will ignore the parameterised part of the selection.

- To change the **Destination** category for the selection click on the button. This will redisplay the **Substitution Values** window

- Change the values and click **OK** followed by the **Build** button to see the new result

N.B. Save your Parameterised selection by dragging the book icon into a read only folder to ensure multiple users can access it without altering its set up.





Recency Frequency Value

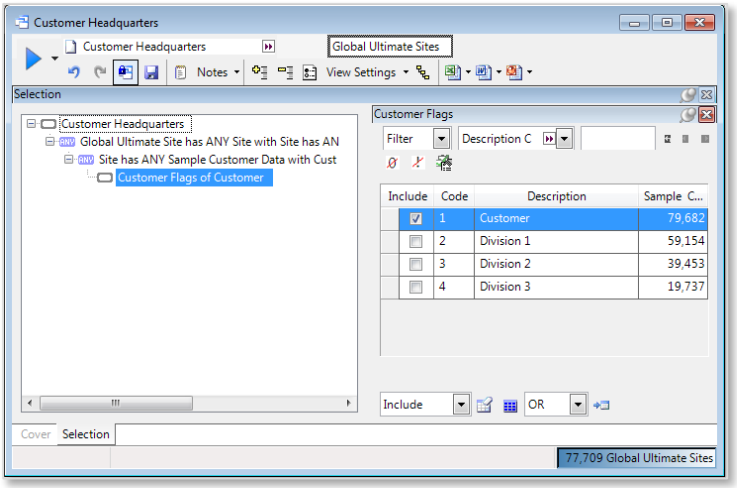
In some situations it is not sufficient to be able to select on the presence or absence of data. This may be the case when you wish to select on the number of times that a particular type of data appears, or to perform averages across data records. In this situation, you will need to use the MI RFV selection mode.

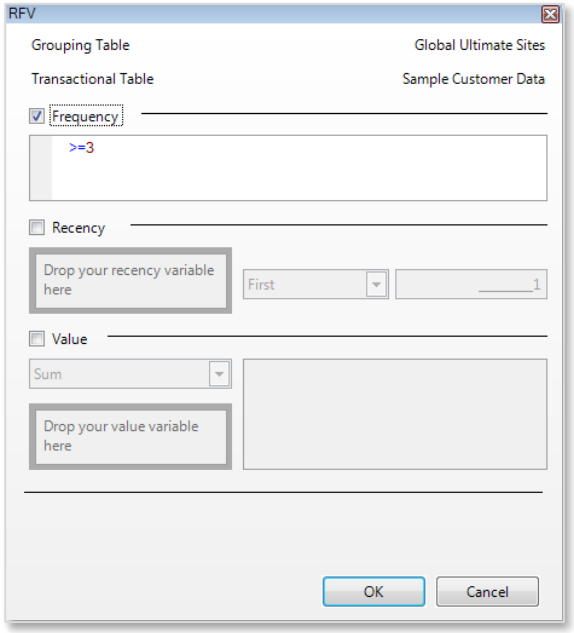
In this example we will start by finding the Global Ultimate Sites of our Sample Customers which have 3 or more individual Sites.

- Drag and drop the **Customer Flags** variable onto the workspace, select **Customer** and click the **Build** button
- Change the table level to **Global Ultimate Sites** – notice how the icon next to the summary line has changed to **ANY**
- Right click on the 2nd **ANY** row and select **Apply RFV to Global Ultimate Sites**
- Mark the check box next to **Frequency** and enter **>=3**
- Click **OK**. Then click **Build** to count the selection

The result is a count of the number of Global Ultimate Sites for our Sample Customers that have at least 3 individual Sites associated with them.

We can further refine our results by using the Recency and Value elements of this tool. By using date and numeric/currency variables we can select and test records against certain criteria.





- Right click on the **RFV** node within the selection
- Click on the **Modify RFV…** option from the pop up menu
- Check the box against **Recency** and then drag the date variable **Date of Last Transaction with Division 1** on to the activated drop zone
- Change the other boxes to show **Last 2**

This will identify the 2 Sites with the most recent last transaction with our Division 1 (where the Headquarters have 3 or more sites)

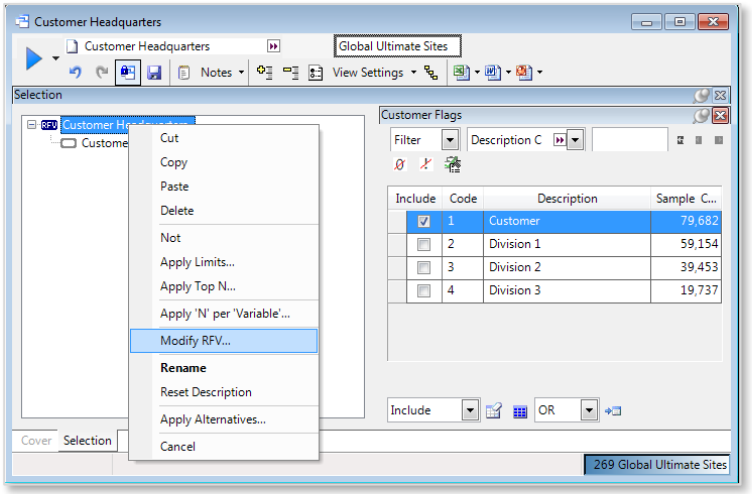
- Check the box against **Value** and then drag the numeric variable **Division 1 Spend** on to the activated drop zone
- Change the other boxes to show **Sum** & **>1000**

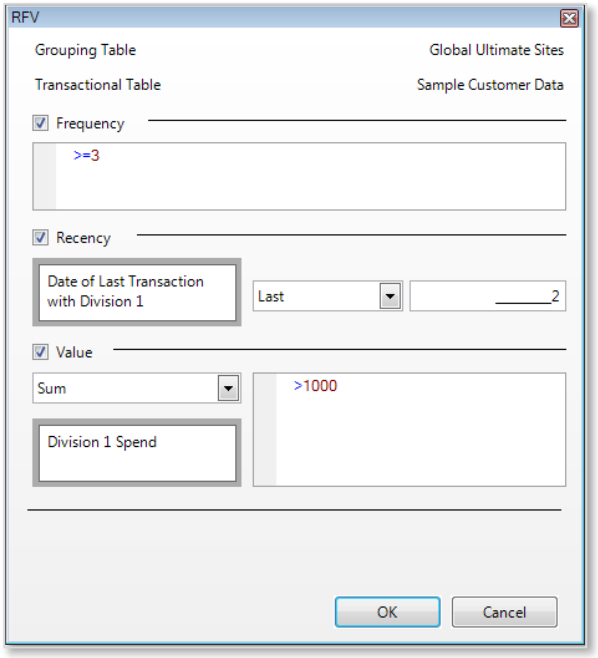
This will look at the 2 records identified in the Recency section, take the spend amount of those 2 transactions, sum them together and if the value is greater than 1000 that Global Ultimate Site will form part of the end count.

- Click **OK** and then click **Build** to count the selection
- Name the selection **Customer Headquarters**
- Keep this selection open or save it for the next section

N.B. The Recency element will only apply to your selection if it is used in conjunction with the Value element

The RFV function can also be used in conjunction with the Count Wizards. An explanation of these and the other Virtual Variable options are described in a later section.





Data Grids

Cell Aggregation

Data Grids have the ability to aggregate results from child tables. This feature has a number of marketing analysis applications including selecting latest transactional details, calculating total and average transaction values, selecting contacts at companies and exploring transactional trends.

By creating a Data Grid (on a GUS table selection) with a mixture of GUS and Sites table information the resolve table will automatically set itself to the lowest table level. This will then display all Sites for the GUS identified in the selection with GUS information repeated against each record.

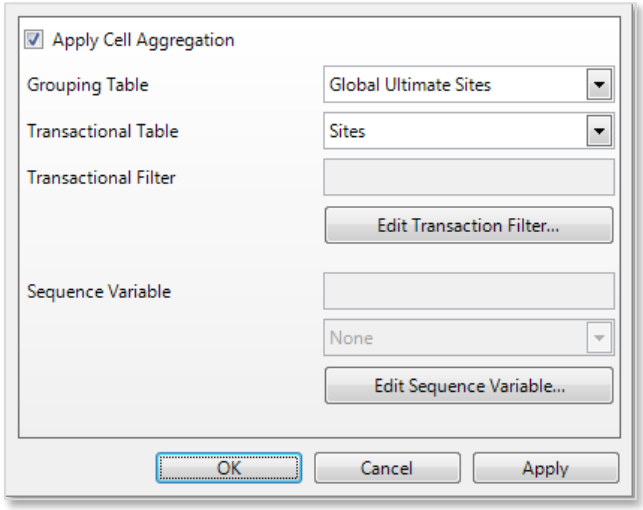
The Cell Aggregation within the Data Grid can be seen as the middle option between showing all or none of the transaction information. It allows you to sequence the transactional data to show just one set of transactional information against one higher level record e.g. sum of employees at all sites.

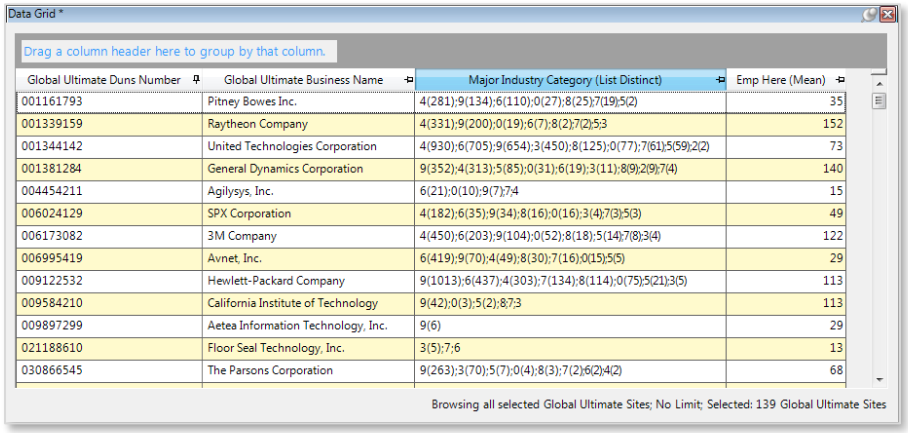
- Display the previous selection **Customer Headquarters**. Drop on a Data Grid and display the following variables – **Global Ultimate Business Name**, **Major Industry Category**, and **Emp Here**

- Change the table level back to **Global Ultimate Sites** and click on the Σ **Cell Aggregation** icon

- Tick the **Apply Cell Aggregation** box, leave the default settings and click **OK**. Click the **Build** button

The default aggregation option will be shown in the description heading for the two lower level variables.

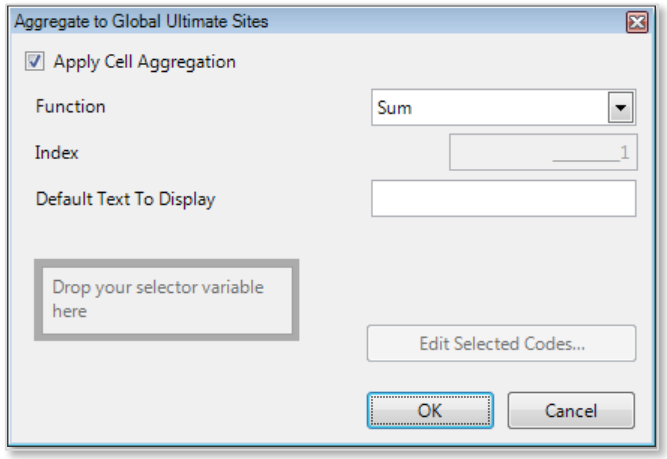




Refer to the Help file for a full list of the aggregation options available for the different types of variable displayed.

To display the sum total number of employees at all the sites associated with a Global Ultimate Site:

- Drag a second **Emp Here** variable on to the Data Grid
- Right click on its column heading and select **Aggregate to Global Ultimate Sites**
- Change the **Function** from **Mean** to **Sum**
- Click **OK** and then the **Build** button to see the results

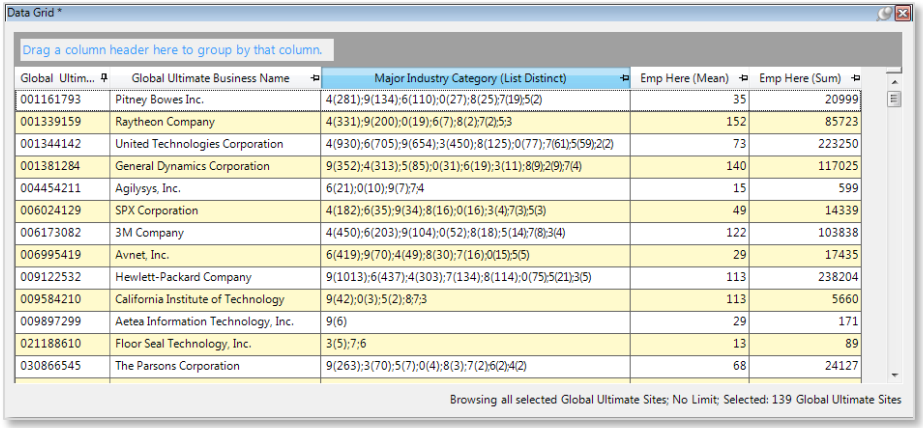


N.B. The main cell aggregation window gives a further option to filter the results and use a sequencing variable to establish an order.

Transactional Filter – create a selection that will restrict who will be displayed in the Data Grid.

Sequence Variable – this could be a purchase date that would then allow you to filter the records by last transaction or first transaction etc.

It is also possible to display side by side individual aggregation of data from sibling tables. See the How too guide – 2016 Q3 – Data Grids – Using Multiple Sibling Tables

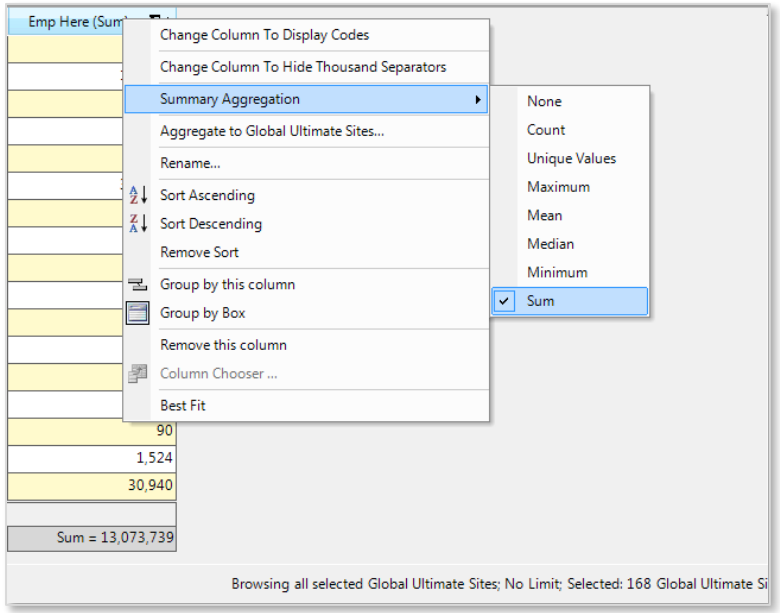


Summary Aggregation

Within a Data Grid it is possible to aggregate the information in a column based upon the variable type being displayed. For example a Currency/Numeric variable may be summarised in terms of the sum or mean of values.

- Right click on the heading **Emp Here (Sum)** and select **Summary Aggregation**
- Click on the **Sum** option

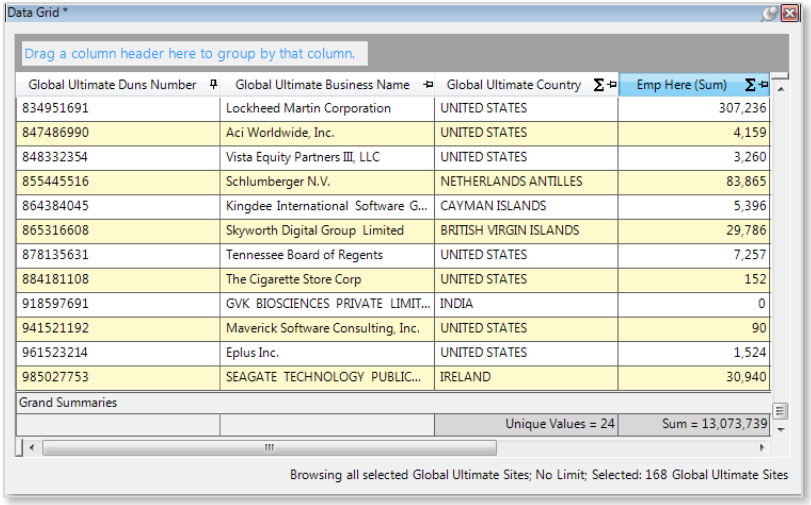
N.B. The result can now be seen at the bottom of the column and is calculated only on the information displayed in that column.



Another example this time using a selector variable with text:

- Drag the **Global Ultimate Country** variable onto the Data Grid and click **Build**
- Right click on the heading **Global Ultimate Country** and select **Summary Aggregation**
- Click on the **Unique Values** option

Once more the results can be seen at the bottom of the column. In this example it has given a count of the number of unique Countries found in this column of data.



Refer to the Help file for a full list of the Summary Aggregation options available for the different types of variable displayed.

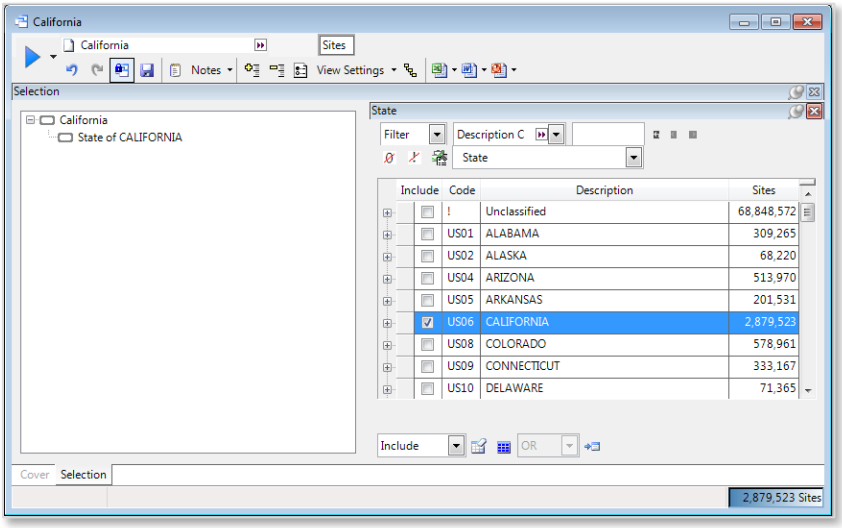
Cubes

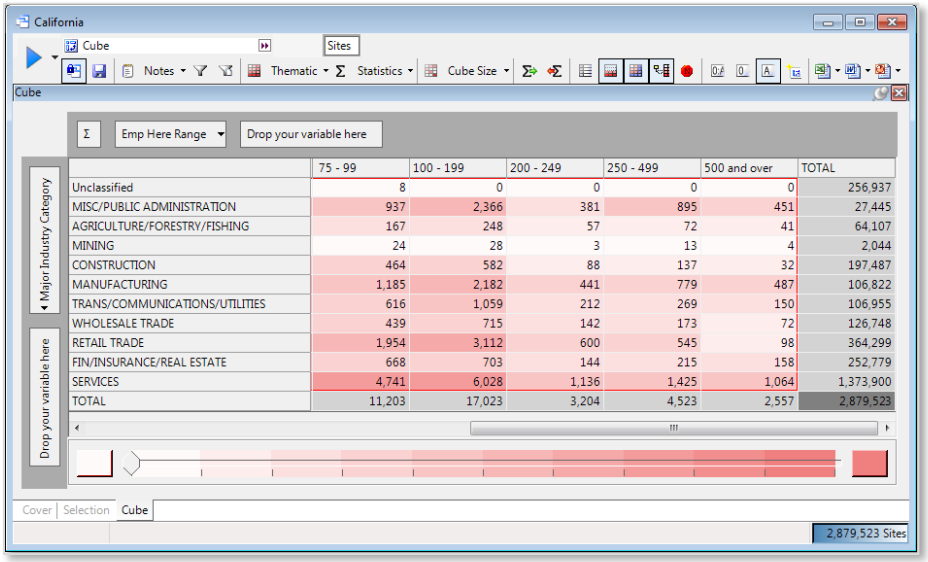
Cubes and Tables

You will recall from the base training that the cube tool enables you to analyze the selected rows by a number of variables. These variables are used as "Dimensions" of the cube. You may create a one, two, three or more dimensional cube.

Most work with cubes is done on the selection resolve table. However, a cube may be resolved to any table independently of the selection resolve table (try changing the resolve table on a cube page by right clicking or dragging on a table). A cube will also process data from multiple tables if either the dimensions or the measures (the statistics in the central body of the cube) are from multiple tables. However, in all cases the cube tool will only process data based upon the underlying selection.

- Create a selection on **California** from the System Explorer **Geography – US City** - **State** variable. Ensure the resolve table is set to **Sites**
- Drop on a **Cube** and display by **Emp Here Range** and **Major Industry Category**
- Click the **Build** button to create the cube





To see the results of this breakdown by another table drag the appropriate table from the System Tables window.

- Drag the **Contacts** table onto the middle of the **Cube** and press the **Build** button

To hide or delete a statistic once it has been added to a Cube:

- Click on the **Statistics** button and then uncheck the **Display** mark box against the item to hide or click **Delete …** to delete

To change the measure a cube is shaded by:

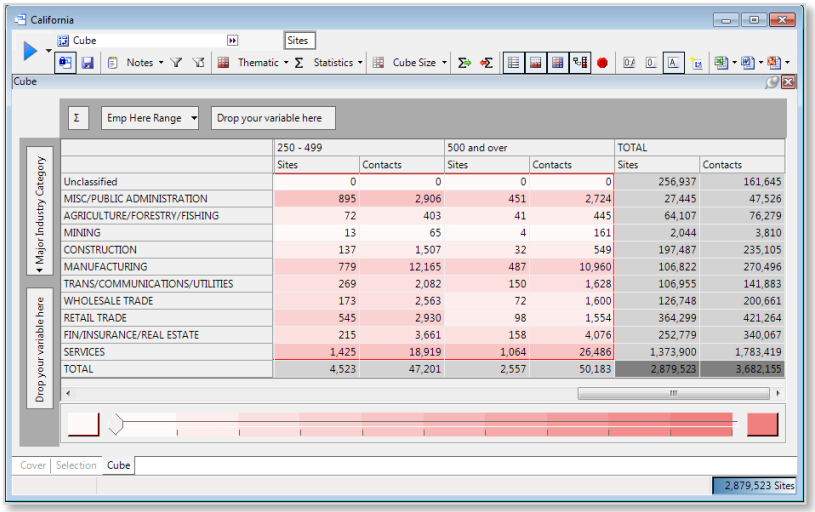
- Click on the **Statistics** button and in the **Primary?** Column put a tick alongside the measure you wish the cube to be shaded by

N.B. By moving the Σ button from the horizontal dimension to the vertical dimension figures shown in separate columns can be displayed in the same cell.

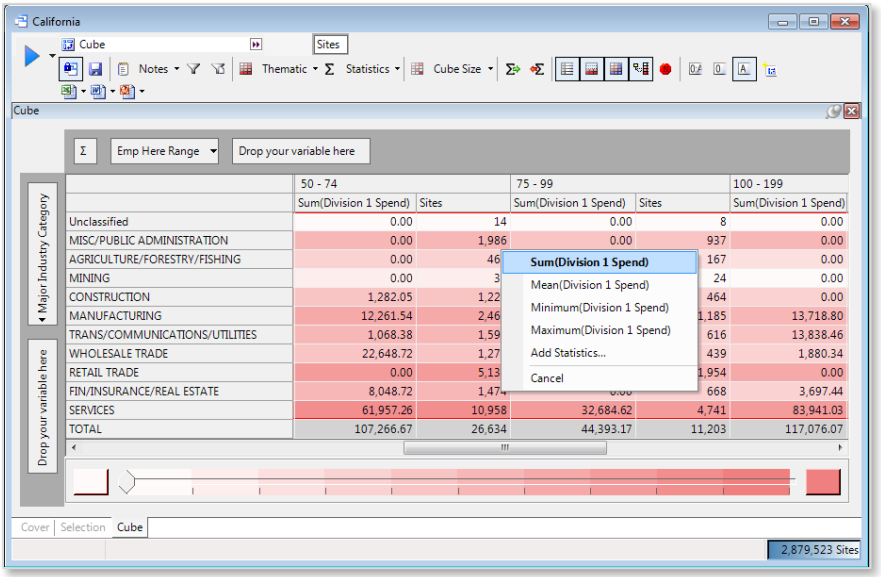
You can apply a calculated statistic within a Cube by right dragging a numeric or currency variable on to the center of the Cube display.

- Right drag and drop the variable **Division 1 Spend** on to the Cube
- From the menu which is displayed select the calculation you would like to perform

N.B. In the screen shot opposite you can see the sum of spend with our Division 1 broken down by MIC and EHR. Note that the numbers of sites are all those in California and not just the ones we have transacted with.



	250 - 499		500 and over		TOTAL	
	Sites	Contacts	Sites	Contacts	Sites	Contacts
Unclassified	0	0	0	0	256,937	161,645
MISC/PUBLIC ADMINISTRATION	895	2,906	451	2,724	27,445	47,526
AGRICULTURE/FORESTRY/FISHING	72	403	41	445	64,107	76,279
MINING	13	65	4	161	2,044	3,810
CONSTRUCTION	137	1,507	32	549	197,487	235,105
MANUFACTURING	779	12,165	487	10,960	106,822	270,496
TRANS/COMMUNICATIONS/UTILITIES	269	2,082	150	1,628	106,955	141,883
WHOLESALE TRADE	173	2,563	72	1,600	126,748	200,661
RETAIL TRADE	545	2,930	98	1,554	364,299	421,264
FIN/INSURANCE/REAL ESTATE	215	3,661	158	4,076	252,779	340,067
SERVICES	1,425	18,919	1,064	26,486	1,373,900	1,783,419
TOTAL	4,523	47,201	2,557	50,183	2,879,523	3,682,155



	50 - 74		75 - 99		100 - 199
	Sum(Division 1 Spend)	Sites	Sum(Division 1 Spend)	Sites	Sum(Division 1 Spend)
Unclassified	0.00	14	0.00	8	0.00
MISC/PUBLIC ADMINISTRATION	0.00	1,986	0.00	937	0.00
AGRICULTURE/FORESTRY/FISHING	0.00	46		167	0.00
MINING	0.00	3		24	0.00
CONSTRUCTION	1,282.05	1,22		464	0.00
MANUFACTURING	12,261.54	2,46		1,185	13,718.80
TRANS/COMMUNICATIONS/UTILITIES	1,068.38	1,59		616	13,838.46
WHOLESALE TRADE	22,648.72	1,27		439	1,880.34
RETAIL TRADE	0.00	5,13		1,954	0.00
FIN/INSURANCE/REAL ESTATE	8,048.72	1,47		668	3,697.44
SERVICES	61,957.26	10,958	32,684.62	4,741	83,941.03
TOTAL	107,266.67	26,634	44,393.17	11,203	117,076.07

Cube Table Filter

It is possible to apply a Table filter to your Cube that allows you to restrict the records in the breakdown.

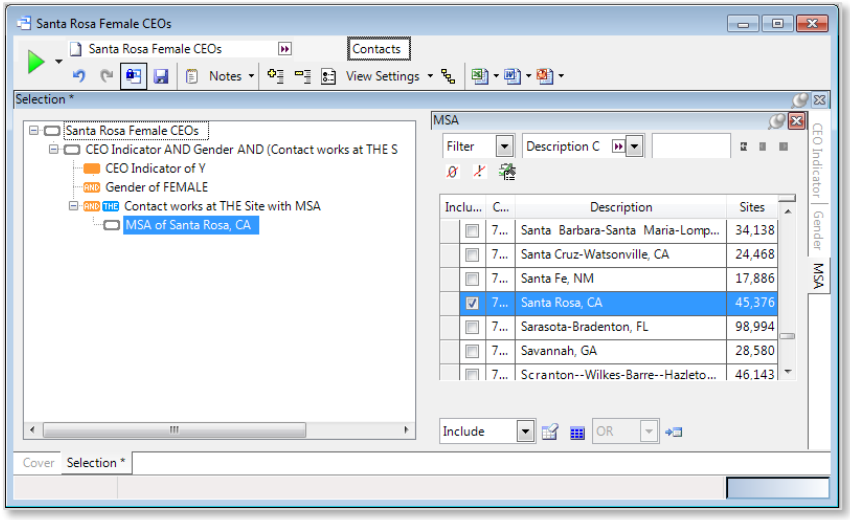
- Recreate the selection and cube display as in the previous example
- Change the Table level to **Contacts**
- Drag the **Site** table onto the middle of the **Cube**
- Click the **Build** button

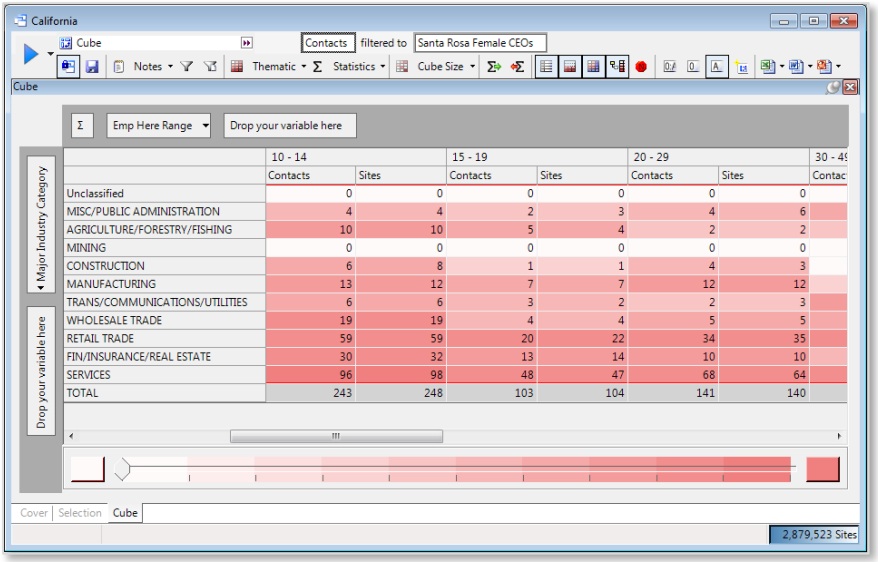
N.B. Note that alongside the Table box at the top of the window a new drop zone has appeared with the words Return all Contacts.

- Create a new selection of **Contacts** where **CEO Indicator - Y** & **Gender – Female** & **MSA – Santa Rosa**
- Rename the selection **Santa Rosa Female CEOs**
- Drag the **Santa Rosa Female CEOs** selection onto the **Return all Contacts** drop zone
- Click **Build**

The display will now be restricted to show only the number of Contacts who are Female CEO's in Santa Rosa.

N.B. Remember you can additionally filter which categories are displayed in each dimension by using the pop-up menu on the variable. You can turn this filter on or off using the filter toggle button .



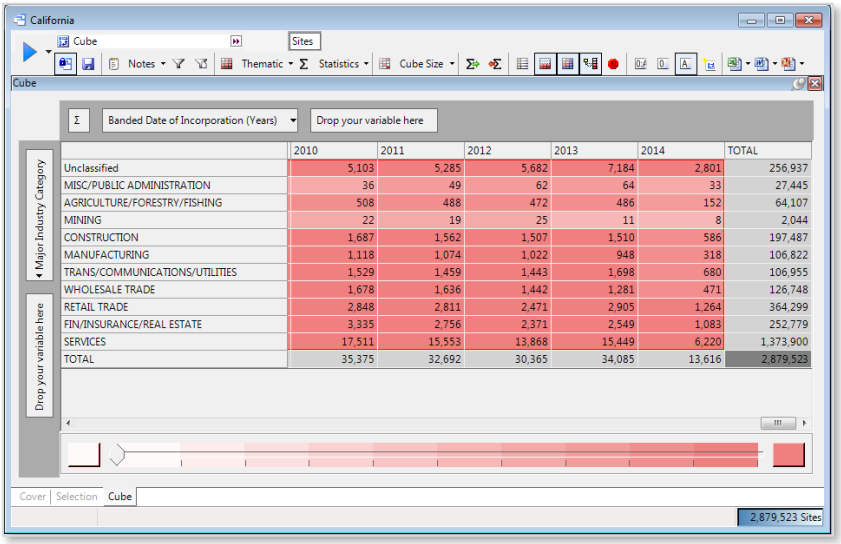


Date and Numeric Variables

The use of Date and Numeric variables on a Cube can give you some extra display options.

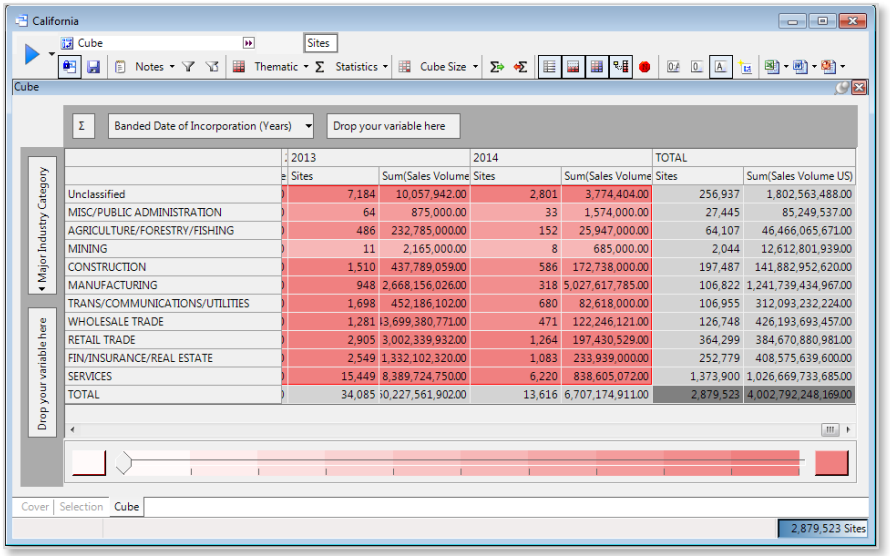
- Redisplay the **Cube** with just the **Major Industry Category** variable on the vertical dimension
- Right drag the **Date of Incorporation** variable onto the drop box on the horizontal dimension and select **Years**
- Click the **Build** button

Here the date variable gives you the ability to display the figures in a number of date divisions including the Full Date which is the default.



	2010	2011	2012	2013	2014	TOTAL
Unclassified	5,103	5,285	5,682	7,184	2,801	256,937
MISC/PUBLIC ADMINISTRATION	36	49	62	64	33	27,445
AGRICULTURE/FORESTRY/FISHING	508	488	472	486	152	64,107
MINING	22	19	25	11	8	2,044
CONSTRUCTION	1,687	1,562	1,507	1,510	586	197,487
MANUFACTURING	1,118	1,074	1,022	948	318	106,822
TRANS/COMMUNICATIONS/UTILITIES	1,529	1,459	1,443	1,698	680	106,955
WHOLESALE TRADE	1,678	1,636	1,442	1,281	471	126,748
RETAIL TRADE	2,848	2,811	2,471	2,905	1,264	364,299
FIN/INSURANCE/REAL ESTATE	3,335	2,756	2,371	2,549	1,083	252,779
SERVICES	17,511	15,553	13,868	15,449	6,220	1,373,900
TOTAL	35,375	32,692	30,365	34,085	13,616	2,879,523

- Right drag the **Sales Volume US** variable onto the **Cube**
- From the popup menu select **Sum(Sales Volume US)** and click the **Build** button



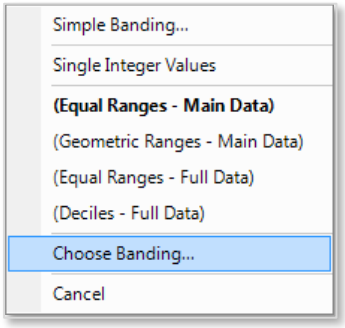
	2013		2014		TOTAL	
	Sites	Sum(Sales Volume	Sites	Sum(Sales Volume	Sites	Sum(Sales Volume US)
Unclassified	7,184	10,057,942.00	2,801	3,774,404.00	256,937	1,802,563,488.00
MISC/PUBLIC ADMINISTRATION	64	875,000.00	33	1,574,000.00	27,445	85,249,537.00
AGRICULTURE/FORESTRY/FISHING	486	232,785,000.00	152	25,947,000.00	64,107	46,466,065,671.00
MINING	11	2,165,000.00	8	685,000.00	2,044	12,612,801,939.00
CONSTRUCTION	1,510	437,789,059.00	586	172,738,000.00	197,487	141,882,952,620.00
MANUFACTURING	948	2,668,156,026.00	318	5,027,617,785.00	106,822	1,241,739,434,967.00
TRANS/COMMUNICATIONS/UTILITIES	1,698	452,186,102.00	680	82,618,000.00	106,955	312,093,232,224.00
WHOLESALE TRADE	1,281	3,699,380,771.00	471	122,246,121.00	126,748	426,193,693,457.00
RETAIL TRADE	2,905	3,002,339,932.00	1,264	197,430,529.00	364,299	384,670,880,981.00
FIN/INSURANCE/REAL ESTATE	2,549	1,332,102,320.00	1,083	233,939,000.00	252,779	408,575,639,600.00
SERVICES	15,449	8,389,724,750.00	6,220	838,605,072.00	1,373,900	1,026,669,733,685.00
TOTAL	34,085	10,227,561,902.00	13,616	6,707,174,911.00	2,879,523	4,002,792,248,169.00

To use a numeric variable as a cube dimension:

- Right drag the **Employee Here** variable onto the cube dimension drop panel

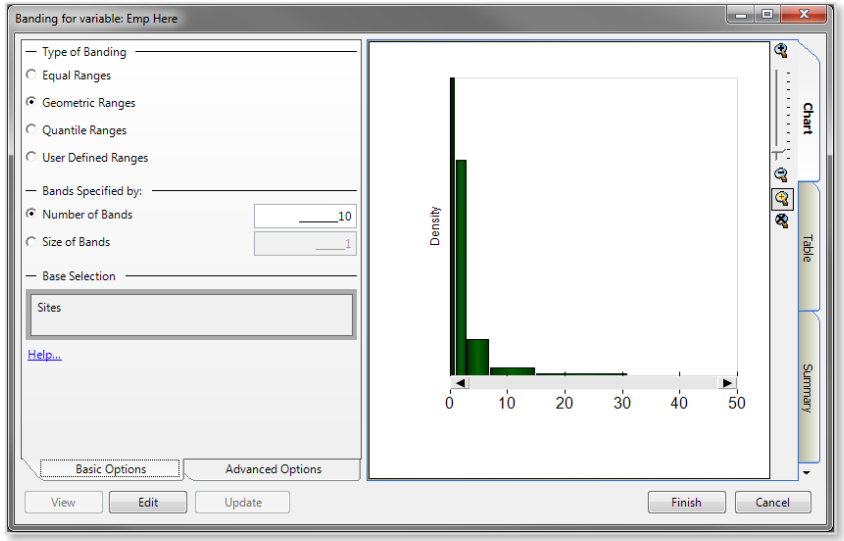
- This allows you to choose from various standard bandings of the data in the variable or use the banding tool to create any other banding

N.B. The brackets in the pop-up menu show which banding options need calculating. Once a banding option is calculated it is shown without brackets



- The "Choose Banding…" option allows use of the numeric distribution tool to analyze and generate a wide range of different banding types including quantiles (quartiles, deciles, centiles, etc.) and specific treatment of outlying values

- Once a banding is defined, you can amend it by right clicking on the dimension and choosing **Edit**. This will display the banding tool



RFV & Cube Dimensions

The functionality of RFV can be used to set the dimensions on a Cube.

For example Frequency can be added on a dimension by right dragging a transaction table onto a Cube dimension.

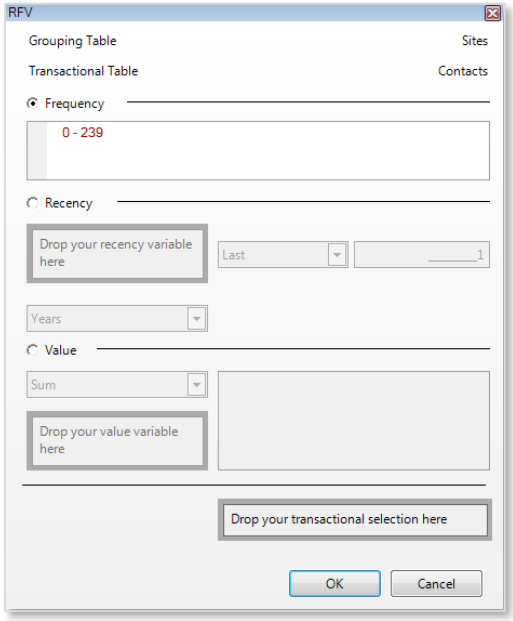
- Open a blank Cube set to **Sites**
- Right drag the Contacts table onto the vertical cube dimension and select **Add RFV Frequency dimension**
- Amend the bandings as required and click **OK**

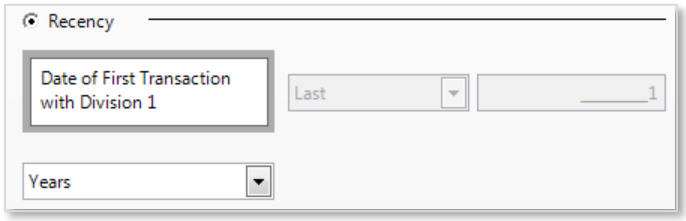
To use the Recency function, right drag a date variable onto a Cube dimension.

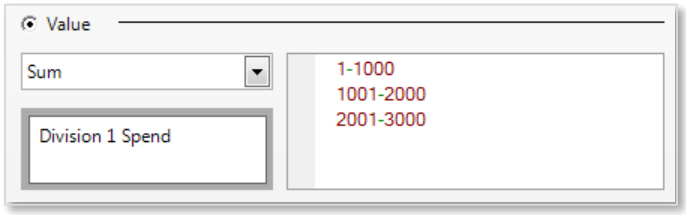
- Open a blank Cube set to **Sites**
- Right drag the **Date of First Transaction with Division 1** variable onto the vertical cube dimension and select **Add RFV Recency dimension**
- Select the date period to band as required and click **OK**

To use the Value function, right drag a numeric or currency variable onto a Cube dimension.

- Open a blank Cube set to **Sites**
- Right drag the **Division 1 Spend** variable onto the vertical cube dimension and select **Add RFV Value dimension**
- Amend the bandings as required and click **OK**







Queries as a Cube Dimension

In addition to using variables as cube dimensions, you can use one or more queries as a dimension. This is appropriate if you wish to analyze by a characteristic that is not directly presented as a variable (for example, analyze by “Bought in the last year”).

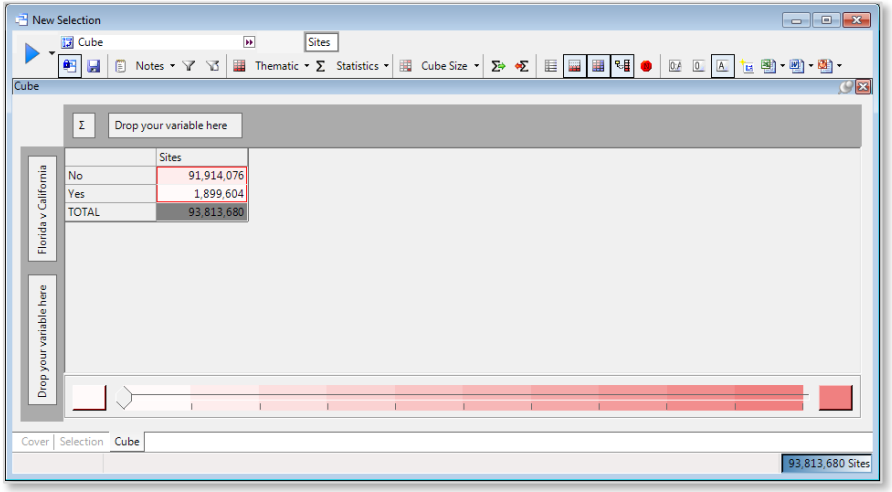
- Define a query in a new selection window
- Drag and drop it onto the cube dimension drop zone

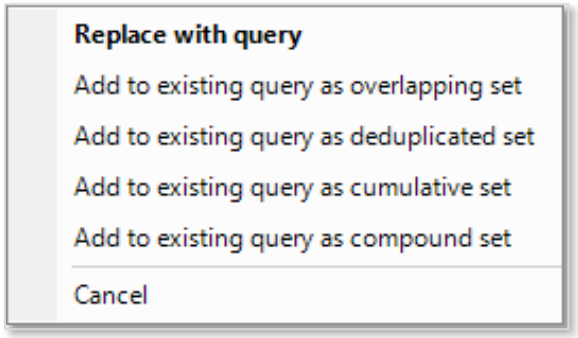
The query will produce two categories, Yes and No and behaves exactly as a variable from the table it is resolved to.

N.B. You can drag the query back off the cube dimension, but if you change it, you will need to drag the amended version back on again. The cube holds a copy of your query; it does not reference any changes you may make in the original definition window.

A single dimension can also have multiple queries added to it. Because in general two selections can both select the same records, you can choose whether you want to see the selections overlap each other (providing results that would look similar to a flag array selector variable) or whether the selections should be "de-duplicated". De-duplicating selections means that if a record is selected by one selection, then it will not be counted in the figures for all subsequent selections.

- Define two additional queries
- Right-drag one of these and drop it on the first query within the cube dimension area. You will see a popup menu as shown opposite





The options for multiple selections on a dimension are:

Overlapped
Where a cell shows every record that meets the criteria for that selection.

Deduplicated
Where a cell shows the count of that selection with any duplicated records from the previous cell(s) having been removed.

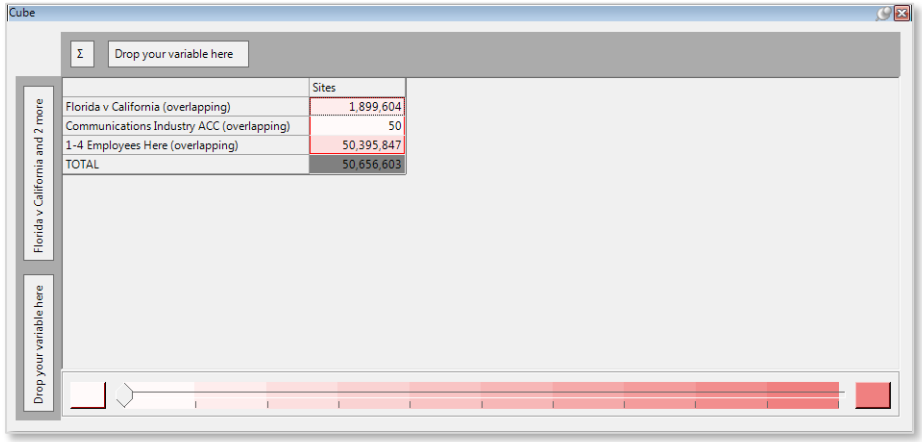
Cumulative
Where a cell adds any new records with those from the previous cell(s).

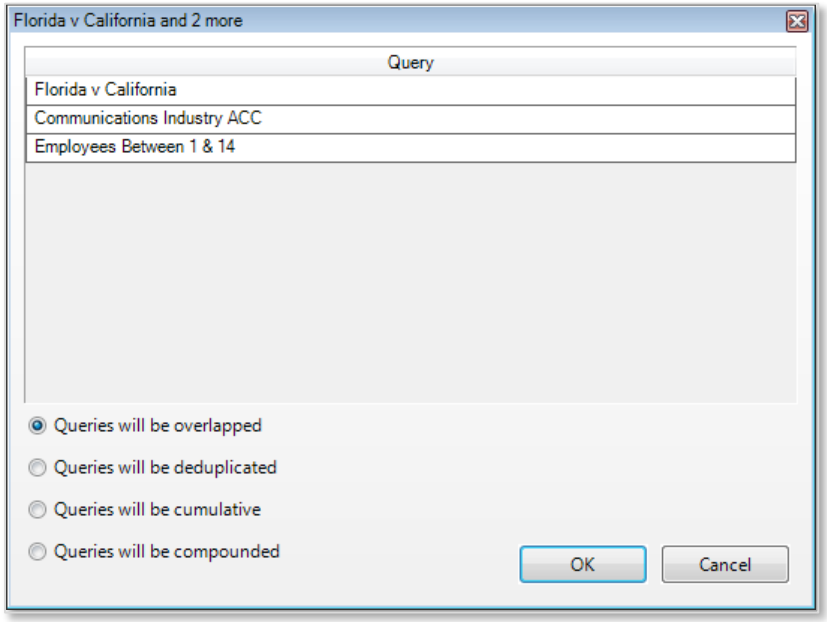
Compounded
Where a cell only shows records where the record meets the condition from the previous cell(s).

- Drag your next selection on to the same dimension drop box and click on the **Build** button

Once you have created your multi selection display, the order of the selections can be altered by using the Edit option.

- Right click on the dimension contains the dragged on selections and choose **Edit**
- Right click on the queries to move them up or down to change the order in which they are applied
- Click **OK**, and then click on the **Build** button





Cube Calculated Measures

This feature allows you to perform simple calculations in and between cells on a Cube or Tree tool.

For example you may wish to display the average Contacts at a Site broken down by Major Industry Category.

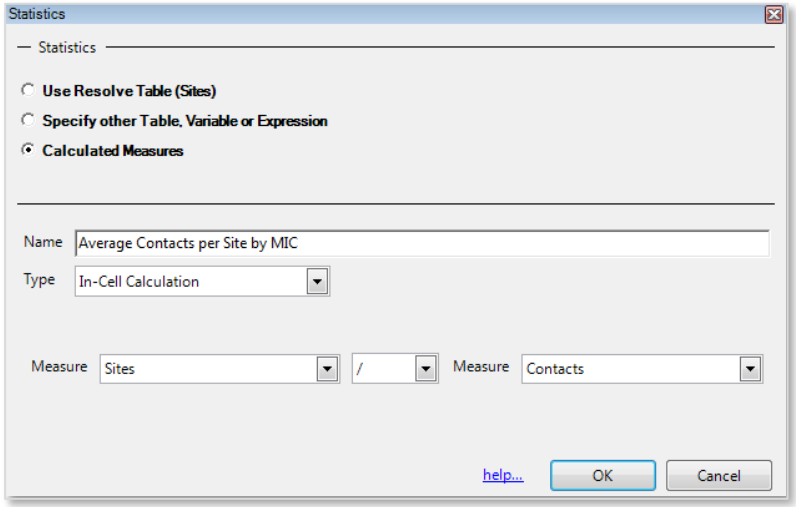
- Display a Cube on a blank selection at the Site table level, with **Major Industry Category** on the vertical dimension
- Drag the **Contacts** table onto the center of the Cube
- Click on the **Statistics** button followed by **Add Statistics…**
- Click on the radio button called **Calculated Measures** and complete as opposite

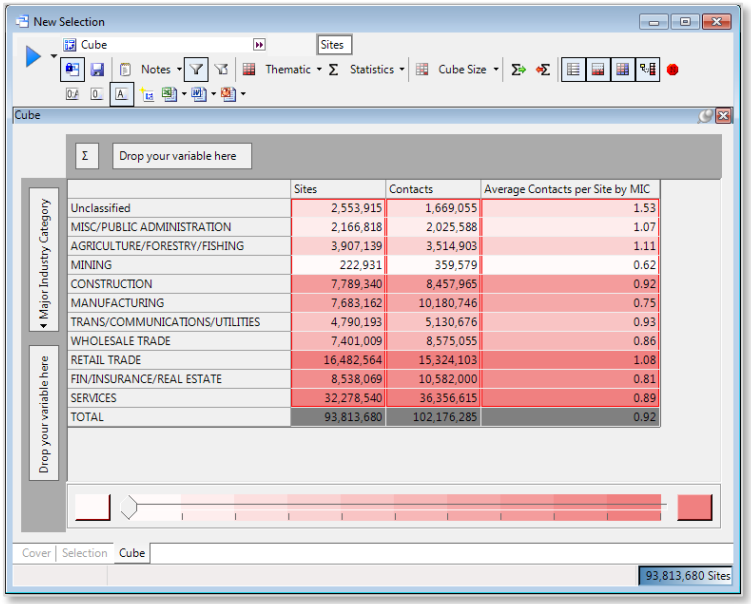
  Name: Average Contacts per Site by MIC

  Type: In-Cell Calculation

  Measure: Sites / Contacts
- Click **OK**

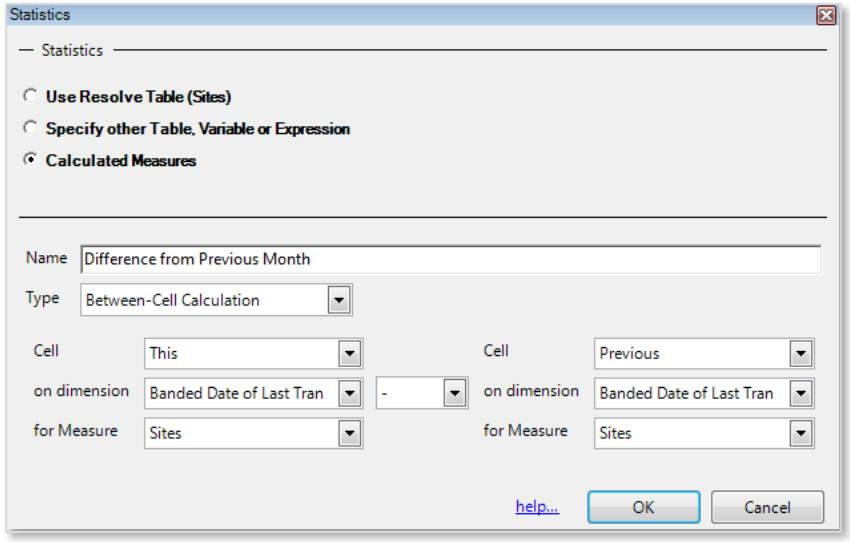
The results of the calculation are displayed immediately on the Cube without the need to click the build button.



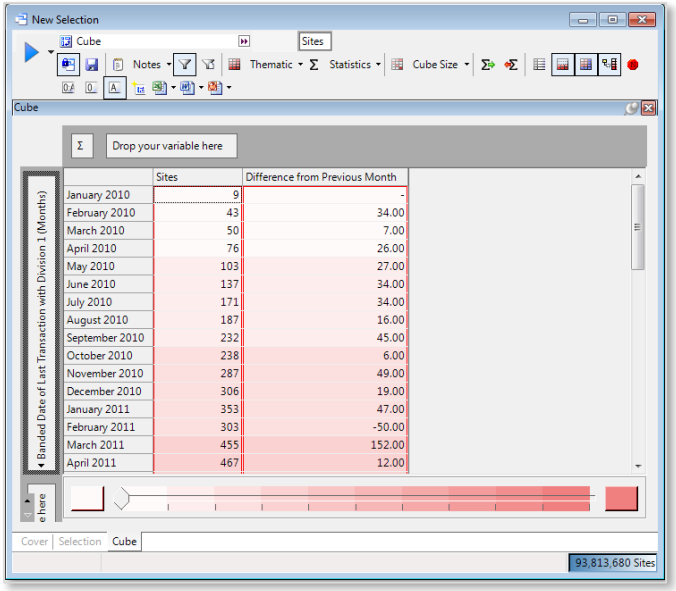


Another example of a Calculated Measure would be to display the difference between 2 cells, perhaps to show the difference in the number of Sites purchased one month to the previous month.

- Display a Cube on a blank selection at the **Site** table level
- Right drag the **Date of Last Transaction with Division 1** variable onto the vertical dimension and select **Months**
- Click on the **Statistics** button followed by **Add Statistics…**
- Click on the radio button called **Calculated Measures** and complete as opposite
- Click **OK**



The results of the calculation are displayed immediately on the Cube without the need to click the build button.



N.B. The Cell contains a list of actions which will be restricted if you use a non-sequential variable:

This	Use the cells in the currently selected measure
Total	Use the total cell of the dimension
First	Use the first cell of the dimension
Previous	Use the previous cell to the one already indicated
Next	Use the next cell to the one already indicated
Last	Use the last cell of the dimension

Virtual Variables & Wizards

Virtual Variables are a way of adding to the information that you can analyze within D&B Market Insight. An initial set of variables are created when the system is built and these cannot be changed without rebuilding the system (which can take a long time and will require your administrator to perform the task). Virtual Variables can be used to import additional information into your system or to derive new data by summarizing or aggregating existing information into new forms.

Virtual Variables are treated in the same way as "normal" variables by the system and can be used in all the same ways. They will appear in the System Explorer and can be dragged onto all the same places as a normal variable.

Unlike normal variables, Virtual Variables can also be updated once they have been created. Virtual Variables can also be deleted if they are no longer needed (whereas normal variables can only be removed by rebuilding the system).

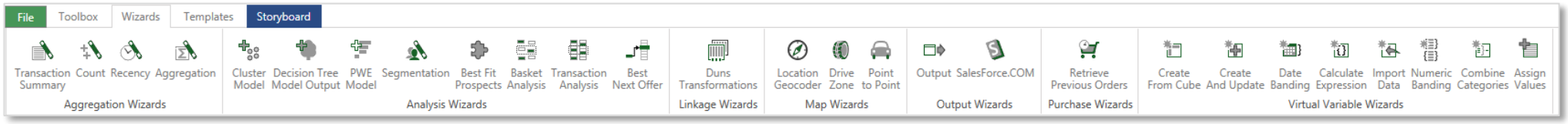
The creation of a Virtual Variable is managed through a step by step Wizard process. A full description of each Virtual Variable Wizard is described in **Appendix 1**. A few examples are shown in this manual.

However, Virtual Variables are automatically invalidated when the system is refreshed (for example when fresh D&B data is loaded). The variables will remain in the system explorer, but will be grayed out until they are refreshed (see below).

N.B. Creating virtual variables uses disk space on the D&B Market Insight servers. This space usage is monitored automatically for each user. Large virtual variables will be archived or referred back to users to delete to ensure disk space usage remains within sensible quotas.

N.B. A fully worked example for each Wizard can be found on the software Help menu. The Wizards displayed on your system may differ to those shown here and are dependent upon individual configuration.

N.B. The EYWTKA Wizards and Virtual Variables manual provides details regarding the use and function of each Wizard.



Manage Virtual Variables

The Manage Virtual Variable dialog allows the user to recreate Virtual Variables that are no longer available due to the Market Insight system being rebuilt. The user can also permanently remove Virtual Variables (or old virtual variable definitions) that are no longer needed.

Most Virtual Variables will be created not by the administrator in FastStats Designer but by end users in Market Insight. When the Market Insight system is rebuilt using Designer it won't necessarily include the definitions for recreating the Virtual Variables and they will become unavailable to the user. These variables can then be recreated using the Manage Virtual Variables dialog available from the Tools menu.

Refreshing a Virtual Variable "By Rule"

Whenever an action is performed on a Virtual Variable, such as creating it or modifying it the definition of the change is saved. These constitute a set of "rules" that are used to put the variable in its current state. When refreshing the Virtual Variable these rules can be rerun to recreate the Virtual Variable. However, most of the rules used to recreate a variable will use selections on other variables. When the Market Insight system is reloaded the data in the system will change and the selections may now select different records. For example, one code of a selector Virtual Variable can be set to a selection of people that bought a certain product. When the Market Insight system is refreshed more purchases will have been added into the system and so more people will have bought the product in question. Therefore if the Virtual Variable is recreated "By Rule" the count for the particular code will go up as the selection now has more people in it.

Refreshing a Virtual Variable "From URN Snapshot"

If you create a Virtual Variable and want it to remain exactly the same after a rebuild of the Market Insight system then you can optionally create a "URN Snapshot" of the variable. This takes the variable and records the value for each record against the Unique Reference Number (URN) for the table that the variable is on. Then when the variable is recreated from this snapshot it has exactly the same values as before.

This could be useful if the same PWE Model is created on a set of data after each rebuild of the Market Insight system and you want to be able to compare how the model has changed between builds. When you create the PWE variable you are given the option to take a URN Snapshot of the variable. This can then be used to recreate the variable after the system rebuild.

A URN Snapshot can also be taken for any virtual variable at any time by right clicking on the variable in the System Explorer and selecting the "Create URN Snapshot" option.

Permanently Deleting Variable

Variables can also be permanently deleted using the Delete button. If the variable is not currently available in the system (indicated by a faded icon to the left of the description) then the information required to refresh the variable will be removed from the server. This will mean that the variable will no longer appear in the list of variables to be refreshed and it will never be able to be recreated.

If the variable is available in the system (indicated by a color variable icon), then the variable will be deleted from the system and also permanently removed from the server.

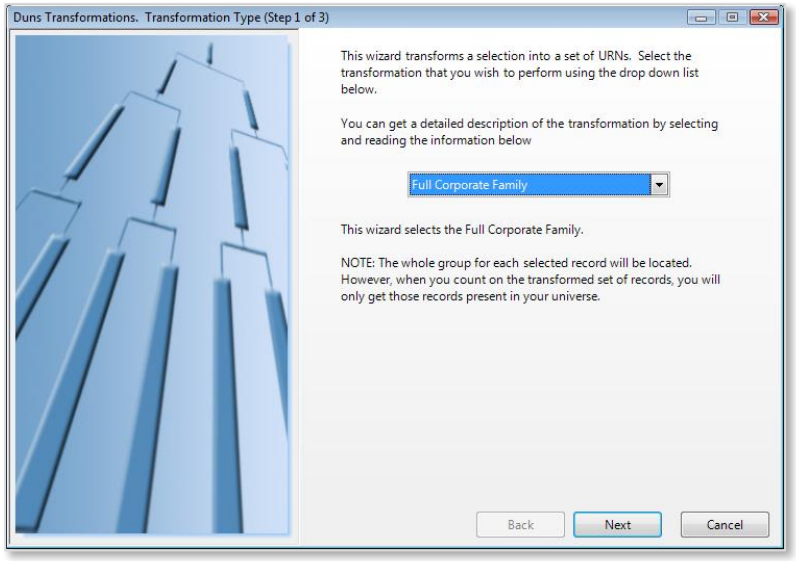
Wizard Example – DUNs Transformations

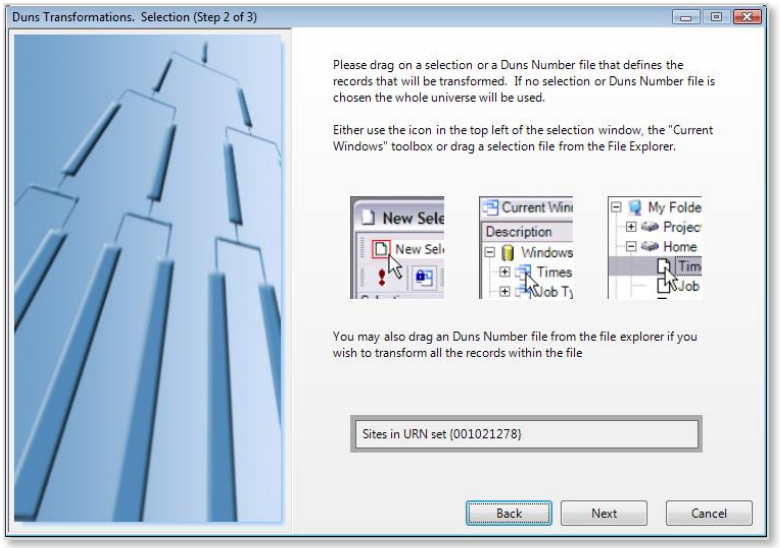
One of the wizards available to you is the Duns Transformation wizard which allows you to transform a selection of company sites into a list of related sites according to the corporate structure. The list is shown as a set of DUNS numbers. You can choose whether the immediate parent companies, ultimate parent companies, subsidiaries, or every company within the whole group, for example, are included in the list.

- Click on the **Duns Transformations** link in the Wizards window
- Step 1 - From the drop down menu select the grouping you want to make based on the Duns numbers determined in the next step. Click **Next**
- Step 2 - Drag onto the drop zone a selection or DUNs file to identify the sites used to obtain the relationship set in step 1. Click **Next**
- Step 3 - Click on the **Browse...** button to determine where you want to save your file
- Enter the name you want to save the file as
- Click **Finish**

The example opposite will find the DUNs of the Full Corporate Family associated with the DUNs you identified in Step 2 of the wizard.

You can use this selection for further analysis in Market Insight or drop it straight onto the Data Purchase wizard (see the Help files for further details).



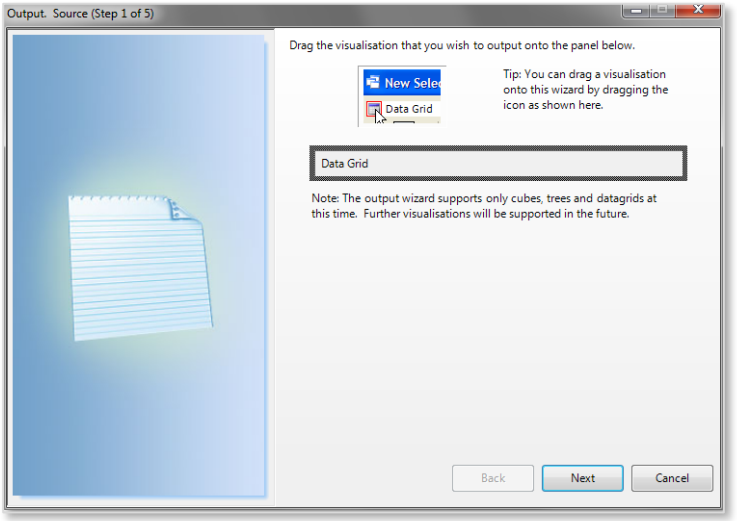


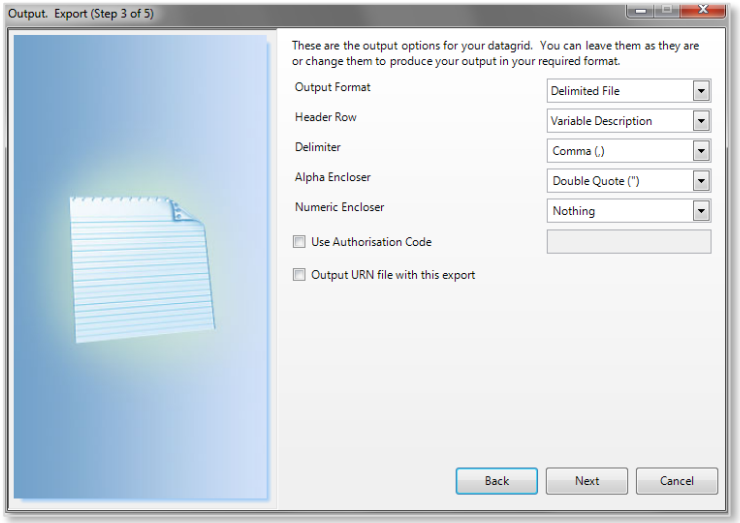
Wizard Example – Output

This wizard allows you to quickly and simply output the records from a Cube, Tree or Data Grid. You may use this method as an alternative export process.

- Click on the **Output** wizard link
- **Step 1** – Drag the **Data Grid** onto the drop zone box. (You will go to Step 3) Click **Next**
- **Step 2** – This step relates to a Cube/Tree tool if selected at Step 1. The options on this step restrict the cells selected by size. Click **Next**
- **Step 3** – Select the output options for the records in the Data Grid. Click **Next**
- **Step 4** – Enter the name for the file. Use the **Browse…** button to select the location for the file. Click **Next**
- **Step 5** – This step will state the number of records that have been output. Click **Finish**

N.B. Both the Cube and Tree tool will follow the above steps and will be output as a text file.





Wizard Example – Combine Categories

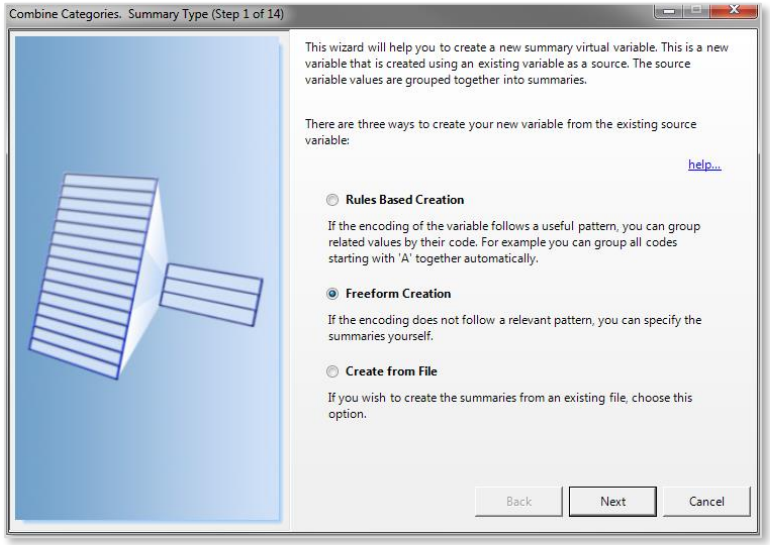
The Combine Categories Wizard allows you to create a virtual variable based upon an existing variable where the values are grouped together into summaries.

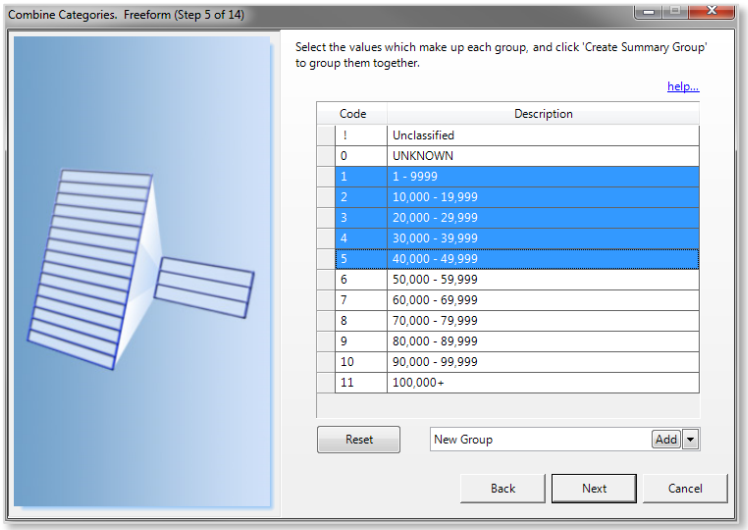
Example

The existing variable Square Footage Ranges breaks down this information into over 10 bands. You may wish to make this simpler with just 2 categories of Below 50,000 sq ft and Above 50,000 sq ft.

- Click on the **Combine Categories** wizard link
- **Step 1** – Choose the appropriate method required for your source data. Select the **Freeform Creation** radio button. The other two options are described in the Help. Click **Next**
- **Step 2** – Drag on the **Square Footage Range** variable. Click **Next**. (You go to step 5)
- **Step 3** – If you chose **Rules Based Creation** in step 1 you now have the opportunity to define how the codes are grouped. Click **Next**
- **Step 4** – If you ticked **Load Descriptions From File** in step 3 you can drag the file on at this step. Click **Next**
- **Step 5** – Highlight the first group of categories you wish to summarise. Click **Add**. (See screen shot opposite)

  Select the other categories you require (all values must belong to a group) and click **Add**. Click **Next** (You go to Step 8)





- **Step 6** – If you chose **Create From File** in step 1 you can drag your file on here. Click **Next**

- **Step 7** – If a **Fixed Format** file is used you can set the column widths here. Click **Next**

- **Step 8** – Enter **Below 50,000 sq ft** for **Description of group 1** and **Above 50,000 sq ft** for the **Description of group 2** etc. Click **Next**

- **Step 9** - You may enter optional notes in this window. Click **Next**

- **Step 10** - Enter the **Description – Square Footage Division**. Click **Next**

  If you have an existing Virtual Variable which will be superseded by this one it can be dragged onto the box **Drop the variable to overwrite here** to overwrite it.

  Tick the URN Snapshot if you wish to recreate this Virtual Variable after a refresh of the data with the exact same records.

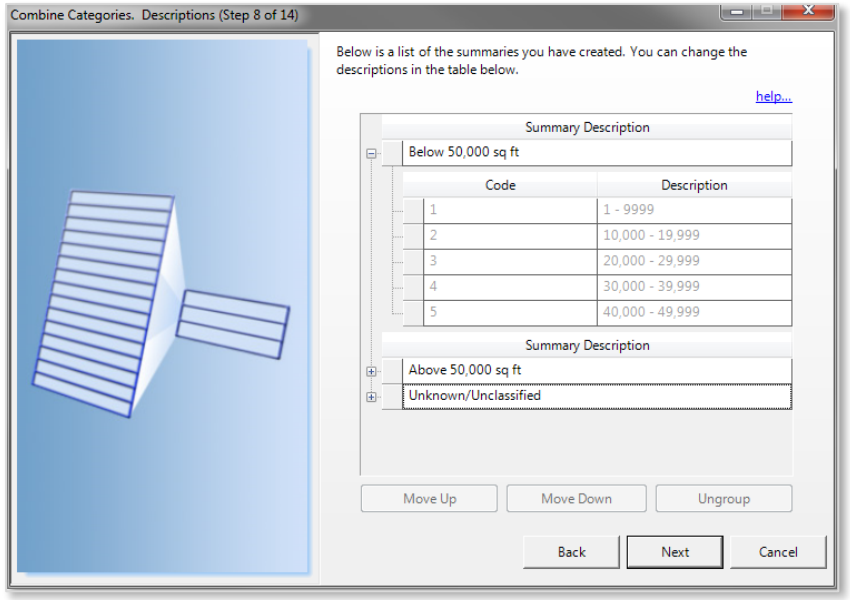
  To **Modify Security Attributes** tick this box and you will go to Step 13. Click **Next**

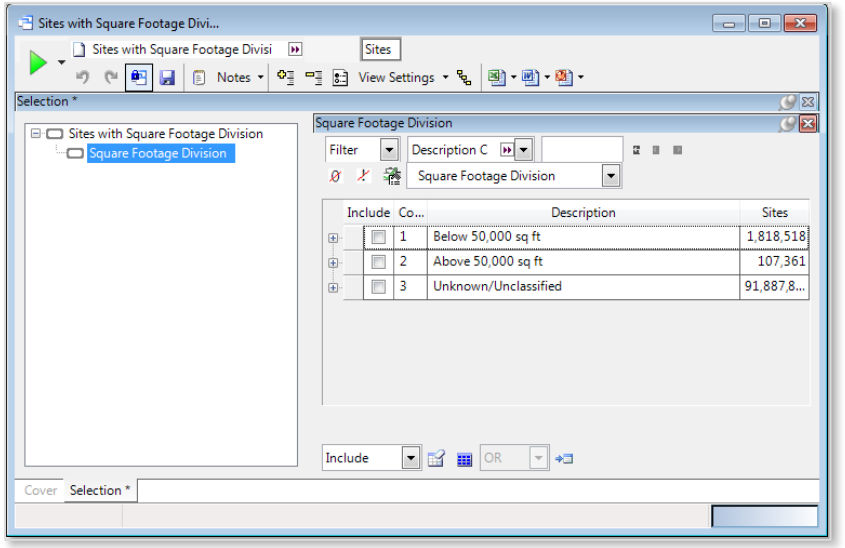
- **Step 11** – This step is only visible if you are running an Enterprise system and you have ticked the **Modify Security Attributes** box in the previous step.

- **Step 12** – This step will tell allow you to see if any variables are dependent upon a variable that is being updated

- **Step 13** – Tick the **Show new variable as a selection** box. Click **Finish**

The virtual variable now displays as a 2 category selector.





Expressions

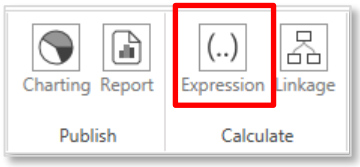
D&B Market Insight supports expressions as a method to calculate numeric results. Expressions can use constants, mathematical, logical and date functions, Market Insight variables and Market Insight Queries as elements of an expression. Use of expressions significantly enhances the power of Market Insight.

Expressions are currently used in three places in D&B Market Insight:

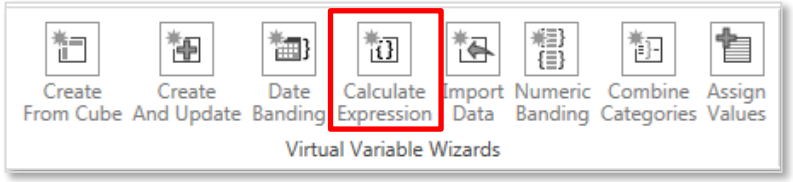
- In the 'Calculate Expression' wizard to populate a new variable according to a mathematical expression or logical rule
- Dragged from the Expression tool onto a Cube as a cube statistic
- Dragged from the Expression tool onto a Data Grid as an output column

Expressions may be saved and edited independently and are automatically saved within the tools they are used on.

N.B. Please note that Virtual Variables can be created directly from some expressions using the icon.



Toolbox Ribbon



Wizard Ribbon

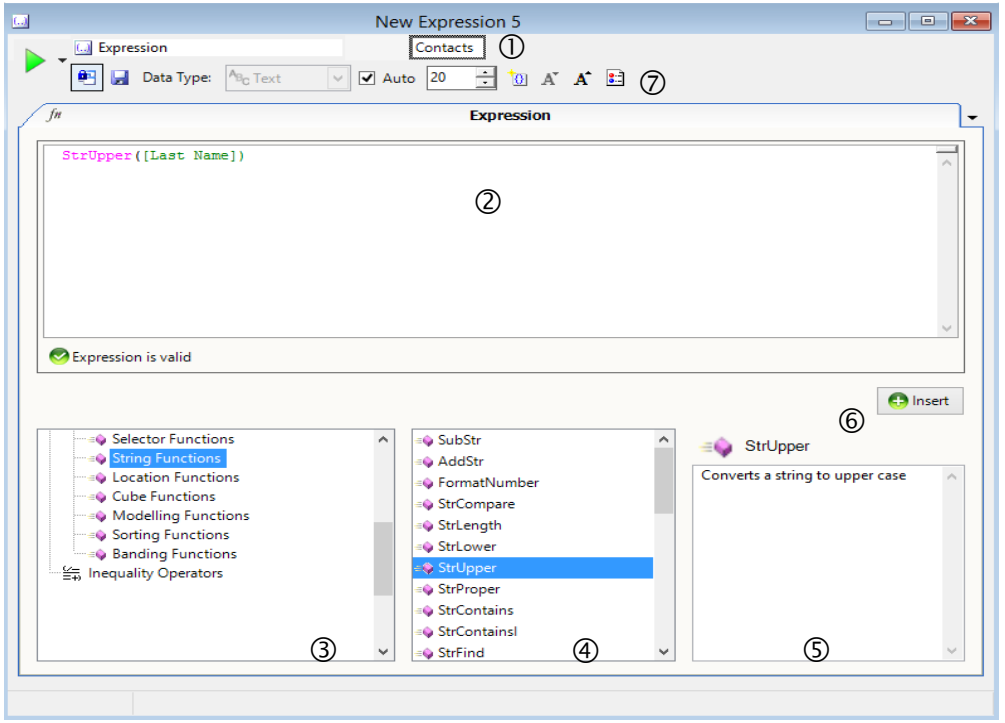
The Expression Window

The Expression tool can be found under the heading Calculate in the Toolbox window. The components of this window are as described below:

1. Set the table level and data type relating to your expression e.g. The variable Last Name is used as a text variable at the Contacts table
2. The expression builder area where the expression is created
3. The section containing the different groups of functions
4. The section containing a list of functions for a selected group
5. The section containing a brief description of the function selected
6. The button that will insert the function selected into the expression builder window
7. The button which allows you to preview an expression on a selection

The example opposite has used a string function called StrUpper that converts a string of text to uppercase. By placing the variable you want to convert in closed brackets after the function you can then display the results by dragging onto a Data Grid.

The Data Grid screen shot opposite shows a selection of records with the original Last Name variable displayed alongside the expression altered view.



Duns Number	Last Name	Upper Case Last Name
078759715	Daniels	DANIELS
078759715	Gerson	GERSON
078759715	White	WHITE
078759715	Peterson	PETERSON
078759715	Shurtleff	SHURTLEFF
078759715	Spurling	SPURLING
078759715	Radler	RADLER
078759715	Duff	DUFF

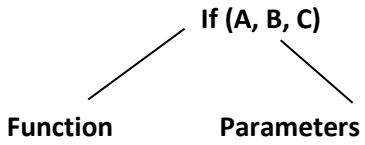
Components of an Expression

As an example take the If logic function to explore the makeup of an expression.

- Drag the **Expression** tool from the **Toolbox** onto the workspace.

This window allows you to build an Expression as shown in the screen shot opposite.

The breakdown of this Expression is as follows:



You could read this as:

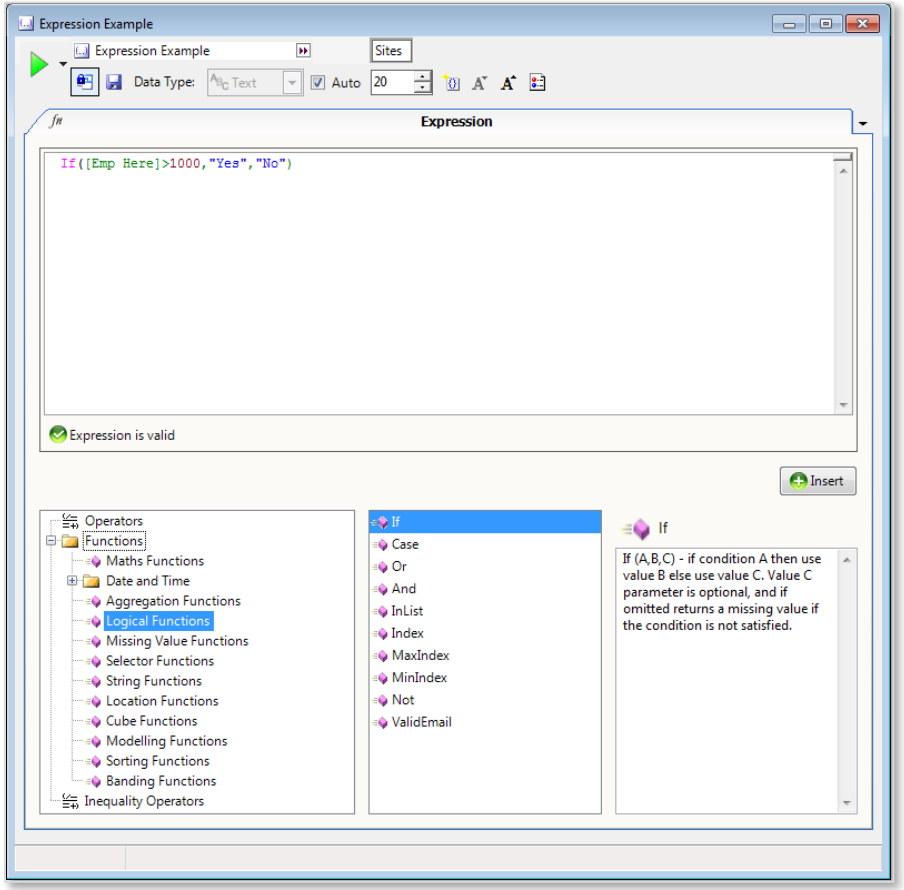
If (A [condition]

then B [outcome if condition met]

else C [outcome if condition not met])

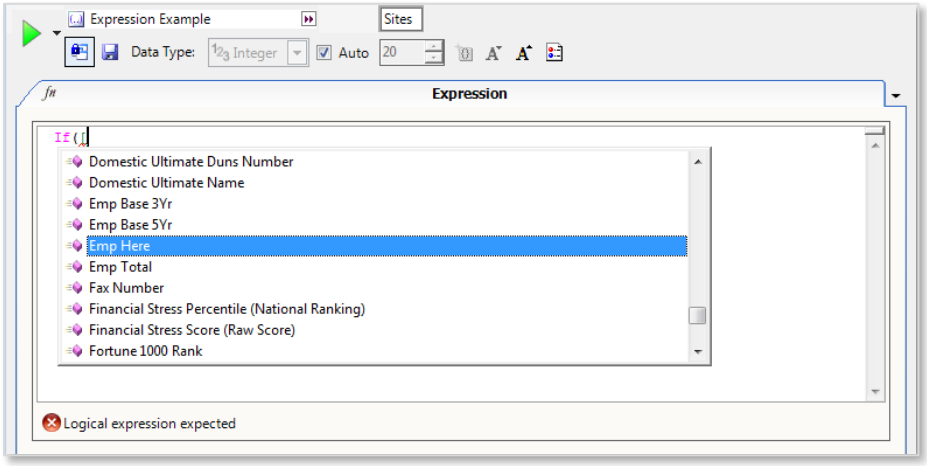
Here a Condition is the test between two values. Those values can result from a Field, Numeric, String, Date or another Expression.

The example opposite has used a test on a Field (Emp Here is greater than 1000) and if that test is met display the word Yes otherwise display the word No. This can be seen when used on a Data Grid.



Creating an Expression

- Change the **Table** to **Sites** and the **Data Type** to **Text**
- Select the **If** statement from the **Logical Functions** option in the **Functions** folder and then click the **Insert** button
- Next to the open black bracket type a left hand square bracket. This will display the available variables that can be used with this function
- Double click on the **Emp Here** variable. Alternatively you could have typed the variable name within square brackets to obtain the same result

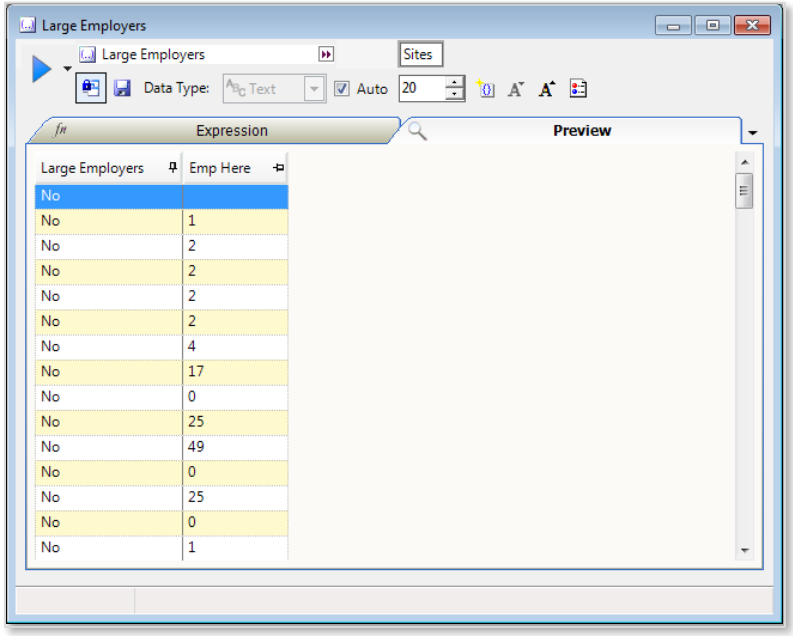


The next part of the Expression is to create a test to find values greater than 1000, therefore you need to insert an Inequality Function.

- From the **Functions** button select **Inequality Functions** and the **> Greater Than** option. Click **Insert**. You may find it quicker to type the symbol directly after the variable
- Type **1000** followed by a comma

The next part of the Expression is to determine the output when the condition is met and when it is not met. As this example is outputting a word (String) in each case you need to ensure they are enclosed in double quotes.

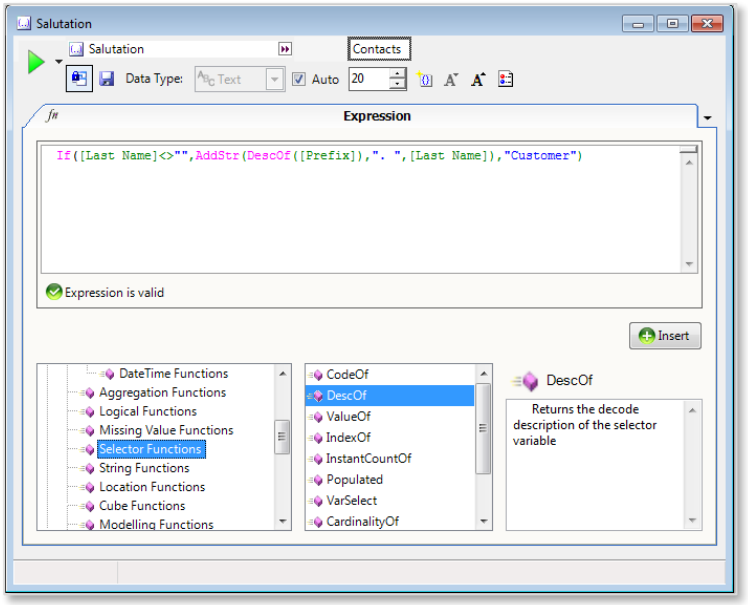
- Type **"Yes","No"** followed by a closing bracket **)**
- Name the Expression window **Large Employers**
- Press the **Build** button to see a **Preview** of the results. This can now be used in conjunction with the **Data Grid** tool



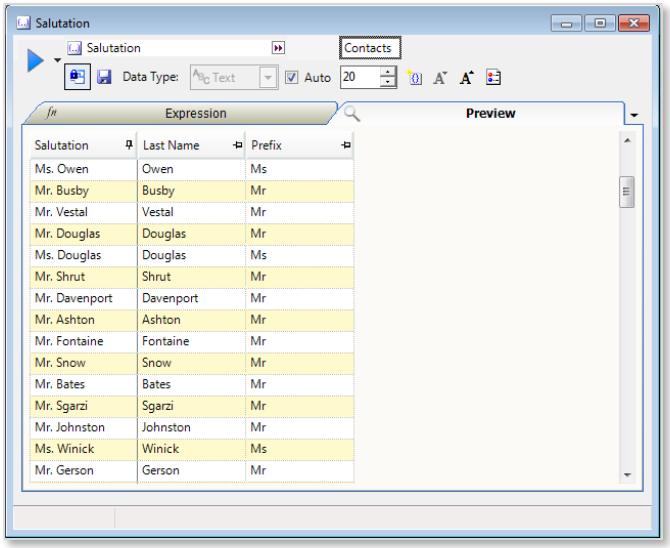
Creating a Salutation Expression

Building upon the previous example you will now see how you can use variable data as part of the output and also combine the results in the display.

Here you will test to see if your records hold a customer's name in preparation for a mail shot. If a record does hold a name the letter will start *Dear* [Title]. [Surname] e.g. Mr. Smith. Otherwise the letter will start *Dear* Customer.



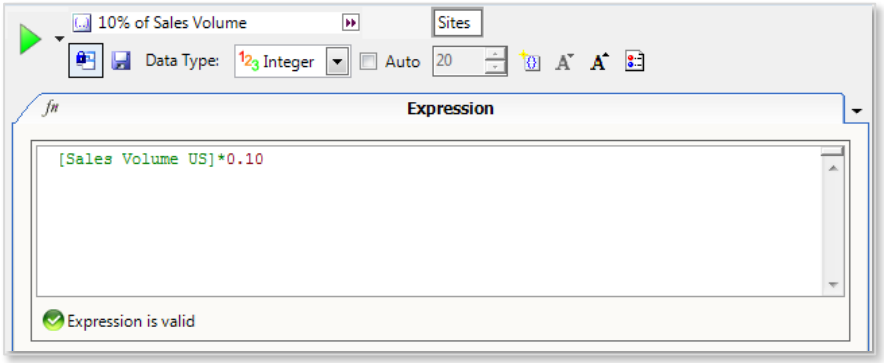
- Open an **Expression** window and select the **If(** function, as in the last example
- Change the **Data Type** to **Text**
- Drag the **Last Name** variable on after the bracket and then type **<> "",** to test if there is a text value
- From the **Strings Function Category** of the **Function** window select **AddStr(**
- Insert after the bracket from the **Selector Functions** **DescOf([Prefix]),". ",[Last Name])** which will display e.g. Mr. Smith if a name is present
- Type a **,** after the bracket and then **"Customer")**. This will then display the word Customer if no name is present
- Click the **Build** button to see a **Preview**



Expressions and Cubes

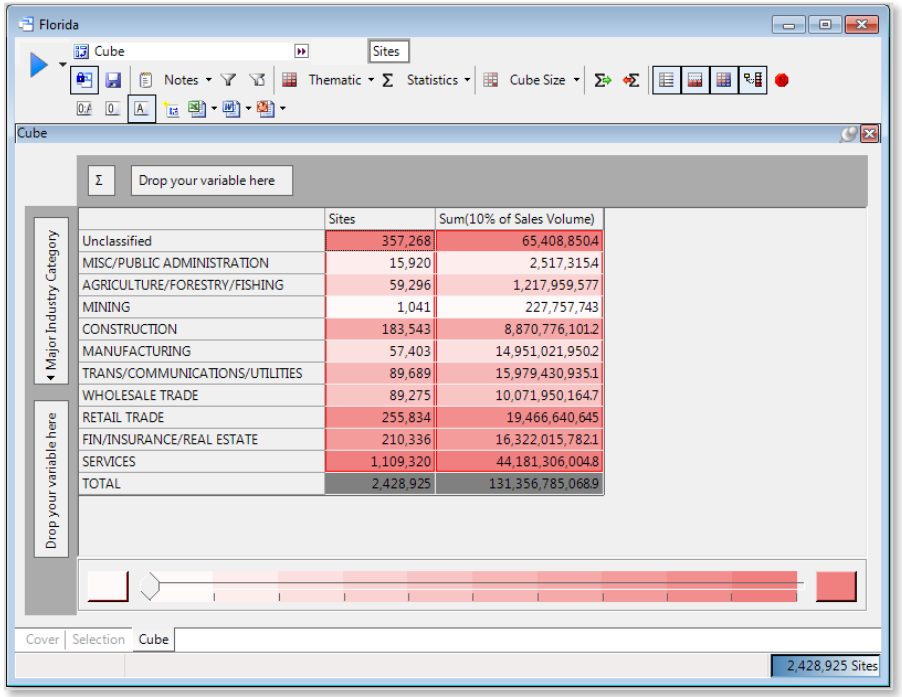
In this example an Expression will be used as a statistic on a Cube display. The Expression itself will calculate 10% of total sales volume for sites in Florida broken down by Major Industry Category.

- Open an **Expression** window
- Ensure the **Table** is set to **Sites** and the **Data Type** is set to **Integer**
- Drag on the **Sales Volume US** variable
- Type ***0.10** which will multiply the sales volume by 0.10 to calculate 10%



To use this expression to help find 10% of total Sales Volume in Florida:

- Drag out the **State** variable and select **Florida**. Ensure the **Table** is set to **Sites**
- Drag on a **Cube** and set the vertical dimension to **Major Industry Category**
- Drag your **10% of Sales Volume Expression** onto the center of the **Cube**
- Press the **Build** button

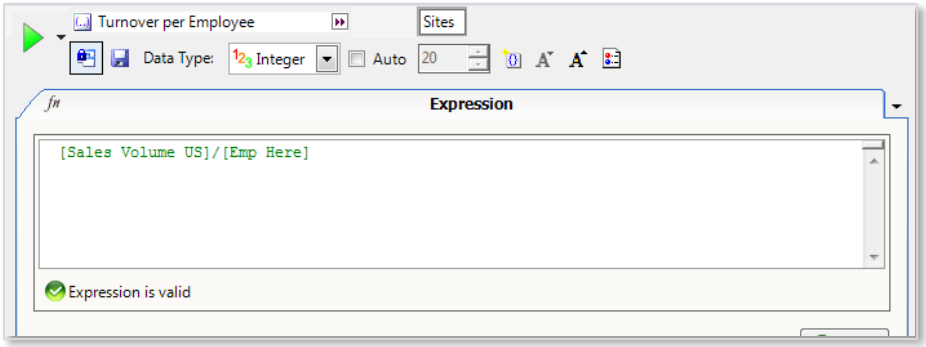


	Sites	Sum(10% of Sales Volume)
Unclassified	357,268	65,408,850.4
MISC/PUBLIC ADMINISTRATION	15,920	2,517,315.4
AGRICULTURE/FORESTRY/FISHING	59,296	1,217,959,577
MINING	1,041	227,757,743
CONSTRUCTION	183,543	8,870,776,101.2
MANUFACTURING	57,403	14,951,021,950.2
TRANS/COMMUNICATIONS/UTILITIES	89,689	15,979,430,935.1
WHOLESALE TRADE	89,275	10,071,950,164.7
RETAIL TRADE	255,834	19,466,640,645
FIN/INSURANCE/REAL ESTATE	210,336	16,322,015,782.1
SERVICES	1,109,320	44,181,306,004.8
TOTAL	2,428,925	131,356,785,068.9

The display now shows the number of Sites in each Major Industry Category in the state of Florida. Also it shows 10% of Sales Volume for all those sites in Florida.

Calculated Expression on a Data Grid

In this example an Expression will be used as a statistic on a Data Grid display. The Expression itself will calculate turnover by employee for Sites in California.

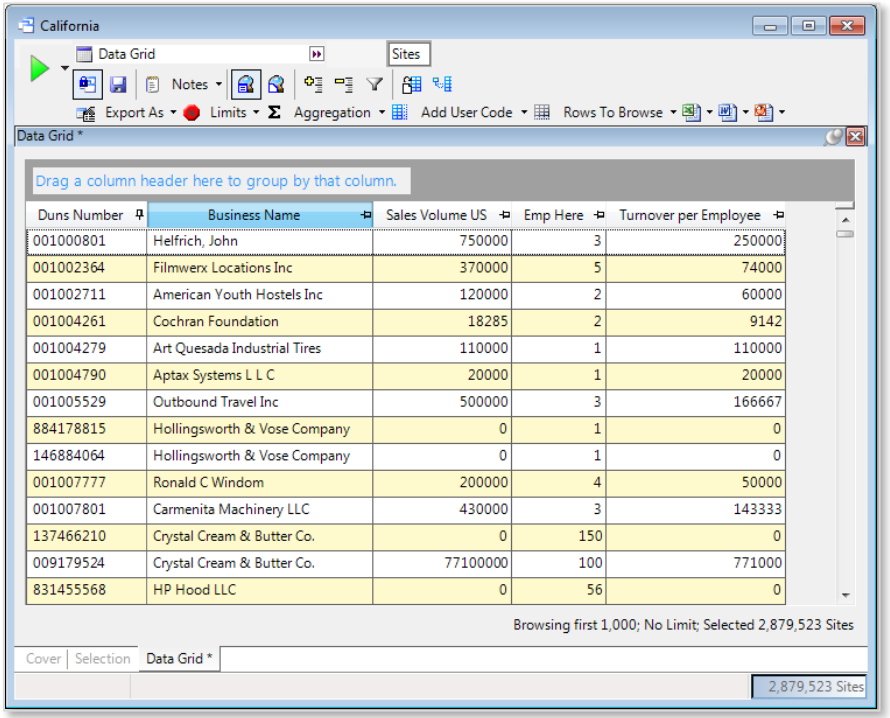


- Open an **Expression** window
- Ensure the **Table** is set to **Sites** and the **Data Type** is set to **Integer**
- Drag on the **Sales Volume US** variable
- Type **/** and then drag on the **Emp Here** variable

To use this expression to display the turnover per employee of businesses in California:

- Drag out the **State** variable and select **California**. Ensure the **Table** is set to **Sites**
- Drag on a **Data Grid** and add the variables **Business Name**, **Sales Volume US** and **Emp Here**
- Drag your **Turnover per Employee** expression onto the Data Grid alongside the other variables
- Press the **Build** button

The display now shows a row for each site in California with the relevant information.



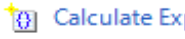
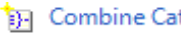
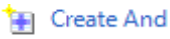
Duns Number	Business Name	Sales Volume US	Emp Here	Turnover per Employee
001000801	Helfrich, John	750000	3	250000
001002364	Filmwerx Locations Inc	370000	5	74000
001002711	American Youth Hostels Inc	120000	2	60000
001004261	Cochran Foundation	18285	2	9142
001004279	Art Quesada Industrial Tires	110000	1	110000
001004790	Aptax Systems L L C	20000	1	20000
001005529	Outbound Travel Inc	500000	3	166667
884178815	Hollingsworth & Vose Company	0	1	0
146884064	Hollingsworth & Vose Company	0	1	0
001007777	Ronald C Windom	200000	4	50000
001007801	Carmenita Machinery LLC	430000	3	143333
137466210	Crystal Cream & Butter Co.	0	150	0
009179524	Crystal Cream & Butter Co.	77100000	100	771000
831455568	HP Hood LLC	0	56	0

Appendix 1 – Wizards & Virtual Variables

Not all of the following wizards may be available to you and will be dependent upon your system configuration.

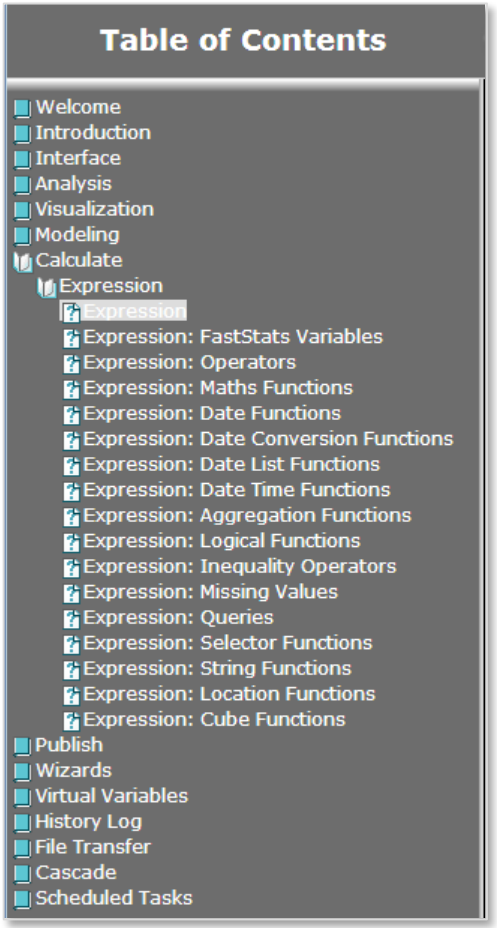
 Count	This wizard will help you create a new virtual variable using the records selected from a RFV analysis, based upon the Frequency option.
 Recency	This wizard will help you create a new virtual variable at a higher table using the RFV Recency option when a lower level variable is used. The results can be displayed in terms of another lower level variable.
 Aggregation	This wizard will help you create a new virtual variable at a higher table using the RFV Value option when a lower level numeric variable is used. The results can be displayed in terms of bands or individual values.
 Transaction Analysis	This wizard will help you create a table (Tree) that analysis's the pattern of transactions within a time frame. The results will show the number of times a pattern occurs.
 Basket Analysis	This wizard will help you create a table (Tree) that analysis the pattern of transactions. The results will show the number of times a pattern occurs.
 Best Fit Prospects	This wizard provides a simple way to create a model of how one selection of records (the Analysis selection) "fits" to another selection.
 Decision Tree Model	This wizard provides a way of capturing the model from the Decision Tree in the form of a Virtual Variable where the categories correspond to the selection rules associated with the nodes of the Decision Tree.
 PWE Model	This wizard provides a way of accessing the PWE scores generated in Profiler in the form of a Virtual Variable. This could be as a series of banded categories or as a numeric variable.

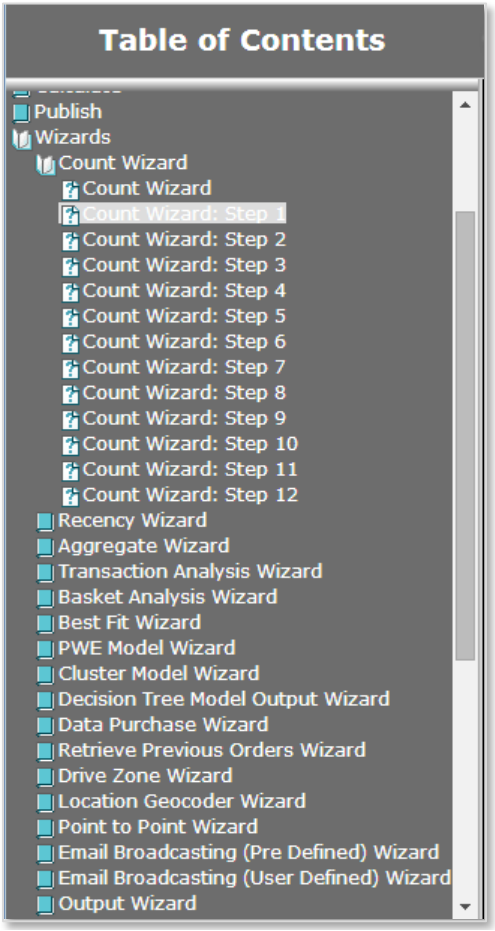
 Duns Transformations	This wizard allows you to perform corporate family tree processes for groups of records. For example, you can find the ultimate parent companies of a group of selected records.
 Drive Zone	This wizard allows you to create a categorical variable that will display travel time or as the crow flies bandings from a geographical location. The end point is defined by a geographical selector.
 Location Geocoder	This wizard allows you to create two virtual variables to identify the Latitude and Longitude of a given set of records.
 Point to Point	This wizard allows you to create a numeric variable that will calculate the travel time, road distance or as the crow flies value from a geographical location. The end point can be defined by a geographical selector.
 Territories	This wizard allows you to create a categorical variable based upon territories defined within Microsoft MapPoint. Each defined category will be represented as a band within the variable.
 Output	This wizard allows you to output records selected through various tools. Currently this wizard supports the Cube, Data Grid and Tree tools.
 SalesForce.COM	This wizard allows you to upload targeted data directly into SalesForce.com so that prospects identified in the system can be seamlessly integrated into CRM sales activities.
 Data Licensing	The wizard is used in systems where not all data is available to users straight away. This wizard can then be used to select what records (and what fields within those records) should be purchased.
 Retrieve Previous Orders	This wizard allows you to go back to orders that have been made in the past and then generate a URN file containing all the URNs from a collection of 1 or more orders.

 Search	This wizard gives the option for certain users to make standard selections without getting confused by the long list of variables available in the System Explorer.
 Initialise	This wizard will help you create a framework for a new virtual variable, where you define the table level and number of categories to be included.
 Assign Values	This wizard will help you define your virtual variable by assigning Descriptions and a Selection to your categories.
 Date Banding	This wizard will help you create a banding based upon a date variable. You are able to define a time period and the number of periods you wish to select upon.
 Numeric Banding	This wizard will help you create a banding based upon a numeric variable. Select from 4 banding options to determine the selection display.
 Calculate Expression	This wizard will help you to create a new numerical virtual variable from a mathematical expression.
 Import Data	This wizard will help you import data into a variable using a key code, which can then be completed by using the Assign Values wizard to add descriptions if necessary.
 Combine Categories	This wizard will help you create a variable by combining the categories within an existing variable.
 Create And Update	This wizard allows you to create a Selector or Flag Array variable based upon selections made within your Market Insight system. This method of variable creation also allows you to edit the variable at a later stage. To create a Numeric or Text variable you will still need to use the Initialise and Assign values wizards, however you will not be able to edit these variables.

Appendix 2 - Expressions

The following pages represent Expression information that can also be found in the D&B Market Insight Help files. Refer to the software Help for the most up to date list of Expression functions and Wizards.





Expressions: Operators

The standard operators supported in expressions are as follows:

+ Add	Addition [Cost] + [Vat] [Cost] + 1000	**- Subtract**	Subtraction [Total] - [Discount] [Total] - 100
*** Multiply**	Multiplication [Cost] * Quantity [Total] * 0.175	**/ Divide**	Division [Total Cost] / [No. of Occupants] [Cost] / 5
% Modulus	Modulus - Divides the value of one expression by the value of another, and returns the remainder [Value] % [Cost] [Value] % 5	**^ Power**	Power - The power operator a^b calculates the b-th power of a [Value] ^ [Numeric] [Value] ^ 3

Note: Operators can be used together. Inequality operators are also available, for use in logical functions.

[Cost] / 100 * 17.5

[Cost] + ([Cost] / 100 * 17.5)

Expressions: Math Functions

Market Insight supports the following mathematical functions:

Log	Log base 10 Log([Value])	**Abs**	Absolute Value Abs([Value])
Sqrt	Square Root Sqrt([Value])	**Ceil**	Round up Ceil([Value])
Floor	Round down Floor([Value])	**Ln**	Natural log Ln([Value])
Exp	Exponential Exp([Value])	**RoundUp**	Round Up to specified precision RoundUp([Value],[precision]) eg RoundUp(123,10) rounds up to the next 10, i.e. 130

RoundDown	Round Down to specified precision RoundDown([Value],[precision]) eg RoundDown(123,10) rounds down to the next 10, i.e. 120	**Round**	Round off to specified precision Round([Value],[precision]) eg Round(123,10) rounds off to the nearest 10, i.e. 120 eg Round(126,10) rounds off to the nearest 10, i.e. 130 egRound(125,10) rounds off to the nearest 10, i.e. 130
Rand	Returns a random number between 0 and its argument, or 0 and 1 if no argument is given Rand([Value])	**Power**	Power Power(x,y) Calculates x to the power of y. Power(x,y) is equivalent to x ^ y e.g. power(3,2) = 3 ^ 2 = 9

Expressions: Date Functions

Date functions are provided to manipulate Market Insight date fields.

Day	Number of days from year 0000 to date Day([Booking Date])	**Week**	Number of weeks from year 0000 to date Week([Booking Date])
Month	Number of months from year 0000 to date Month([Booking Date])	**Year**	Number of years from year 0000 to date Year([Booking Date])
Today	Number of days from year 0000 to today Today()	**AgeDays**	Age of date in days relative to today AgeDays([Booking Date])
AgeWeeks	Age of date in weeks relative to today AgeWeeks([Booking Date])	**AgeMonths**	Age of date in months relative to today AgeMonths([Booking Date])

AgeYears	Age of date in years relative to today AgeYears([Booking Date])	**DateDay**	Day of date 1.. 31 DateDay([Booking Date])
DateWeek	Week of date 1..53 DateWeek([Booking Date])	**DateMonth**	Month of date 1=Jan, 2=Feb, … 12=Dec DateMonth([Booking Date])
DateQuarter	Quarter of date 1=Q1, 2=Q2, 3=Q3, 4=Q4 DateQuarter([Booking Date])	**DateYear**	Year of date e.g. 2005 DateYear([Booking Date])
DayOfWeek	Day of date 1=Monday, 2=Tuesday, … 7=Sunday DayOfWeek([Booking Date])	**Date2Days**	The number of days since 0000 Date2Days(2006,01,01)

MakeDate	Makes a constant date from numbers MakeDate(Years, Months, Days) MakeDate(2007,08,22) MakeDate(DateYear([Booking Date],01,01) returns the start of the year in which the booking was made.	**DateShift**	DateShift – shift a time by a number of units (years, quarters, months, weeks, days). DateShift([Date],[Date],"Units") DateShift([BookingDate],30,"Days") calculates 30 days after booking. DateShift([DepartureDate],-30,"Days") calculates 30 days before departure. DateShift([DOB],18,"Years") calculates 18th birthday
DayOfYear	Calculates the day number in the year 1 .. 366 DayOfYear([date]) DayOfYear(20070105) = 5	**BuildDate**	The Market Insight system build date shifted by n days BuildDate(n) e.g. BuildDate (0) = build date BuildDate (-1) = day before build date
DateDiff	Calculates [Date] - [Date] measured in the specified units (years, quarters, months, weeks, days). DateDiff([Date], [Date], "Units") Units specifier is optional. If missing, the default units are days. If only one date is specified, the given date is compared to today. Examples opposite:		DateDiff([DepartureDate],[Booking Date],"Days") Is the number of days between departure and booking. DateDiff([DepartureDate],20051231,"Days") Is the number of days between departure and 31 Dec 2005. DateDiff([DepartureDate],Today(0),"Days") Is the number of days until departure. DateDiff([DepartureDate],[Booking Date],"Weeks") Is the number of weeks between departure and booking.

Expressions: Aggregation Functions

Aggregation functions take between 2 and 255 parameters, and result in a single value.

Min	Minimum of a list Min(2.3, 3.4, 5.2, 2.2) Min([Value1], [Value2], [Value3])	**Max**	Maximum of a list Max(2.3, 3.4, 5.2, 2.2) Max([Value1], [Value2], [Value3])
Mean	Mean average of a list Mean(2.3, 3.4, 5.2, 2.2) Mean([Value1], [Value2], [Value3])	**Median**	Median average of a list Median(2.3, 3.4, 5.2, 2.2) Median([Value1], [Value2], [Value3])
StdDev	Standard deviation average of a list StdDev(2.3, 3.4, 5.2, 2.2) StdDev([Value1], [Value2], [Value3])	**Mode**	The Modal average of a list Mode(1, 2, 3, 4) Mode(1,2,3,2,4,2,3) is 2 The mode is undefined if there is a tie but by convention we return the first tied candidate. Mode(1,7,3,7,4,7,3,1,3) is 3

Expressions: Logical Functions

Logical functions in Market Insight allow conditional operation.

They can also be used to set a value according to the result. The value returned is 0.0 if the expression evaluates to false, 1.0 if the expression evaluates to true, or missing value if the answer is undefined due to missing values.

If	If (A,B,C) - if condition A then use value B else use value C.
	If (A,B) - if condition A then use value B else use missing_value.
	
	If([Cost] < 1000 , 1, 0)
	If(AgeWeeks([Booking Date]) <= 52,0,69)
		Case	Case(A,a,B,b,C,c ...) – if condition A then use value a, condition B then use value b, etc..
			
			Case([Cost] = 100, 1,
			[Cost] = 200, 2,
			[Cost] = 300, 3)
			
			Case([Cost] < 500, 1,

Or	Or(A,B,C...) - evaluates to true if condition A is true **OR** condition B is true etc., otherwise evaluates to false. If(Or ([Cost]<1000,[Cost]=1000),1,0)	**And**	And(A,B,C...) - evaluates to true if condition A is true **AND** condition B is true etc., otherwise evaluates to false. If(And([Cost]>500,[Cost]<1000),1,0)
InList	InList(X,A,B,C ...) - true if expression X takes value of expression A or B or C ... if(InList(790.00, [Cost],[Cost]+10,[Cost]-10), 1,		

Expressions: Inequality Operators

Inequality operators are used within conditions in Logical Functions.

They can also be used to set a value according to the result. The value returned is 0.0 if the expression evaluates to false, 1.0 if the expression evaluates to true, or missing value if the answer is undefined due to missing values.

>	Greater than [Cost] > 1000 [Cost] > [Revenue]	**>=**	Greater than or equal to [Cost] >= 1000 [Cost] >= [Revenue]
<	Less than [Cost] < 1000 [Cost] < [Revenue]	**<=**	Less than or equal to [Cost] <= 1000 [Cost] <= [Revenue]
=	Equal to [Cost] = 1000 [Cost] = [Revenue]	**<>**	Not equal to [Cost] <> 1000 [Cost] <> [Revenue]

Expressions: Missing Values

Market insight handles Missing values in the context of expressions and operators.

IsMissing	Determines whether a value is Missing, returning true or false. IsMissing([Cost])

Expressions that reference Missing values are evaluated as follows:

Expression	**Result**	**Expression**	**Result**
Missing + Missing	Missing	Value + Missing	Value
Missing - Missing	Missing	Value - Missing	Value
Missing - Value	-Value	Missing * Value	Missing
Missing / Value	Missing	Value / Missing	Missing
Missing / 0	Missing	Missing % Value	Missing

Expression	Result	Expression	Result
Value % Missing	Missing	Missing % 0	Missing
Missing ^ Value	Missing	Value ^ Missing	Missing
Value > Missing	true	Missing > Value	false
Missing > Missing	false	Value < Missing	true
Missing < Value	false	Missing < Missing	false
Value >= Missing	true	Missing >= Value	false
Missing >= Missing	false	Value <= Missing	true
Missing <= Value	false	Missing <= Missing	false

Expression	Result	Expression	Result
Value = Missing	false	Missing = Missing	false
Value <> Missing	true	Missing <> Missing	false
Function(Missing)	All single parameter functions except IsMissing() evaluate to Missing		

Expressions: Date Conversion Functions

To use a date as a string in an expression it is necessary to specify the format. Dates are formatted into strings by using the **FormatDate** function.

FormatDate	Converts a date variable to its string representation FormatDate([Date Variable],"Format Specifier")

The following expressions demonstrate its use:

Expression	Result
FormatDate([Date Variable],"%d-%m-%Y")	29-09-2006
FormatDate([Date Variable],"")	2006-09-29
FormatDate([Date Variable],"%A, %B %#d")	Friday, September 29
FormatDate([Date Variable],"Date: %#x")	Date: Friday, 29 September 2006

The format specifiers are detailed below, with examples of the output produced by each format specifier.

Format Specfier	Description	Example Output
%a	Abbreviated weekday name	Wed
%A	Full weekday name	Wednesday
%b	Abbreviated month name	Feb
%B	Full month name	February
%d	Day of month as decimal number (01 – 31)	31
%j	Day of year as decimal number (001 – 366)	047
%m	Month as decimal number (01 – 12)	01
%U	Week of year as decimal number, with Sunday as first day of week (00 – 53)	09
%w	Weekday as decimal number (0 – 6; Sunday is 0)	0

%W	Week of year as decimal number, with Monday as first day of week (00 – 53)	09
%x	Date representation for current locale	07/03/2000
%y	Year without century, as decimal number (00 – 99)	06
%Y	Year with century, as decimal number	2000
%z, %Z	Either the time-zone name or time zone abbreviation, depending on registry settings; no characters if time zone is unknown	GMT Standard Time
%%	Percent sign	%

The # flag may prefix any formatting code. In that case, the meaning of the format code is changed as follows.

Format Code	Meaning
%#x	Long date representation, appropriate to current locale. For example: "Tuesday, March 14, 1995".
%#d, %#H, %#I, %#j, %#m, %#M, %#S, %#U, %#w, %#W, %#y, %#Y	Remove leading zeros (if any).

Expressions: Date List Functions

These functions help you manipulate lists of dates.

FirstDate	Returns the first date in the list of dates. Dates can also be included as yyyymmdd. FirstDate (date1, date2,...) FirstDate([Date1], "20060101") FirstDate([Date1], [Date2], [Date3])
LastDate	Returns the last date in the list of dates. Dates can also be included as yyyymmdd. LastDate (date1, date2,...) LastDate([Date1], "20060101") LastDate([Date1], [Date2], [Date3])

Expressions: Selector Functions

The following functions are available to map the values from selector variables to strings and numerics. Once mapped they can be used with any functions and operators that support those types.

CodeOf	Returns the code of the selector variable CodeOf([Selector Variable]) CodeOf([Income Band]) = "04"	**DescOf**	Returns the decode description of the selector variable DescOf([Selector Variable]) DescOf([Income Band]) = “30 – 40k”
ValueOf	Returns the description of the selector variable as a number ValueOf([Selector Variable]) ValueOf([Income Band]) = 30 as it just takes the first number	**IndexOf**	Returns the index of the code of the selector variable as a number IndexOf([Selector Variable]) IndexOf([Income Band]) is 4. Unclassified is zero

Expressions: String Functions

These functions allow string manipulation.

Left	Left part of a string Left(string, n) Left(“Red Nose Day”,6) = “Red No”	**Right**	Right part of a string Right(string, n) Right(“Red Nose Day”,3) = “Day”
Mid	Middle part of a string Mid(string, start, length n) Extracts n characters from the string. The start position is 1-based. Mid(“Red Nose Day”,5,4) = “Nose”	**FormatNumber**	Formats a number to the given precision FormatNumber(x,ndp) Formats x, where ndp is the number of decimal places. if [Cost] is 56.78 AddStr(“Your bill is £”,[Cost]) is “Your bill is £56.78” AddStr(“Your rounded bill is £”,FormatNumber(Round([Cost]),0) is “Your rounded bill is £56”
SubStr	Returns a portion of a string. SubStr(String, Start, Size) Start is a zero based offset. If variable Surname = "Pants" in current record SubStr([Surname],1,3) returns "ant" SubStr([Surname],0,3) returns "Pan"	**AddStr**	Concatenates strings together. AddStr([Text],[Text],[Text],…) AddStr("Dear ",[Title]," ",[Initial]," ",[Surname]) AddStr([Title],” “,[Initial],” “,[Surname]) might return “Mr J Smith”.

StrCompare	Compares two strings StrCompare(string, string , case_sensitive) Case_sensitive is an optional numeric parameter. If case_sensitive evaluates to 1 then the comparison is case sensitive. If case_sensitive evaluates to 0 or missing_value or is not provided then the comparison is not case sensitive. StrCompare(string1, string2) compares two strings alphabetically and returns: 1 if [string1] > [string2] -1 if [string1] < [string2] 0 if [string1] = [string2]	**StrLength**	Gets length of a string StrLength(string) StrLength ("Apteco") = 6
StrLower	Converts a string to lower case StrLower(string) StrLower("Apteco") = "apteco"	**StrUpper**	Converts a string to upper case StrUpper(string) StrUpper("Apteco") = "APTECO"
StrProper	Converts a string to proper case StrProper(string) StrProper case capitalises the first letter after a space or punctuation character. StrProper (“redfern house”) = “Redfern House”	**StrReverse**	Reverses a string StrReverse(string) StrReverse("Apteco") = "ocetpA"

StrContains	Finds substrings within a string StrContains(x, a, b, c, ...) StrContains searches string x for substrings a, b, c, etc., and returns the index of the first match found, or zero if none are found. StrContains(“Huddersfield Town”,”City”,”Town”) = 2 StrContains(“Smith”,”Smith”,”Jones”,”Walker”) = 1 StrContains(“Huddersfield Town”,”udders”,”cow”,”field”) = 1 StrContains(“Huddersfield Town”,”Udders”,”Cow”,”Field”) = 0 StrContains(“[email]”,”hotmail”,”yahoo”,”aol”,”ntlworld”) detects if the email field has any of the listed ISPs. Note StrContains is case sensitive. Use StrUpper / StrLower to equalise the case if case insensitive match is required. StrContains(StrUpper(“Huddersfield Town”),”UDDERS”,”COW”,”FIELD”) = 1	**StrFInd**	Locates a substring within a string StrFind(search_string, target_string) Returns the zero based start offset of the substring or -1 if not found found. StrFind (“Huddersfield Town”,”Town”) = 13 StrFind (“Huddersfield Town”,”Hudd”) = 0 StrFind (“Huddersfield Town”,”Udders”) = -1 Note StrFind is case sensitive. Use StrUpper / StrLower to equalise the case if case insensitive match is required.

Combining String Functions with Missing Values functions and Logical Functions can create powerful expressions. The following example creates a salutation, like 'Dear Mr J Smith'. If the surname is missing for the record, the salutation would be 'Dear Customer'.

```
AddStr("Dear ",If(IsMissing([Surname]),"Customer",AddStr(DescOf([Title])," ",[Surname])))
```