TARGIT Data Discoverer

Level: Intermediate

TARGIT Decision Suite 2019.0 – document version 3.3 US

TARGIT A/S

Aalborgvej 94

DK-9800 Hjørring

Denmark

Phone: (+45) 9623 1900

Telefax: (+45) 9623 1999

E-mail: info@targit.com

Internet: http://www.targit.com

Contents

Introduction to Data Discovery 5
Requirements 5
R language requirement 5
Source Data download 5
Lesson 1: Adding single files 7
Raw data and Measure aggregation types 10
Aggregation types 10
Updating the source file 11
Exercises Lesson 1 13
Lesson 2: Fine-tuning Source Data Detection 15
Changing Column Data Types 15
Changing Aggregation Type 17
Showing / Hiding fields in the cube 19
Creating a hierarchy 20
'Order' and 'Member properties' 21
Exercises Lesson 2 22
Lesson 3: Data mashup 25
Adding a new Data Source 25
Exercises Lesson 3 30
Lesson 4: Formats – enriching data 38
Functions Library 38
Create a Format 39
Exercises Lesson 4 42
Lesson 5: Accessing Data Warehouse data 51
TARGIT Analyses as Data Sources 51
Un-pivoting pivoted data 53
Exercises Lesson 5 56
Lesson 6: Merging data with R 63
Creating an R language data source 63
Exercises Lesson 6 66
Lesson 7: Data Discovery Collaboration 67
Sharing Data Sources and Cubes 67
Setting Permissions 68
Exercises Lesson 7 68
Lesson 8: Administrator options 69
Data Service Settings 70
Exercises Lesson 8 70

Introduction to Data Discovery

TARGIT Data Discovery enables an end-user to consume data from a wide variety of data sources and combine them with other data without knowing about data types and complicated query languages. We call these combinations for cubes.

TARGIT Data Discovery is not meant to be a replacement for a traditional BI solution, but rather be the add-on that gives business analysts and other user types with basic understanding of data structures a tool to instantly deploy company-wide solutions in a collaborative spirit.

The engine of the TARGIT Data Discovery module is a service called *TARGIT Data Service*. The TARGIT Data Service is an in memory engine designed to work with datasets of millions of rows and must be installed on the same server hosting the ANTserver component.

TARGIT Data Service will work with up to two million rows of data out of the box, but with the addition of the ***Data Discovery* license module** to the TARGIT solution, it can exceed the two million rows limit – limited only by memory.

Please note that installation and configuration requires knowledge of Windows Server, IIS configuration.

Requirements

The following requirements must be present for successful installation of the TARGIT Data Discovery module:

- TARGIT Decision Suite 2015 or newer
- A newer browser (Chrome, IE10 and newer)
- A valid license that includes the Data Discovery module
- .NET Framework 4.5 or newer
- Enabled Windows Features for:
    - WCF Activation over port and http
    - IIS ASP.NET 4.5

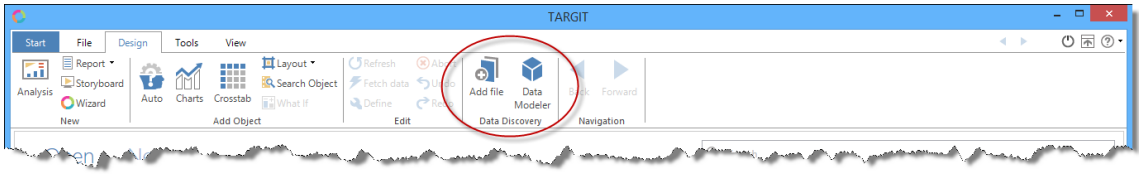
R language requirement

Some of the content of this manual requires the open source 'R' programming language to be installed. Go to the R homepage at https://www.r-project.org/ or go directly to one of the mirrored download pages, e.g. http://cran.uib.no/.

Source Data download

The demos and exercises in this manual are based on a number of different data sources. Some of them are supposed to be accessed from the Internet during the course, while others are to be found in a package of prepared files. The package of prepared files can be downloaded from the TARGIT Portal http://portal.targit.com. Go to the *Download Center, Other Downloads* – in the *TARGIT University* section you can find the *DataDiscoverySourceDataFiles.zip* compressed file. Unpack the content to a C:\TU folder.

Lesson 1: Adding single files



The "**Add file**" option is the option we would like to exploit as part of this end-user training. This is where you can easily add ad-hoc data – typically single Excel files or single CSV files – and instantly use the TARGIT client to analyze data from those files.

The "**Data Modeler**" option is for more advanced use – including combining data from multiple sources. E.g. combining Excel data with data from a relational database.

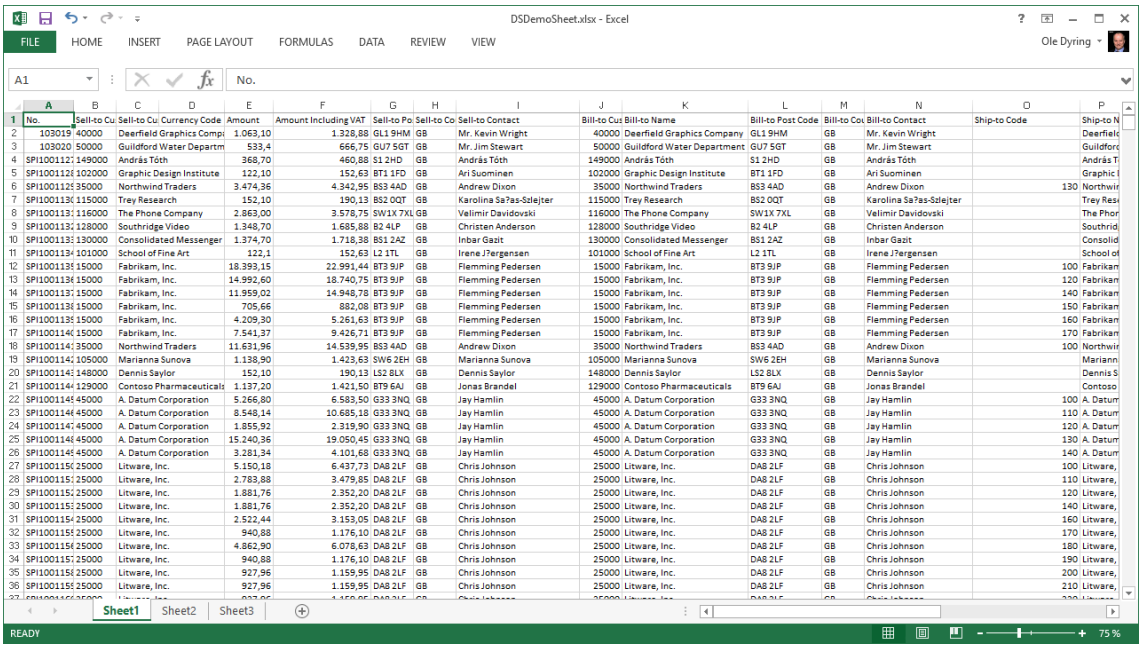
In the following, in the demo description as well as in the exercises for this lesson, we will be using a couple of Sample Excel files. These file scan be downloaded from the TARGIT Portal, http://portal.targit.com, in this section:

- Download Center / Other Downloads / TARGIT University.
- The zipped file is called "Data Discovery source data files".

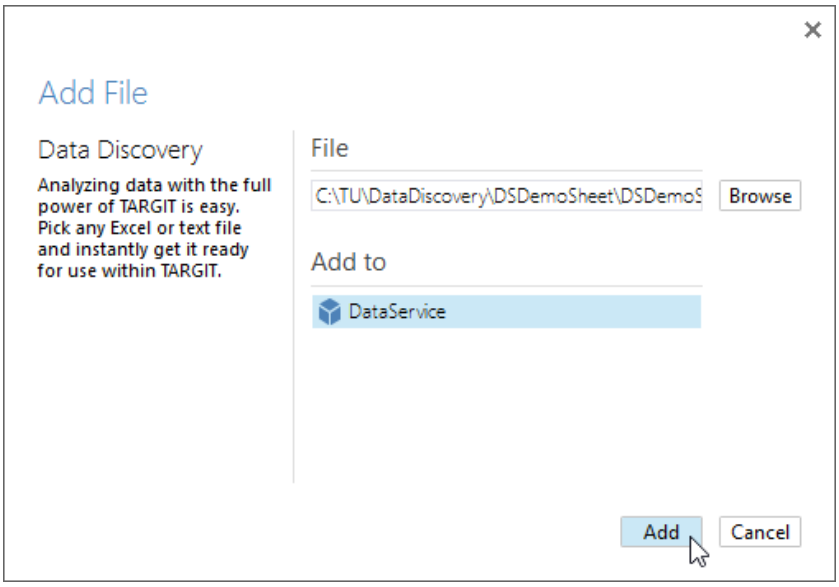
If we have a look at the DSDemoSheet.xlsx file, we will see a typical extract from a database in a format that adheres to the required format for Data Discovery. This file contains approximately 4200 rows across 30 columns.

- **Rows**: Each row represent a single transaction.
- **Columns**: The columns represent the information that has been logged for each transaction. Data Discovery will read the content and data types of these columns to automatically detect if they are to be used as measures and/or dimensions.

Notice the Date columns: Posting date, Document date and Shipment date columns. They all contain a full date, and the Cell data type has also been set to “Date”. This will ensure that Data Discovery reads dates correctly and furthermore create correct Time hierarchies from these.

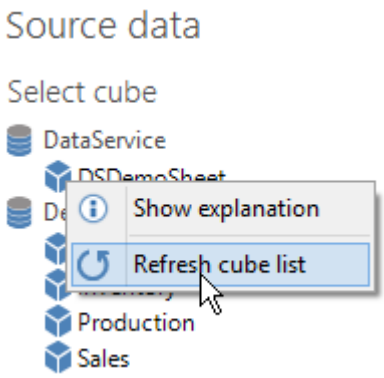


To analyze the data of this file, we should first add it to DataService:

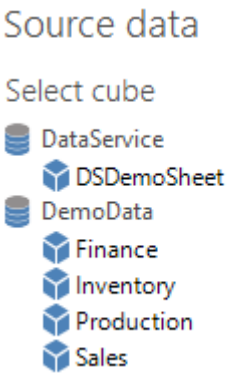


(Note: If the TARGIT Xbone component had been installed, this would be a second option to which source data files could be added.)

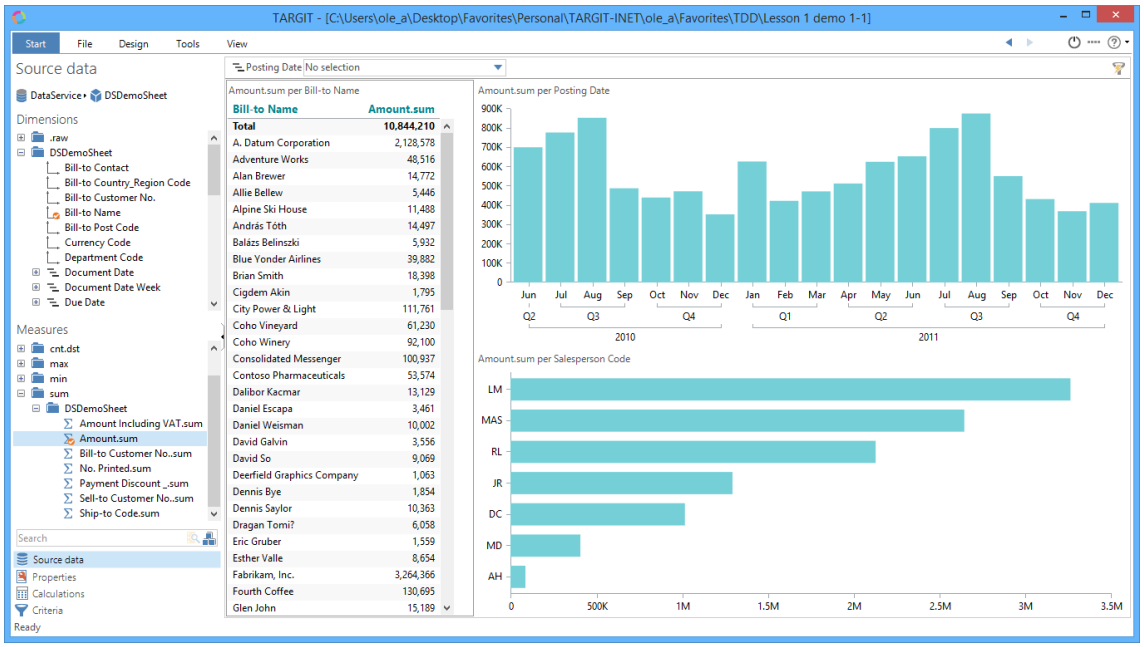
Depending on the size of the file, it may take a little while for it to become visible as a Data Source in the TARGIT Client. You do not need to log off to see it, as it will automatically appear in your *Source data* overview tab. Sometimes, however, you may need to right click one of the existing cubes to refresh the cube list.



Once it is available, it will appear as a new cube in the DataService connection.



Now, the data can be used to create an analysis like this:



Notice that dimensions and measures have been arranged into various Display folders.

Often, you will just be using the Dimension folder named identical to the source file and the measures in the *sum* measure folder.

Raw data and Measure aggregation types

A few of the other Display folders may need a little explanation:

The **.raw** folder contain *all* columns as dimensions – including columns that would normally be regarded as measure columns. This can be useful e.g. if you are analyzing a survey where survey participants have rated statements with a score 1 to 5. With the .raw dimension you would be able to categorize data by score value.

E.g., in this screenshot, Amount.raw has been used as a dimension together with the *.cnt measure.

The information we see here, is that the file contains in total 4,198 records, and that the Amount value of “55,2” appear in 24 of those records.

In the *cnt* measure folder you will find count measures for each dimension and measure, but these are by default *distinct counts*. Only the **.cnt* will give a full count.

Amount.cnt per Amount.raw

Amount.raw	Amount.cnt
Total	4,198
55,2	24
55,6	39
91,2	2
122,1	12
123,3	24
133,76	1
135,28	3
136,8	1
142,4	2
152,1	92
153,3	220
153,8	

Aggregation types

In the screenshot below, a number of different measure types have been added to the crosstab together with the Bill-to Name dimension. Furthermore, to the right, the cross tab based on the No. dimension will in fact show the individual transactions behind the aggregated data – in this case for the selected Bill-to Name, Allie Bellew.

Amount.sum, Amount.avg, Amount.max, Amount.min, Amount.cnt, *.cnt and Bill-to Name.cnt.dst per Bill-to Name

Bill-to Name	Amount.sum	Amount.avg	Amount.max	Amount.min	Amount.cnt	*.cnt	Bill-to Name.cnt.dst
Total	10,894,210	2,595	50,369	55	4,198	4,198	78
A. Datum Corporation	2,128,578	6,652	22,959	134	320	320	1
Adventure Works	48,516	693	3,171	56	70	70	1
Alan Brewer	14,772	410	2,000	55	36	36	1
Allie Bellew	5,446	419	1,015	55	13	13	1
Alpine Ski House	11,488	499	2,055	55	23	23	1
András Tóth	64,497	2,389	50,369	152	27	27	1
Balázs Belinszki	5,932	395	1,015	55	15	15	1
Blue Yonder Airlines	39,882	688	3,565	56	58	58	1
Brian Smith	18,398	497	2,125	55	37	37	1
Cigdem Akin	1,795	256	724	153	7	7	1
City Power & Light	111,761	1,415	4,246	152	79	79	1
Coho Vineyard	61,230	862	3,140	152	71	71	1
Coho Winery	92,100	1,123	3,979	152	82	82	1
Consolidated Messenger	100,937	1,278	3,390	56	79	79	1

Amount.sum per No.

No.	Amount.sum
Total	5,446
SPI1002189	55
SPI1002461	547
SPI1100020	153
SPI1100048	277
SPI1100283	56
SPI1101097	325
SPI1101733	724
SPI1101812	689
SPI1101958	155
SPI1102151	280
SPI1102255	1,015
SPI1102302	1,015
SPI1102357	155

Aggregation type columns explained:

- **Bill-to Name**: This is a standard dimension that has been picked to be displayed along the vertical axis of the crosstab.
- **Amount.sum**: This is probably the most commonly used measure type. For *Allie Bellew*, we can see that the total / summarized / aggregated value is 5,446.
- **Amount.avg**: This is the average value across all transactions. For *Allie Bellew*, the average value is 419, which by the way is equal to Amount.sum divided by *.cnt (5,446 / 13).
- **Amount.max**: This is the maximum value across all transactions.
- **Amount.min**: This is the minimum value across all transactions.

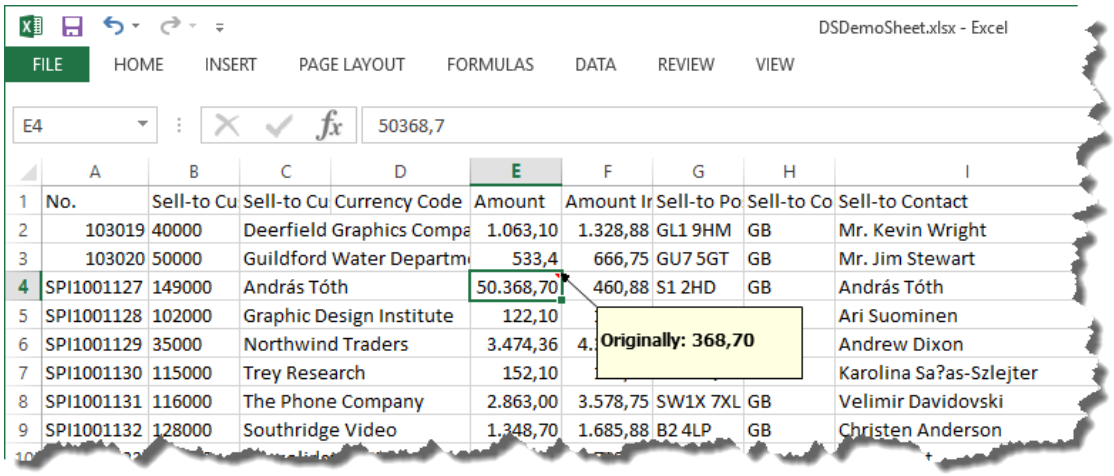
- **Amount.cnt**: This is the *distinct* count of unique Amount values. For *Allie Bellew* it is 11. Notice in the table to the right that the values '155' and 1,015' is represented twice each. This is why the Amount.cnt can be less than the total number of transactions (*.cnt = 13).
- ***.cnt**: Counts the total number of transactions for each dimension member. *Allie Bellew* appear in 13 transactions.
- **Bill-to Name.cnt**: Again, this is a distinct count, so unique appearances of *Allie Bellew* will only count as 1 – as it will with any other customers in this crosstab. From the Total row, however, we may see that we have in total 78 distinct customers.

Updating the source file

Suppose you have already added a file to Data Discovery and created analyses upon these data. Now you have an updated file, and you want to replace the old source file with the new one.

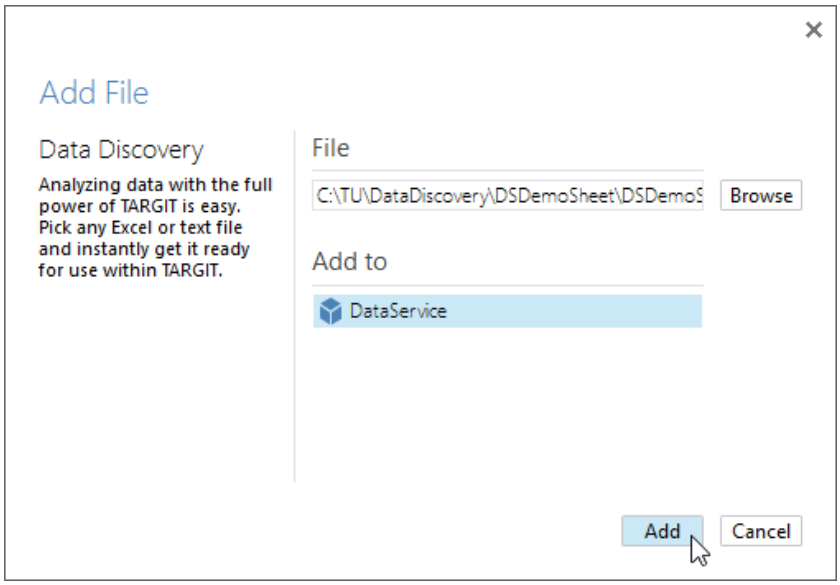
In fact, this is just a matter of adding the updated file to Data Discovery once again. It is a requirement, however, that the new file is named identically to the old file – otherwise you would create a new cube with the new name.

To illustrate this, we can make a simple modification to the current Excel file, e.g. by adding 50,000 to one of the cells in the sheet:

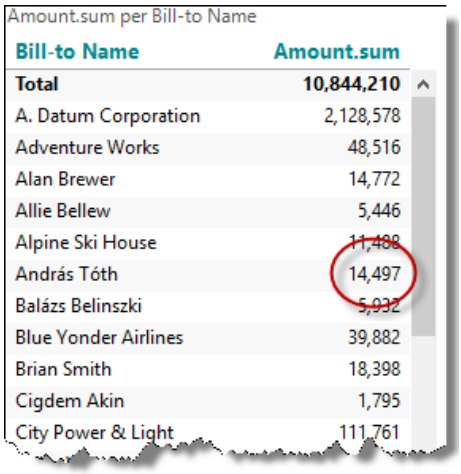


	A	B	C	D	E	F	G	H	I
1	No.	Sell-to Cu	Sell-to Cu	Currency Code	Amount	Amount Ir	Sell-to Po	Sell-to Co	Sell-to Contact
2	103019	40000	Deerfield Graphics Compa		1.063,10	1.328,88	GL1 9HM	GB	Mr. Kevin Wright
3	103020	50000	Guildford Water Departm		533,4	666,75	GU7 5GT	GB	Mr. Jim Stewart
4	SPI1001127	149000	András Tóth		50.368,70	460,88	S1 2HD	GB	András Tóth
5	SPI1001128	102000	Graphic Design Institute		122,10				Ari Suominen
6	SPI1001129	35000	Northwind Traders		3.474,36	4.			Andrew Dixon
7	SPI1001130	115000	Trey Research		152,10				Karolina Sa?as-Szlejter
8	SPI1001131	116000	The Phone Company		2.863,00	3.578,75	SW1X 7XL	GB	Velimir Davidovski
9	SPI1001132	128000	Southridge Video		1.348,70	1.685,88	B2 4LP	GB	Christen Anderson

Now, save the file and re-add it to DataService:

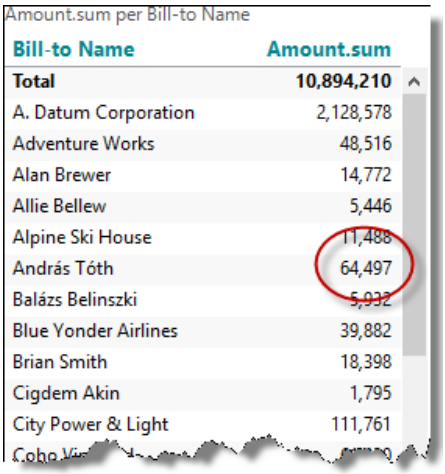


Wait a few seconds, and then refresh your analysis to see the updated data right away.



Amount.sum per Bill-to Name

Bill-to Name	Amount.sum
Total	**10,844,210**
A. Datum Corporation	2,128,578
Adventure Works	48,516
Alan Brewer	14,772
Allie Bellew	5,446
Alpine Ski House	11,488
András Tóth	14,497
Balázs Belinszki	5,932
Blue Yonder Airlines	39,882
Brian Smith	18,398
Cigdem Akin	1,795
City Power & Light	111,761



Amount.sum per Bill-to Name

Bill-to Name	Amount.sum
Total	**10,894,210**
A. Datum Corporation	2,128,578
Adventure Works	48,516
Alan Brewer	14,772
Allie Bellew	5,446
Alpine Ski House	11,488
András Tóth	64,497
Balázs Belinszki	5,932
Blue Yonder Airlines	39,882
Brian Smith	18,398
Cigdem Akin	1,795
City Power & Light	111,761

Exercises Lesson 1

In this exercise and the following exercises, you must use some predefined data sources. These files can be downloaded from the TARGIT Portal, http://portal.targit.com, in this section:

- Download Center / Other Downloads / TARGIT University.
- The zipped file is called “Data Discovery source data files”.
- Unzip to C:\TU.

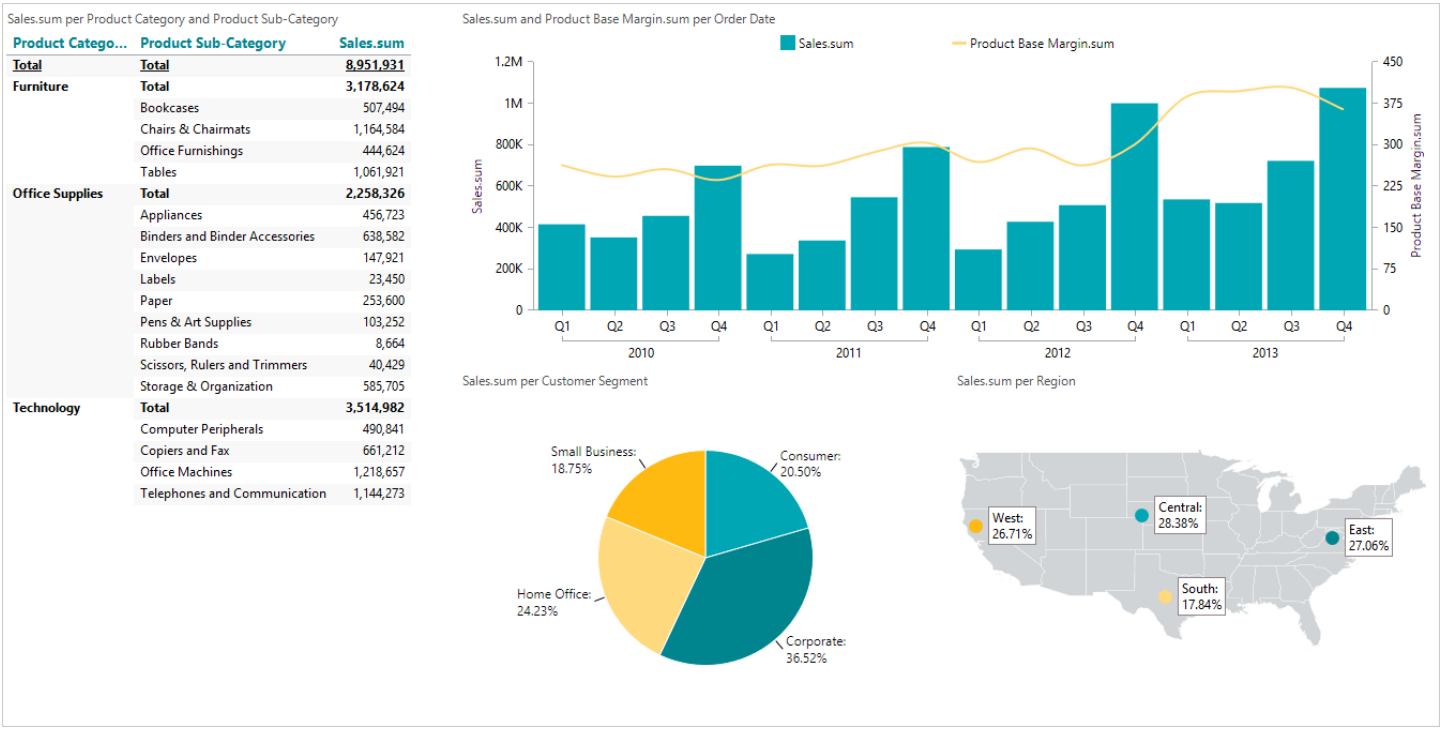
In this exercise we will be adding data from the *SuperStoreSampleData* Excel file to Data Discovery.

Task 1 – SuperStore data

Source Data file location: C:\TU\DataDiscovery\SuperStore.

Add the SuperStoreSampleData Excel file to Data Discovery.

Create an analysis similar to this:



Product Catego...	Product Sub-Category	Sales.sum
Total	Total	8,951,931
Furniture	Total	3,178,624
	Bookcases	507,494
	Chairs & Chairmats	1,164,584
	Office Furnishings	444,624
	Tables	1,061,921
Office Supplies	Total	2,258,326
	Appliances	456,723
	Binders and Binder Accessories	638,582
	Envelopes	147,921
	Labels	23,450
	Paper	253,600
	Pens & Art Supplies	103,252
	Rubber Bands	8,664
	Scissors, Rulers and Trimmers	40,429
	Storage & Organization	585,705
Technology	Total	3,514,982
	Computer Peripherals	490,841
	Copiers and Fax	661,212
	Office Machines	1,218,657
	Telephones and Communication	1,144,273

Save the analysis as: **Lesson 1 Super Store**

Task 2 – Questions:

Focus on **2013** and Customer Segment = **Home Office**. Which Product Category / Sub-category has the highest Sales.sum?

Answer: _____

Further narrow down your focus to this particular Product Sub-category and Region = **East.** Which month has the highest Product Base Margin.sum?

Answer: _____

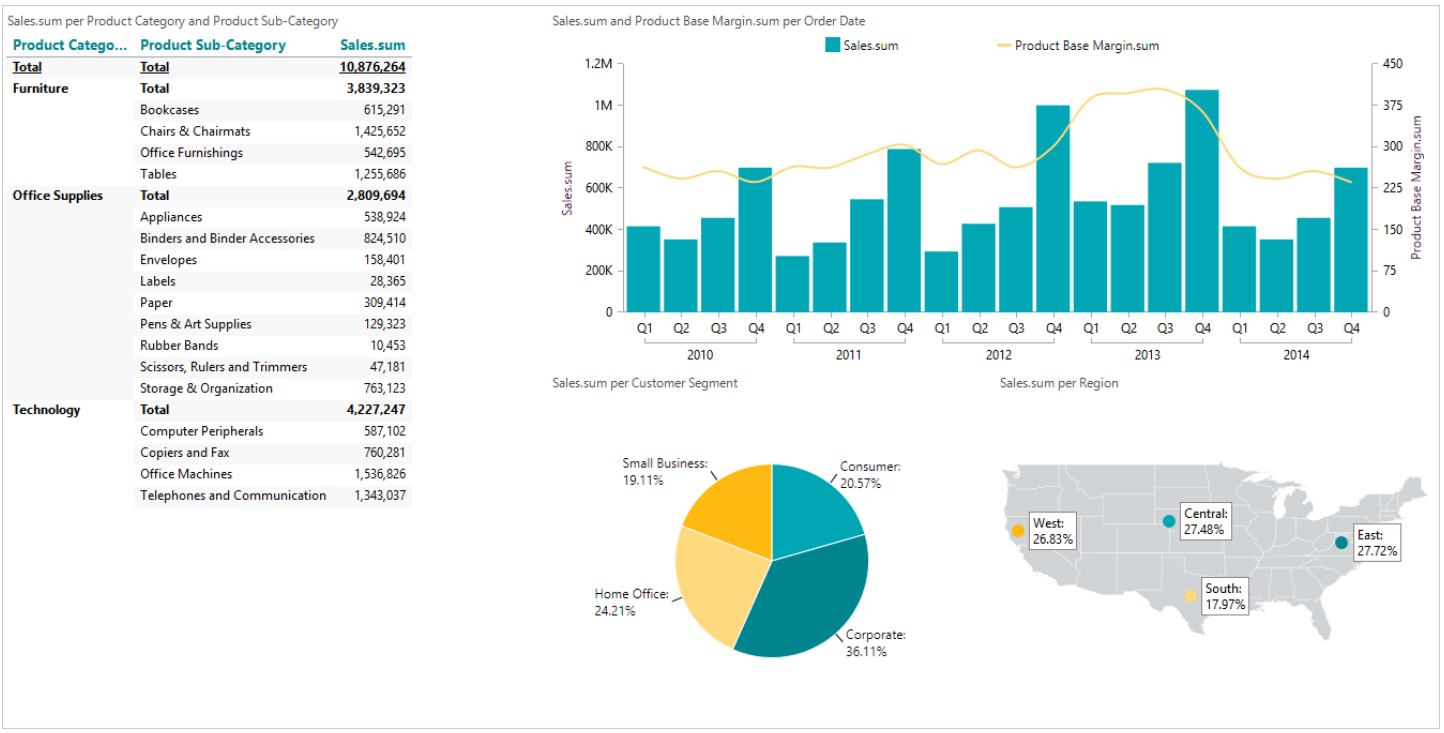
Task 3 – Update SuperStore data

As seen in the previous analysis, the original SuperStoreSampleData Excel sheet contains data for the period 2010-2013.

In the file folder containing the SuperStoreSampleData Excel file there is a sub folder called 'PlusOneYear'. In this sub folder there is an updated version of the SuperStoreSampleData Excel file which resembles an updated version of the original SuperStoreSampleData file, with one additional year (2014) of data.

- Add the updated SuperStoreSampleData Excel file to Data Discovery.

When you refresh your analysis, it will now look like this, with the one year of additional data:



Product Catego...	Product Sub-Category	Sales.sum
Total	**Total**	**10,876,264**
Furniture	**Total**	**3,839,323**
	Bookcases	615,291
	Chairs & Chairmats	1,425,652
	Office Furnishings	542,695
	Tables	1,255,686
Office Supplies	**Total**	**2,809,694**
	Appliances	538,924
	Binders and Binder Accessories	824,510
	Envelopes	158,401
	Labels	28,365
	Paper	309,414
	Pens & Art Supplies	129,323
	Rubber Bands	10,453
	Scissors, Rulers and Trimmers	47,181
	Storage & Organization	763,123
Technology	**Total**	**4,227,247**
	Computer Peripherals	587,102
	Copiers and Fax	760,281
	Office Machines	1,536,826
	Telephones and Communication	1,343,037

Lesson 2: Fine-tuning Source Data Detection

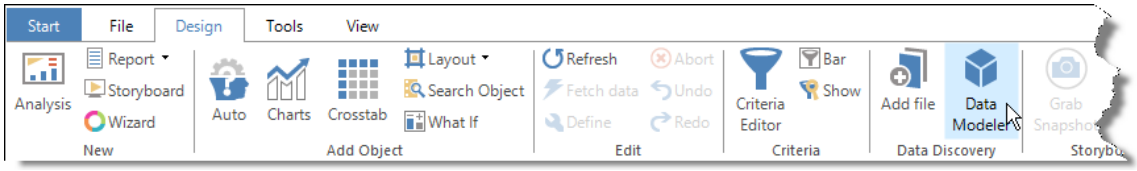
The Data Discovery tool will automatically detect data types of the individual columns in the source file. Only columns containing numeric data will be treated as Measures, while all other columns will be treated as dimensions. Furthermore, date columns will be detected and treated specifically as time dimensions.

However, sometimes you will want to fine-tune this detection and also how and if a specific column should be presented in the TARGIT client.

Your options for fine-tuning and displaying data are:

- Change the column data type, e.g. to read an ID as a dimension instead of as a measure.
- Change a measure to be shown only as a measure. Change a dimension to be shown only as a dimension. Change a measure to be shown as just a single aggregation type.
- Hide irrelevant columns from the source data.

To work with these options, you must first start the Data Modeler from the Design ribbon:



Changing Column Data Types

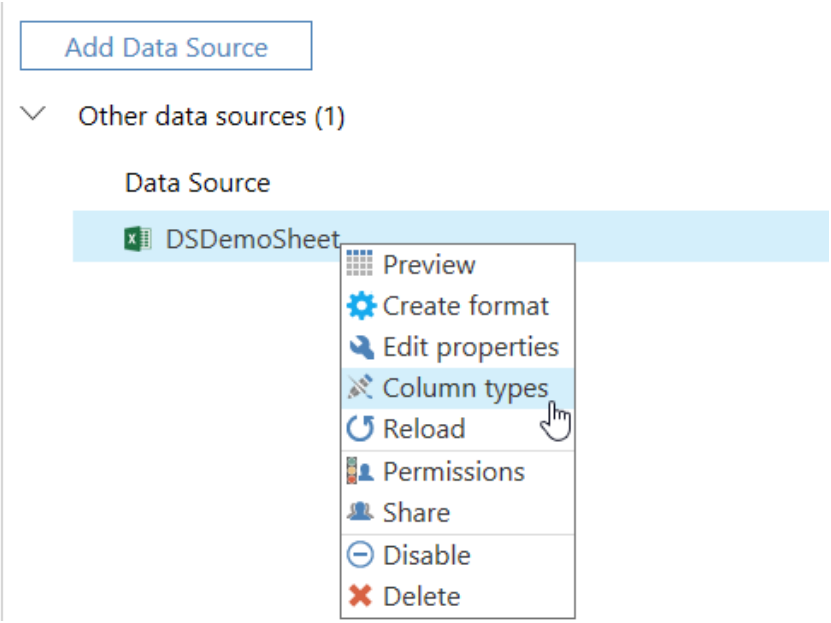
When Data Discovery reads a source file it will automatically detect data types of the source columns.

This will subsequently be used to determine whether a column is source for a measure or for a dimension. As a rule, columns of numeric data types will become measures.

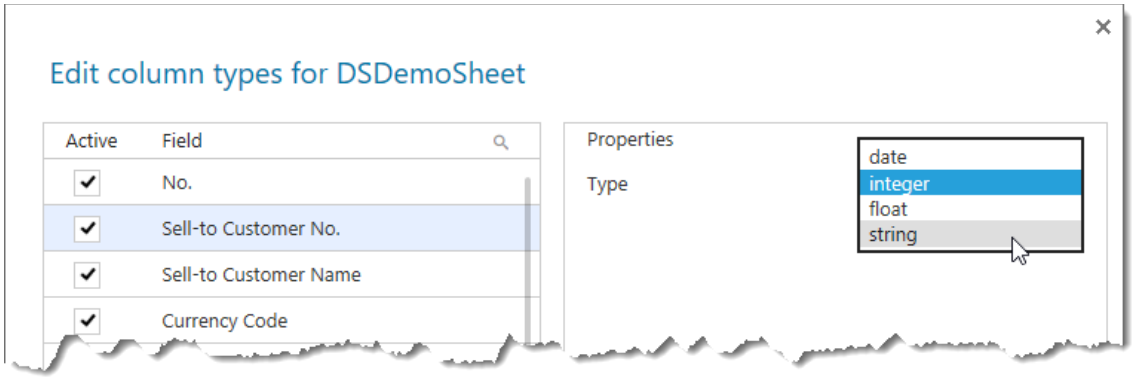
However, very often ID columns, such as Customer IDs and Product IDs will be misinterpreted as measures.

To prevent this, these columns can be read as strings instead of numeric columns.

Go to the Data Sources tab of the Data Discovery tool, and then right click any data source to work with column types.



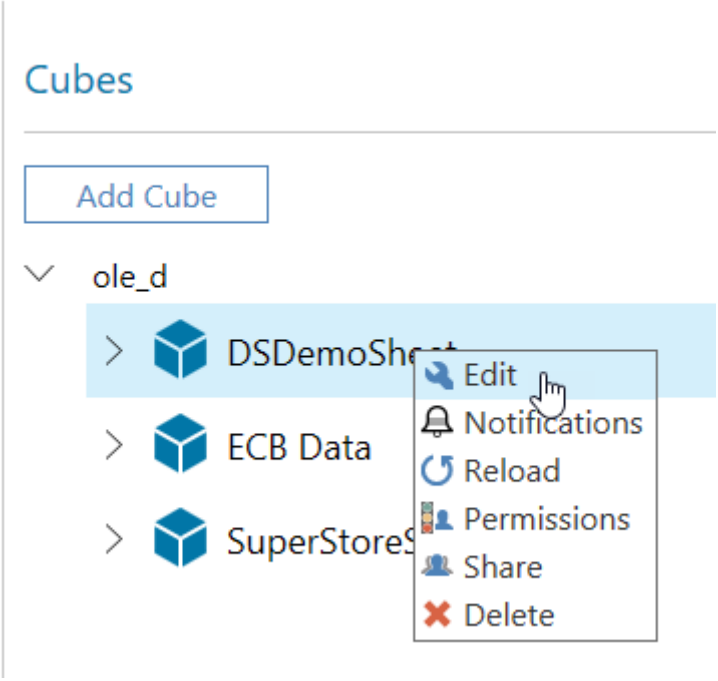
In this file, e.g. the *Sell-to Customer No.* column represents a column that is in fact just a customer ID. If it is kept as an *integer* data type, it will be treated as a measure. So we change it to a *string* to get it treated as a dimension.



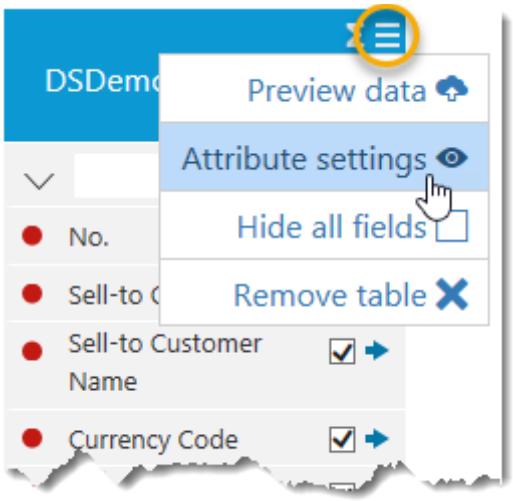
Also, notice that each column may be totally de-activated if they are not to be used in subsequent cubes.

Changing Aggregation Type

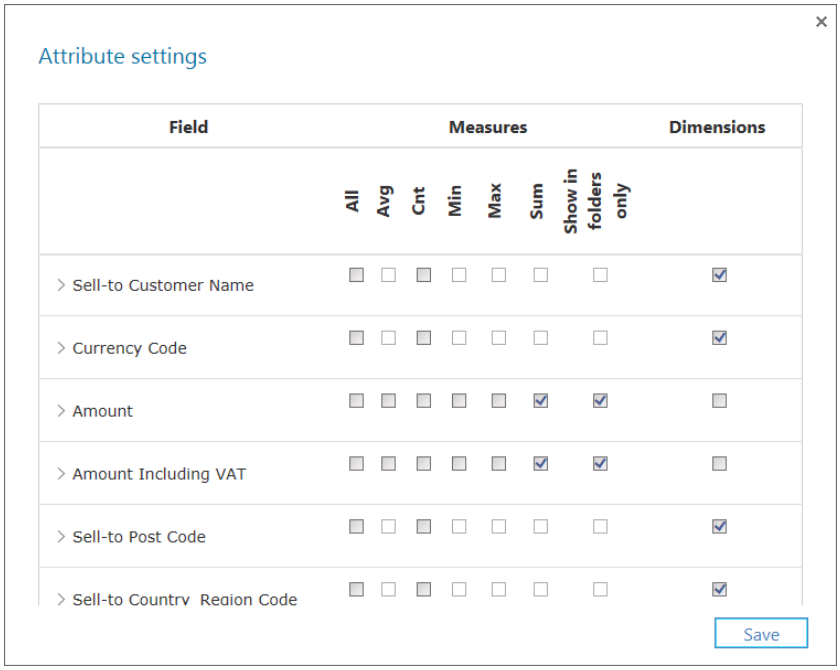
By default, Data Discovery will produce each measure with an abundance of aggregation types – sum, avg, cnt etc. This is just to ensure that all options are available to the end-user, as Data Discovery is not able to make a qualified decision on e.g. one proper aggregation type. You may however lessen the options by making this qualification yourself.



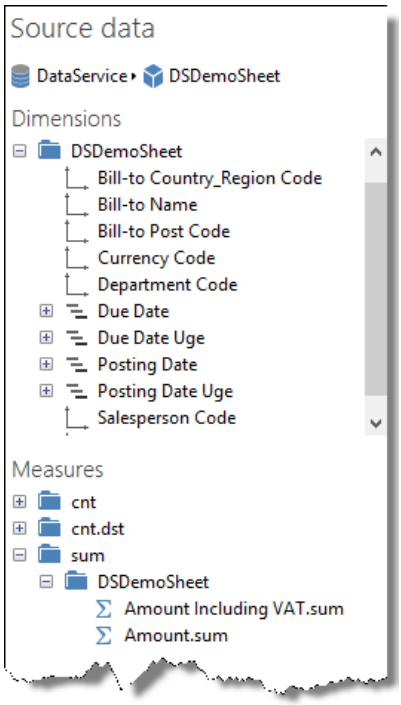
First, edit the cube and then work with *Attribute settings* options on the fact table.



In the example below, only the two Amount fields will become measures, and only as *Sum* measures. All the other fields have been marked as *Dimensions only* fields.



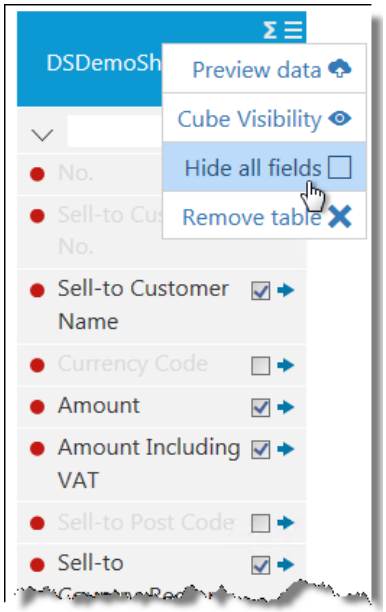
In the TARGIT client's Source Data tab, this will likewise produce a less confusing picture:



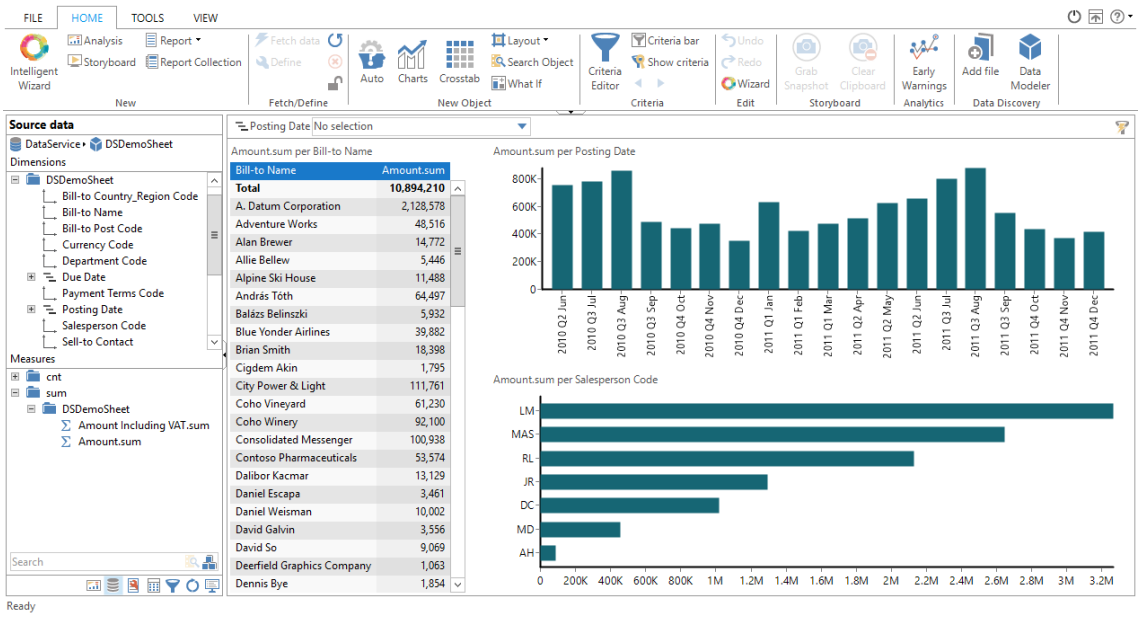
Showing / Hiding fields in the cube

A cube that is based on one or more data sources may still contain fields that are unnecessary for the end-user to see. These fields may then be hidden.

The easiest way is very often to hide everything, and then just unhide the columns you would like to be available to the end-users.



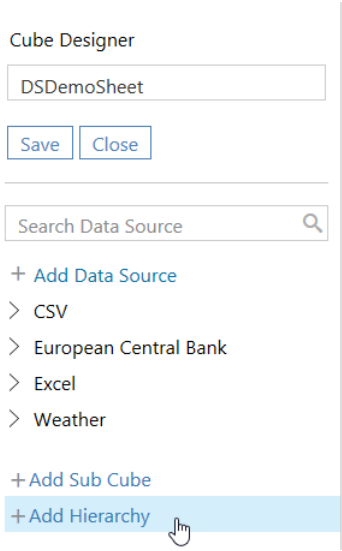
After having saved our changes to the cube, we can open the saved analysis from lesson 1 and see the changes in the source data.



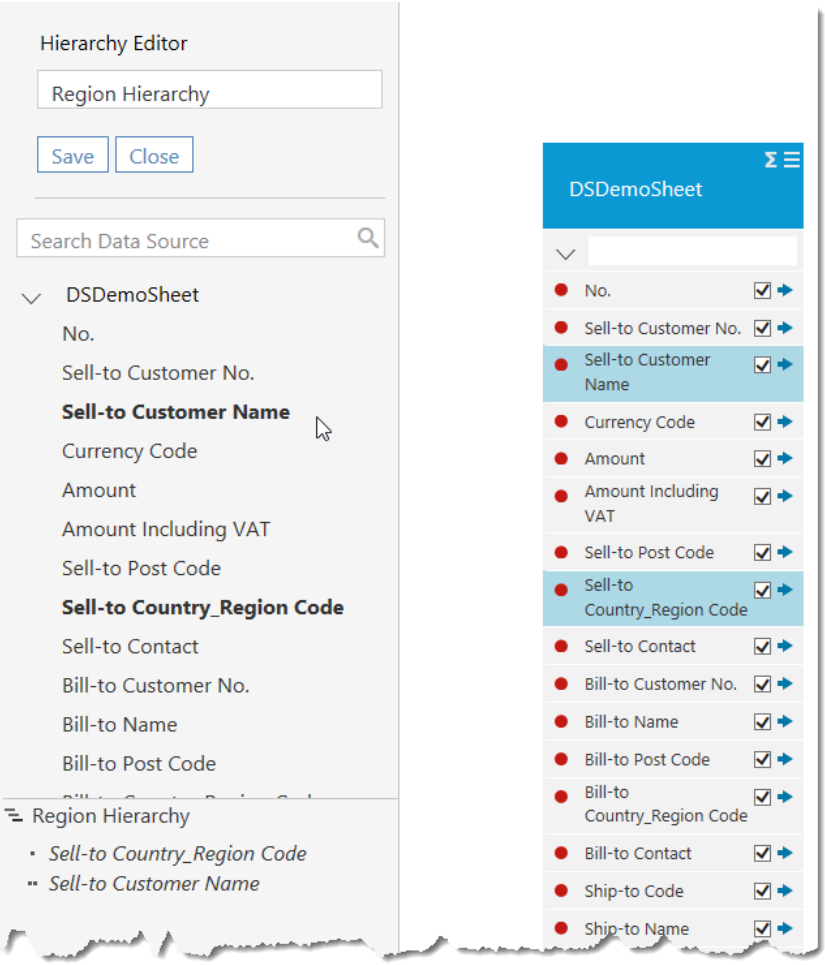
Creating a hierarchy

You can create custom hierarchies from one or more attributes in your cube.

First, edit your cube to get to the 'Add hierarchy' option:

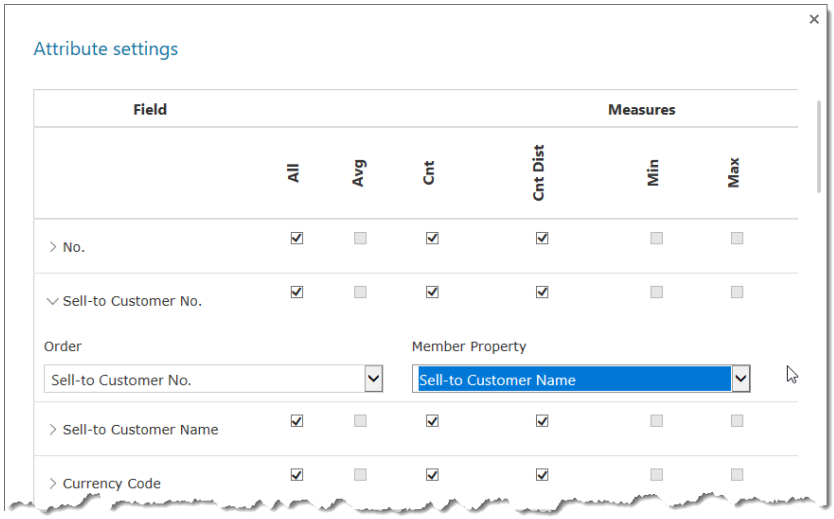


You add levels to the hierarchy, simply by clicking the fields in the left hand side panel:



‘Order’ and ‘Member properties’

With attribute settings you can also define sorting keys – e.g. if you want to display weekday names (Monday, Tuesday, Wednesday etc.), then you need to sort it by weekday numbers (1, 2, 3 etc.).



Member properties are useful when you one-to-one relations between related dimensions attributes. E.g. ‘Vendor number’ and ‘Vendor name’ may be two separate dimension attributes, but by setting up one as a member property to the other, you can get better performing queries – and at the same prevent the client from displaying unnecessary subtotals.

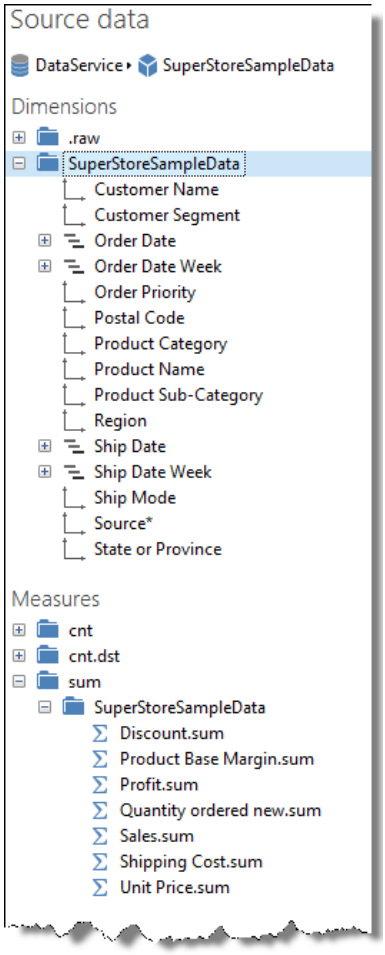
Exercises Lesson 2

Task 1 – fine tuning SuperStore data presentation

Edit the SuperStoreSampleData cube you created in lesson 1.

- Hide the data that is not interesting in an analytical context, i.e. ROW ID, CUSTOMER ID, PRODUCT Container and LOCATION 1.
- Make sure that measures are not used as dimensions.
- Make sure that POSTAL CODE is not used as a measure.

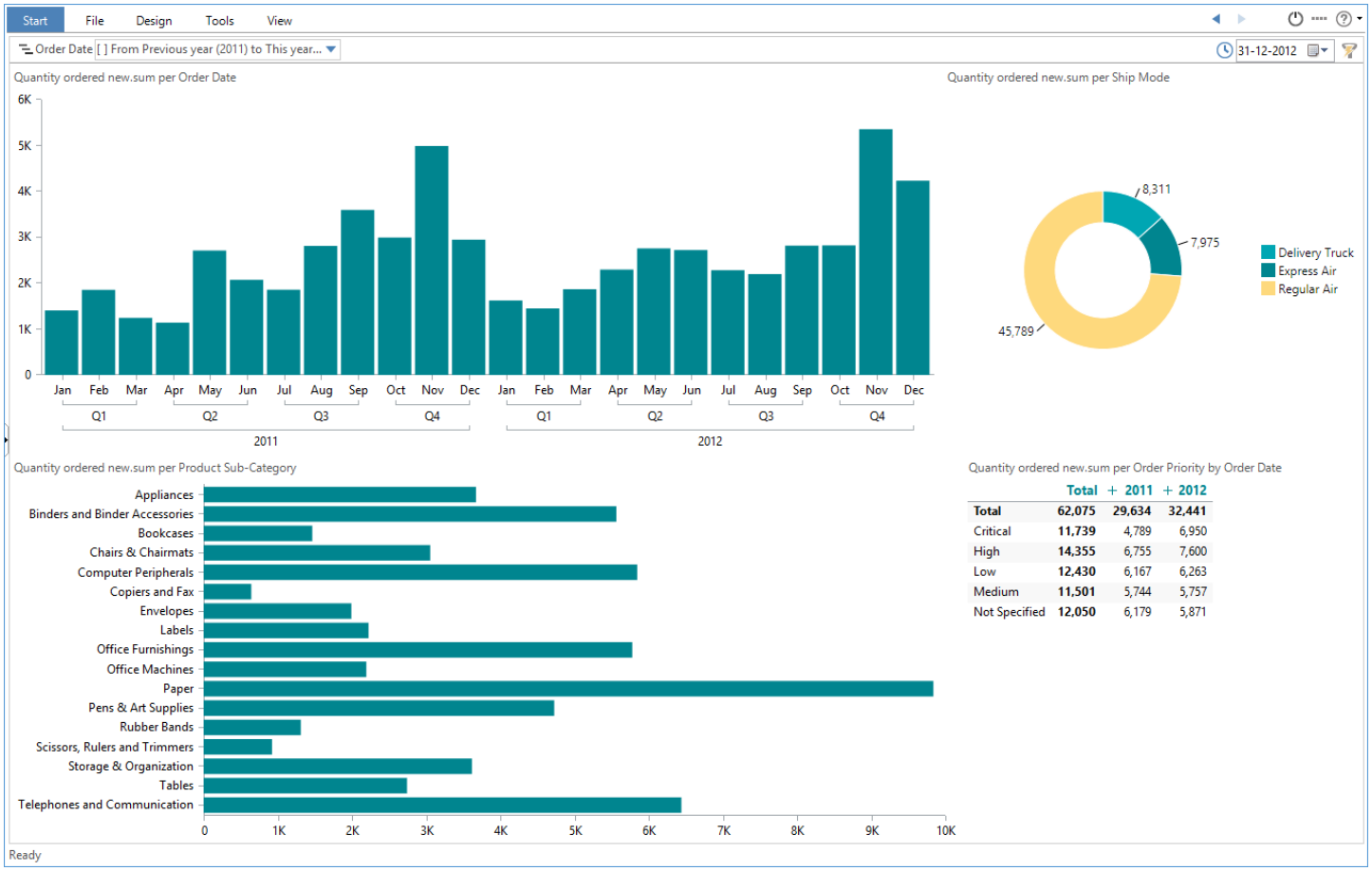
Your Source data tab should look like this:



Task 2 – create analysis with fine-tuned data

Create an analysis that matches the screenshot.

- Set dynamic date origin to 31st December 2012.
- Set Dynamic criteria on Order date to *Between Previous year and This year.*



Save the analysis as Lesson 2 Quantity Ordered New Analysis.

Task 3 – Questions:

Which month in 2011 had the highest Quantity Ordered New for Ship Mode *Express Air*.

Answer: _____

Further, narrow down your focus to this particular Month, which Product Sub-category has the highest Quantity Ordered New?

Answer: _____

For this Product Sub-category, including all previous criteria, how is the split on Order Priorities?

Answer: _____

Lesson 3: Data mashup

So far, we have created very simple cubes with very simple data models. Actually, every single cube so far has been based on a single source file.

TARGIT Data Discovery may of course also work with multiple, related data sources. The data sources do not need to come from the same source or to be of the same type or same format – in fact, as long as TARGIT Data Discovery is able to read the data, these data can be mashed up to fulfill any analytical needs.

To perform a successful data mashup, you will still need to be able to relate data from different to each other. This relation requires common keys across the data sources to be mashed up.

Looking at the DSDemoSheet cube that was created previously, we notice that some of the dimensions, e.g. the Salesperson Code dimension only contain the salespersons' initials.

Amount.sum per Salesperson Code

Salesperson Code	Amount.sum
Total	**10,844,210**
LM	3,264,366
MAS	2,645,487
RL	2,128,578
JR	1,294,330
DC	1,016,650
MD	407,617
AH	87,182

Suppose that we actually had information about these salespersons' real names, but in a separate data source. Then, by mashing up these new data with data from the original DSDemoSheet data source, we will be able to enrich the Salesperson dimension.

Adding a new Data Source

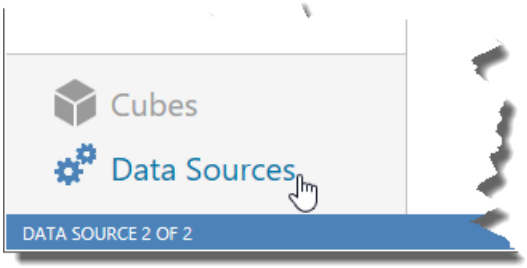
The additional information with Salesperson names is located in a separate Excel sheet in the same folder as the DSDemoSheet file:

A1 | SalespersonCode

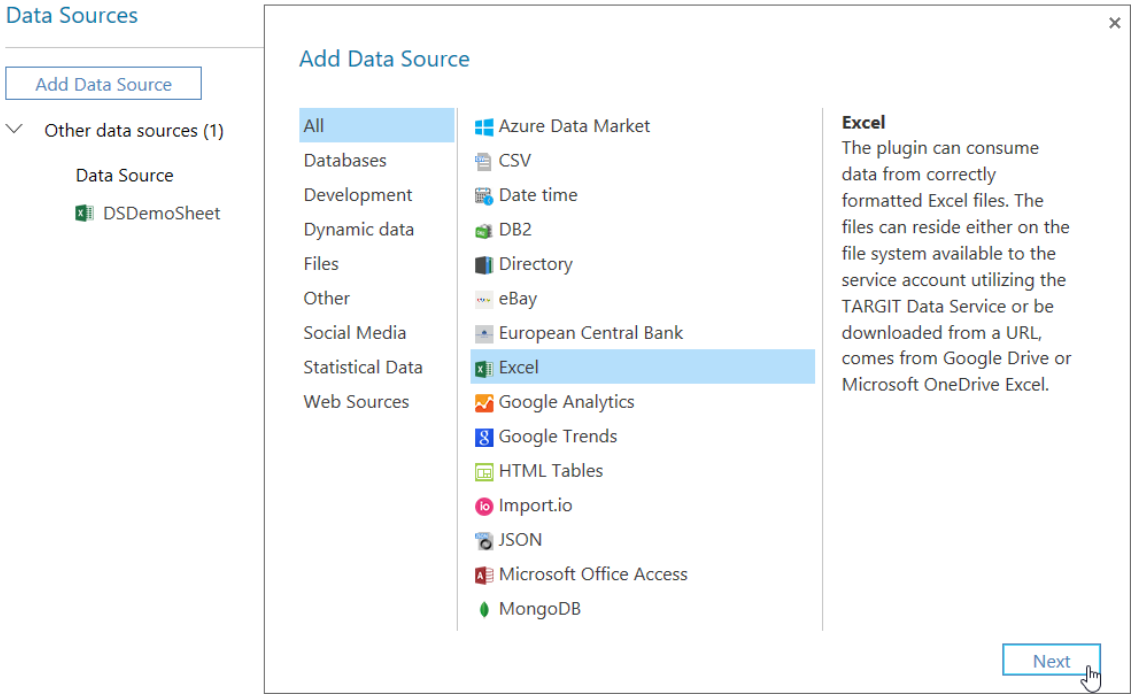
	A	B	C	D
1	SalespersonCode	SalespersonName		
2	AH	Adriana Hewitt		
3	DC	Douglass Cobb		
4	JR	Jeremy Robson		
5	LM	Lucas Michelson		
6	MAS	Margarita Aphrodite Silva		
7	MD	Minna Dunno		
8	RL	Richard Lobster		
9				

First, we will need to add this extra Excel sheet as a new Data Source to TARGIT Data Discovery.

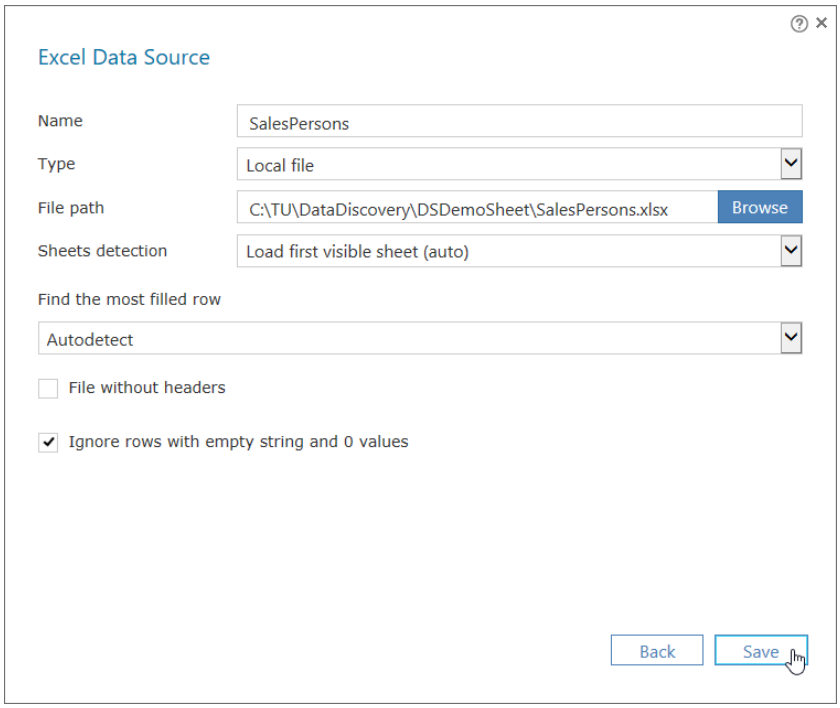
Open the Data Modeler, and go to the Data Sources tab:



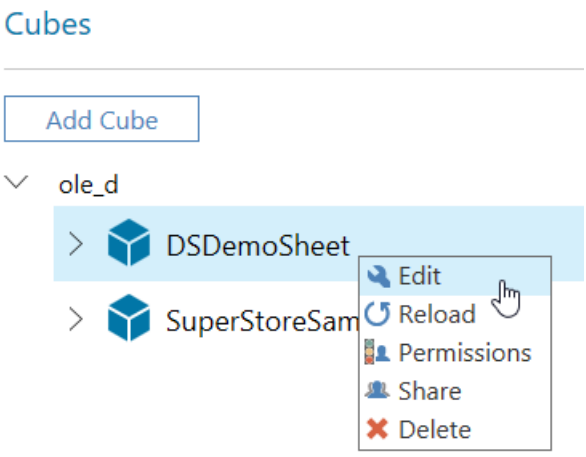
Click the “Add Data Source” button and select an Excel type data source:



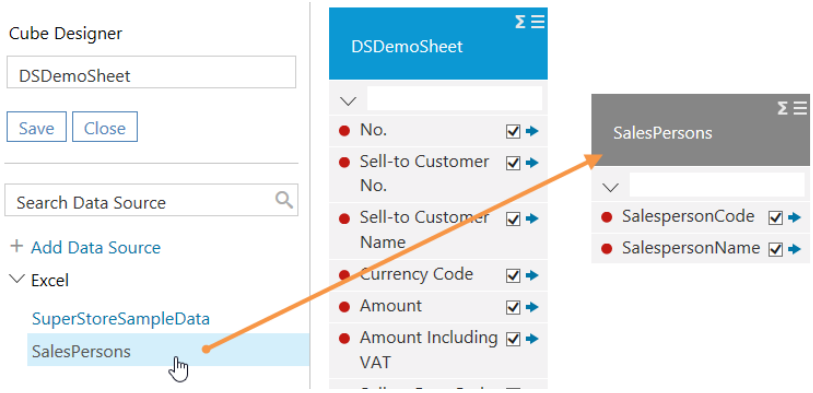
Set up properties for loading data from the new Excel data source:



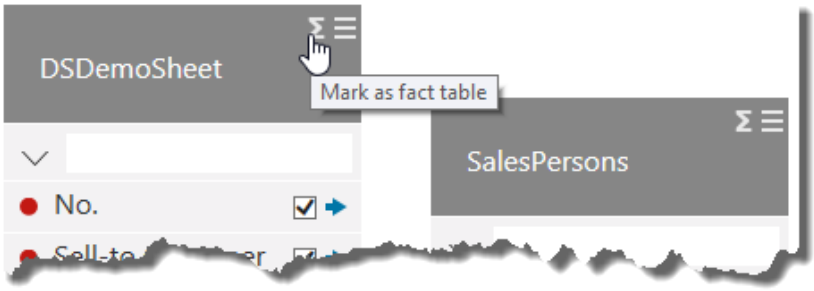
Now we will need to open the Cube Designer, either as a new cube or for an existing cube. In this case we will edit the existing DSDemoSheet cube:



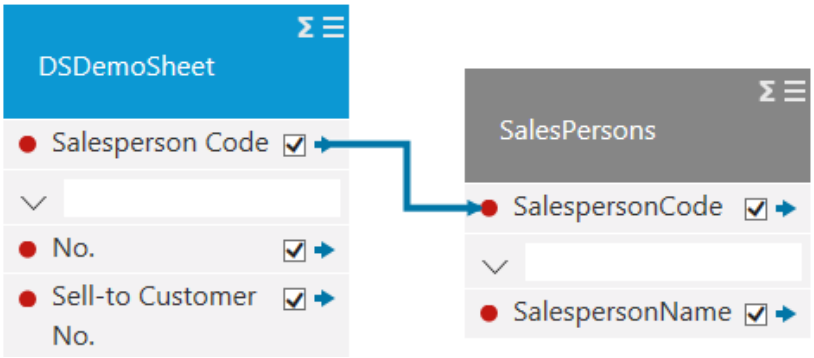
On the left hand side of the Cube Designer, the available Data Sources are arranged by type. Add the new Excel SalesPersons data source to the model by dragging in the table.



Notice the different colors of the two tables. The blue color indicates that the DSDemoSheet table has been marked as a *fact* table. All tables containing measures should be marked as fact tables. Clicking the Sigma sign on top of each table will toggle between fact table and dimension table.

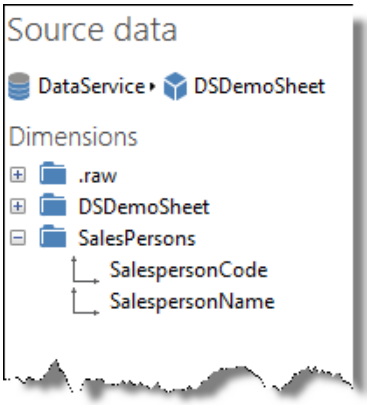


Finally, we will need to set up the correct relation between those two tables. The tables are related by the Salesperson Code. Create the relation by dragging from the blue arrow in the fact table to the red dot in the dimension table.



At this point, the cube is ready to be saved. If it is a new cube, you will need to provide a cube name before you can save it.

Notice that Data Discovery automatically creates a Display Folder for each data source in your cube:



Now it should be possible to create analyses with enriched information about Salespersons' names:

Amount.sum per SalespersonCode and SalespersonName

SalespersonCode	SalespersonName	Amount.sum
Total	**Total**	**10,844,210**
AH	Adriana Hewitt	87,182
DC	Douglass Cobb	1,016,650
JR	Jeremy Robson	1,294,330
LM	Lucas Michelson	3,264,366
MAS	Margarita Aphrodite Silva	2,645,487
MD	Minna Dunno	407,617
RL	Richard Lobster	2,128,578

Exercises Lesson 3

Task 1 – Absence data

Source Data file location: C:\TU\DataDiscovery\Absence.

Suppose we just bought a new HR system and want to start analyzing the data, but we are not quite sure what our possibilities are.

We extracted some data from the system and saved it in two Excel files and we are now ready to build a simple data model based on these files.

We have an Excel file with employee information:

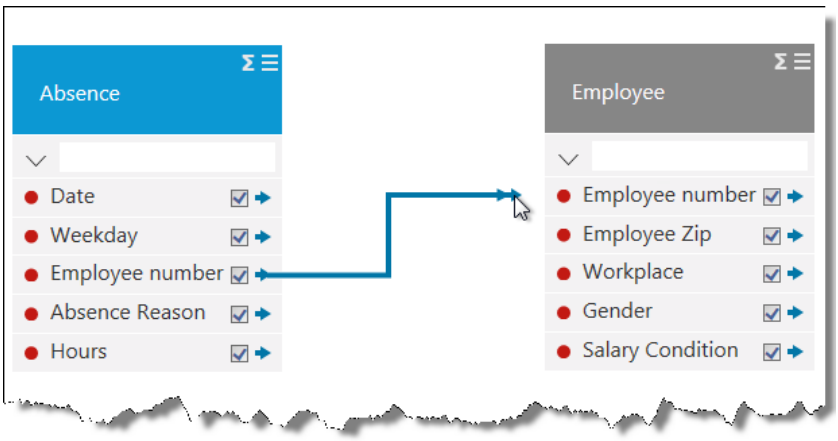
	A	B	C	D	E
1	Employee number	Employee Zip	Workplace	Gender	Salary Condition
2	0101010001	N19 3AE	St Mary's Pre-School	Male	Fixed monthly, monthly in advance
3	0101381382	N19 3AP	CC's Nursery	Female	Hourly, monthly in arrears
4	0101451577	E11 1AH	Tenderlinks Day Nursery	Male	Fixed monthly, monthly in arrears
5	0101462358	E11 1AY	Woodlands Pre-School	Female	Fixed monthly, monthly in arrears
6	0101481603	WC1E 6HG	Hill Park Day Nursery	Male	Hourly, monthly in arrears
7	0101490904	SE24 9AY	Nursery on the Green	Female	Fixed monthly, monthly in advance
8	0101491188	E6 4NT	Hill Day Pre-School	Female	Fixed monthly, monthly in arrears
9	0101500942	E11 1BJ	Chapel House Pre-School	Female	Fixed monthly, monthly in arrears
10	0101502406	N20 9BS	Oak Tree Day Nursery	Female	Fixed monthly, monthly in arrears
11	0101559009	N12 8RP	The Little Academy	Male	Fixed monthly, monthly in arrears

… and an excel file with some absence information:

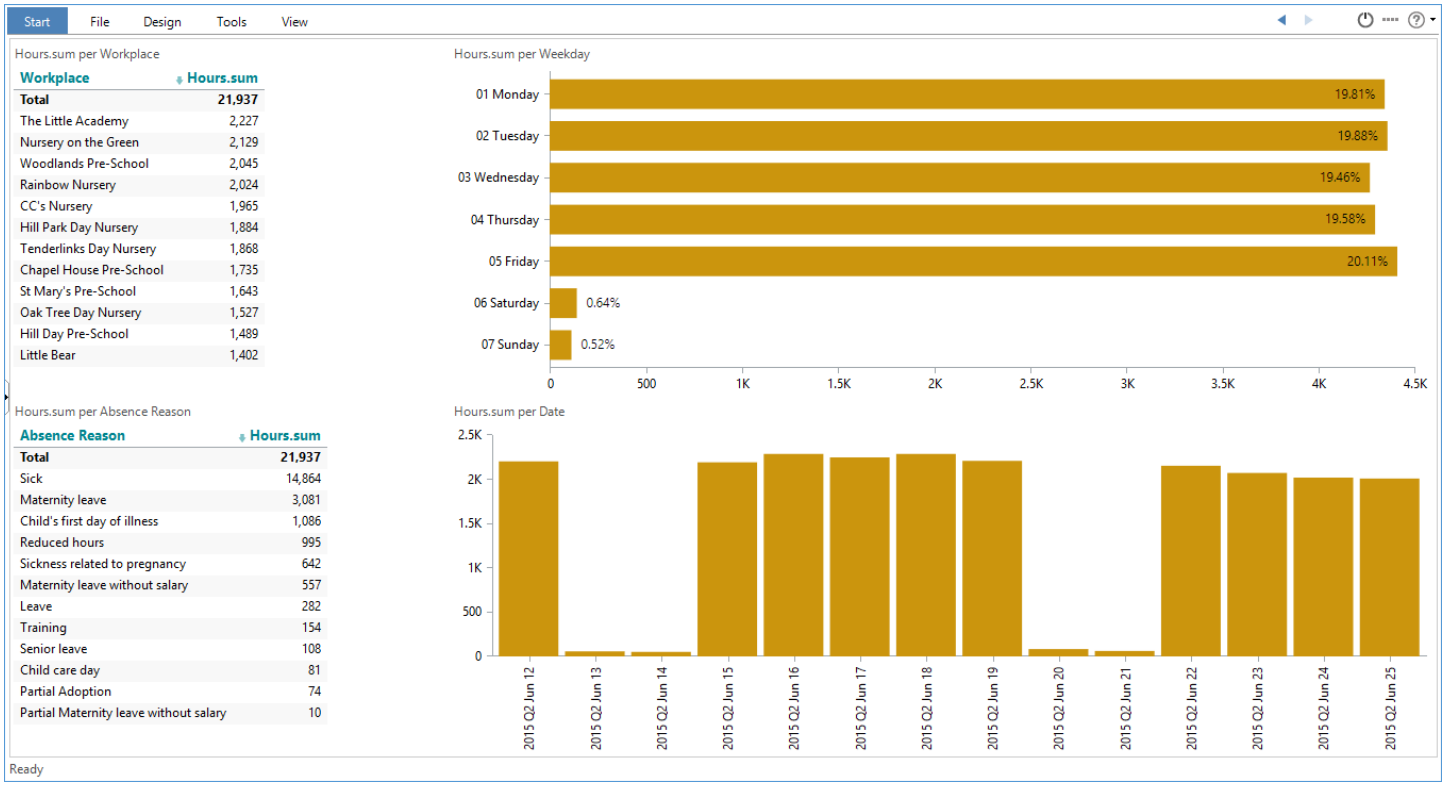
	A	B	C	D	E
1	Date	Weekday	Employee number	Absence Reason	Hours
2	6/16/2015	02 Tuesday	2102691124	Reduced hours	14.80
3	6/16/2015	02 Tuesday	1506661958	Sick	9.71
4	6/16/2015	02 Tuesday	2505801752	Sick	9.33
5	6/16/2015	02 Tuesday	2607700578	Sick	8.63
6	6/16/2015	02 Tuesday	0307690888	Sick	8.50
7	6/16/2015	02 Tuesday	2803511554	Maternity leave	8.46
8	6/16/2015	02 Tuesday	0406651932	Maternity leave	8.00
9	6/16/2015	02 Tuesday	0901541248	Sick	8.00
10	6/16/2015	02 Tuesday	1605562120	Sick	8.00
11	6/16/2015	02 Tuesday	2401771606	Sick	8.00
12	6/16/2015	02 Tuesday	0310721360	Sick	7.99

Add these two Excel files as Data Sources to Data Discovery and create a data model and a cube in order to start analyzing on these data.

The two data sets are related in this way:



Create an analysis to match this screen shot:



Save the analysis as Lesson 3 Absence Analysis

Questions

What weekday is the most common weekday for having *Child's first day of illness* absence?

Answer: _____, in percent: _____

On this weekday, which Workplace tops the list for *Child's first day of illness* absence?

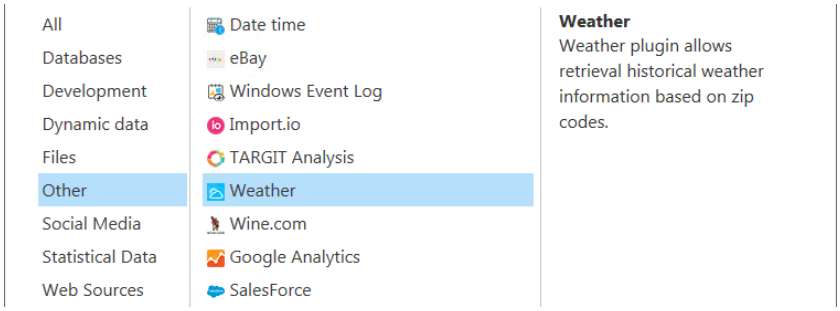
Answer: _____

Task 2 – Weather

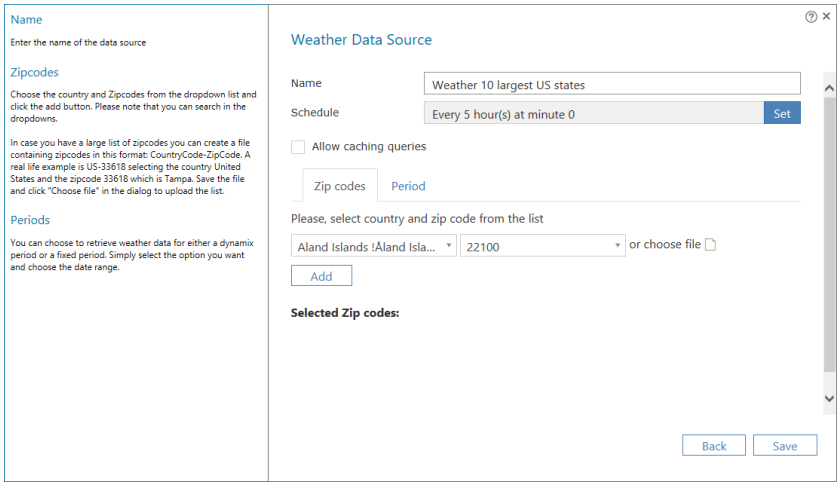
Source Data file location: C:\TU\DataDiscovery\Weather.

Remember how the weather was much better in the good old days? The summers were sunnier, the winters snowier, or maybe our memories are clouded.

In this task, we will look into how we can pull weather information using one of the online web services plug-ins:

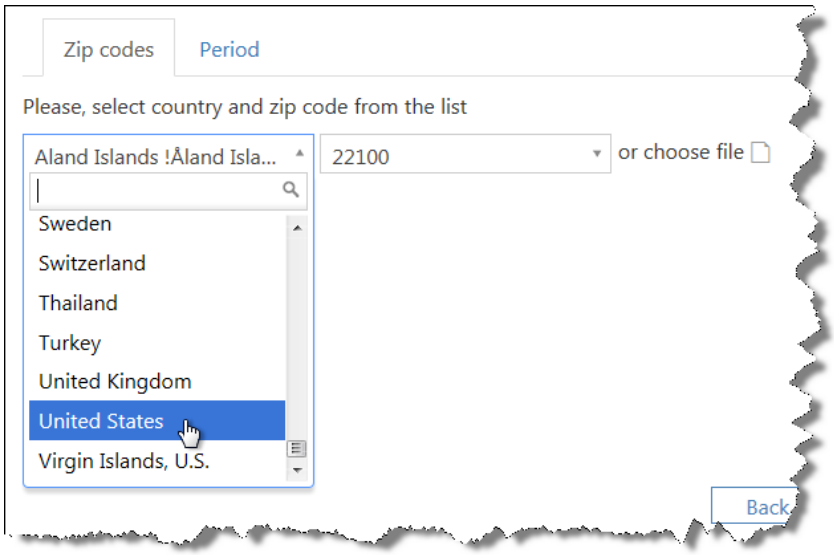


This dialog box lets you specify your query. We have clicked the question mark that opens the help panel in the left side.

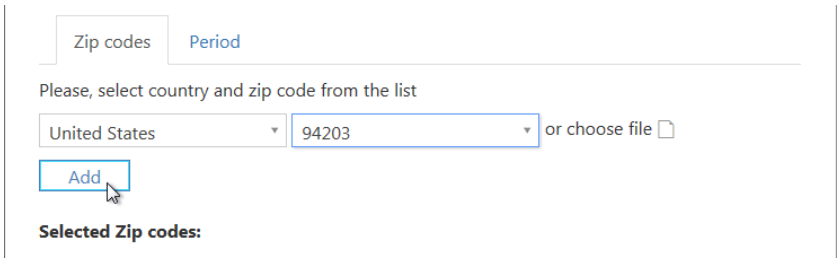


We want to analyze the weather in the 10 largest states in the US by population, hence the name.

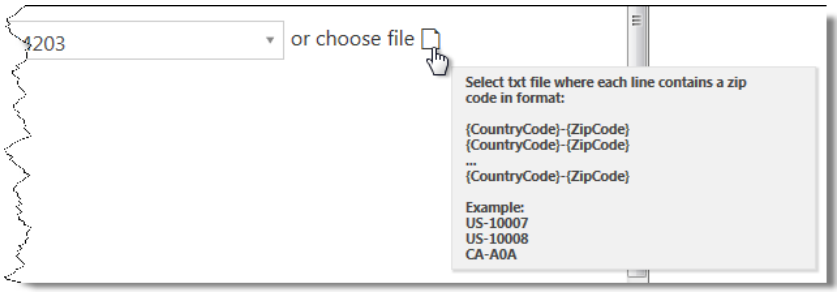
We can choose country from the drop-down list and manually enter the zip code:



Then it is just a matter of clicking add and searching for the next zip code.

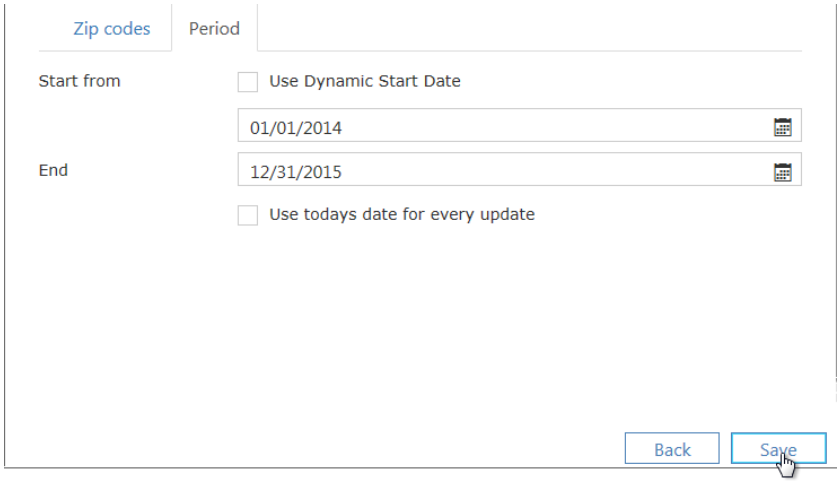


An even easier way to add zip codes would be to compile a list of codes in a file and use the choose file option. It even gives you information about the correct format when you mouse over the option. Click the button and navigate to the file location.

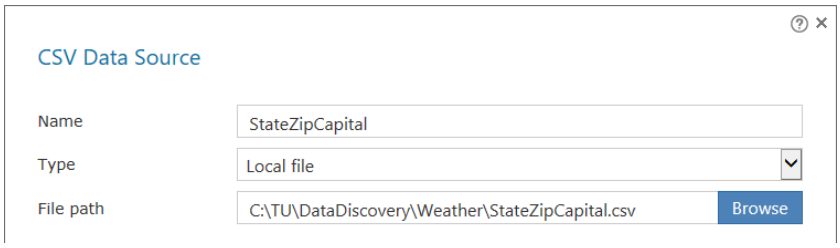


The list of zip codes can now be seen in the **Selected zip codes:** list

For this exercise we limit the data to a period of two years before we click save:



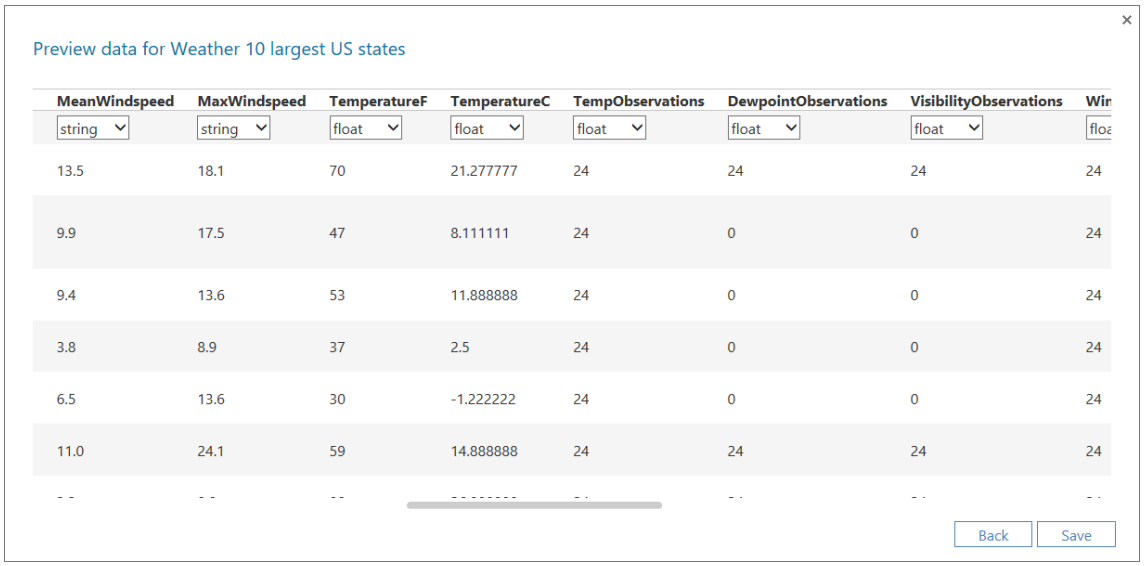
Now the data source is created and will be ready when the status changes from “Creating” to “Available” The next data source is a small CSV file with information about state, zip code and state capital.



This small table will be used as dimension information for categorizing and filtering states.

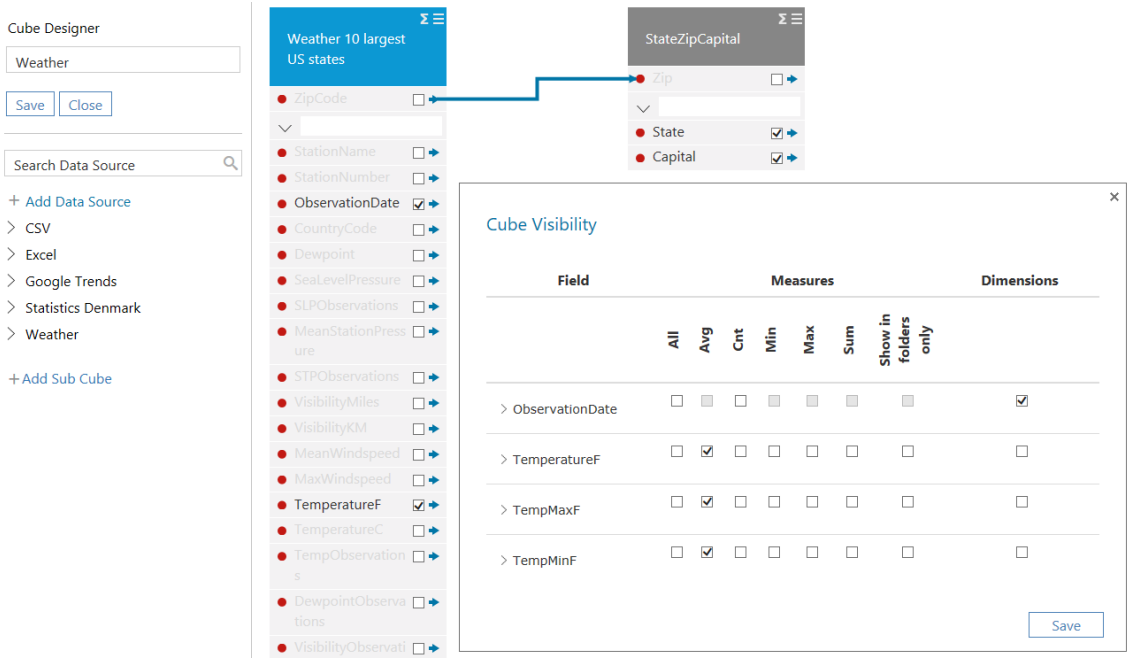
	A	B	C	D
1	State	Zip	Capital	
2	California	94203	Sacramento	
3	Florida	32301	Tallahassee	
4	Georgia	30301	Atlanta	
5	Illinois	62701	Springfield	
6	Michigan	48901	Lansing	
7	New York	12201	Albany	
8	North Car	27601	Raleigh	
9	Ohio	43201	Columbus	
10	Pennsylva	17101	Harrisburg	
11	Texas	73301	Austin	

If we preview the Weather data source, we can see that we get a large amount of information including temperature information:



MeanWindspeed	MaxWindspeed	TemperatureF	TemperatureC	TempObservations	DewpointObservations	VisibilityObservations	Wi
string	string	float	float	float	float	float	flo
13.5	18.1	70	21.277777	24	24	24	24
9.9	17.5	47	8.111111	24	0	0	24
9.4	13.6	53	11.888888	24	0	0	24
3.8	8.9	37	2.5	24	0	0	24
6.5	13.6	30	-1.222222	24	0	0	24
11.0	24.1	59	14.888888	24	24	24	24

We are ready to build the cube and we use the weather data source as our primary table and relate the zip code from this file to the small CSV file with the state and capital information.



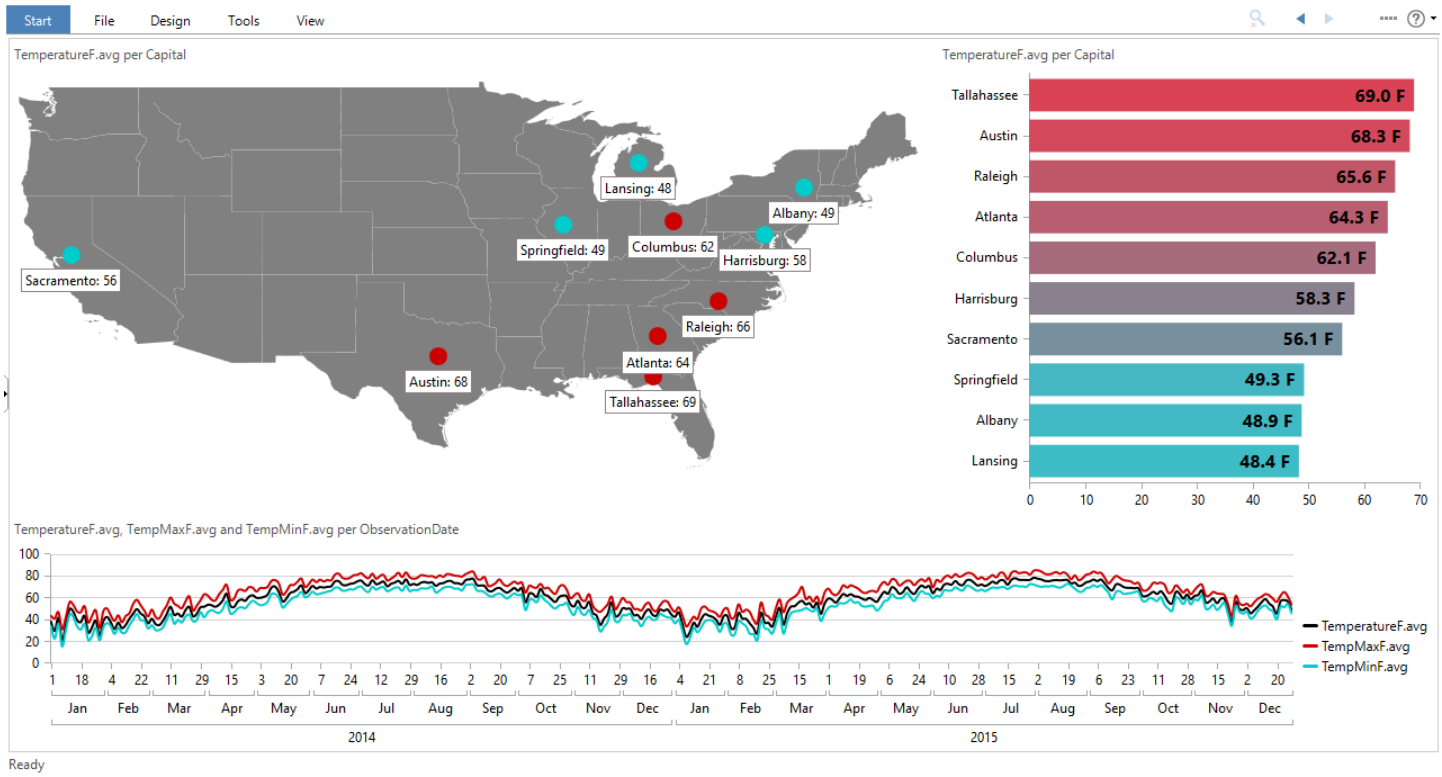
The process of finding the desired level of information from this big volume of data can be quite difficult, but you can check and uncheck the various fields from both tables look at the cube and return to the cube designer.

Question

Why is it important to use the Avg aggregation for the temperature measures? Why not use the Sum aggregation?

Answer: _____

Based on these Weather data, you should now be able to create an analysis like this:



Task 3 – Absence. Weather related?

Maybe there is a correlation between absence and weather conditions?

Create a new Weather Data Source based on data from United Kingdom, zip codes N12 and E11.

The absence data is from June 2015, which should be taken into consideration when you define the period.

Add these weather data to your absence cube by connecting the Date from the Absence data source to the ObservationDate in the Weather data source.

Maybe – or maybe not – you can find correlations between absence and weather?

Lesson 4: Formats – enriching data

TARGIT Data Discovery comes with an extensive library of functions for extracting, modifying, cleansing and enriching data from your Data Sources.

This is something you may often need, especially when working with data through web services, as sometimes the format of these data may initially be very different from what you require.

Functions Library

If you are familiar with the functions in Excel, it will not take you long to learn the more than 100 available functions in TARGIT Data Discovery, as the two sets of functions are similar or almost identical.

As an example, let us assume that we would like to manipulate the data in the DSDemoSheet data source. Specifically, we would like to cleanse data in the very first column, containing transaction IDs – the No. column. Most of the IDs have been prefixed with "SPI", and now we would like to remove this prefix.

Preview data for DSDemoSheet

No.	Sell-to Customer No.	Sell-to Customer Name	Currency Code	Amou
103019	40000	Deerfield Graphics Company		1063.1
103020	50000	Guildford Water Department		533.4
SPI1001127	149000	András Tóth		50368.7
SPI1001128	102000	Graphic Design Institute		122.1

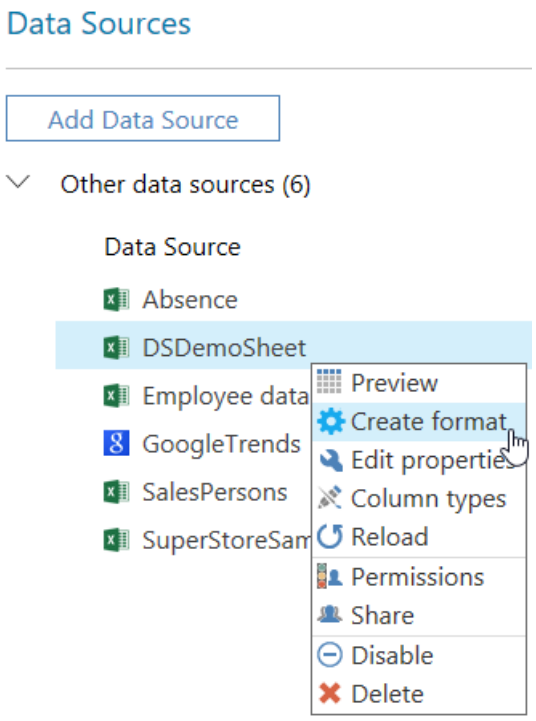
To do this, we will need a number of conditional and text manipulating functions:

- **if**(statement_to_be_tested, value_if_true, value_if_false). Conditional function that can return different results.
- **left**(string, number_of_characters). Returns a substring with number of characters counted from the left.
- **right**(string, number_of_characters). Returns a substring with number of characters counted from the right.
- **len**(string). Returns number of characters of a string.

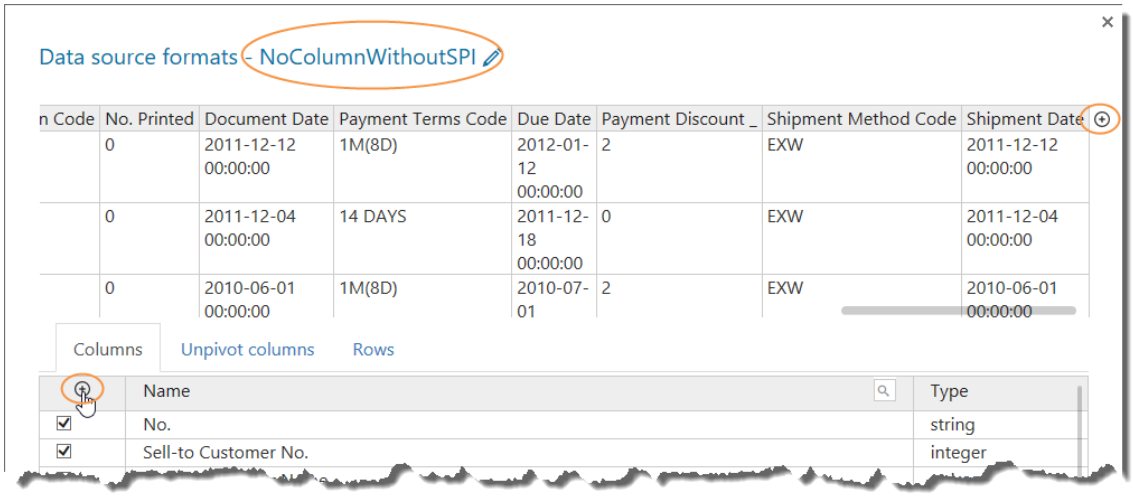
Tip: All functions have been documented on http://doc.targit.com in the Data Service / Formats / Formulas section.

Create a Format

To create a new Format, right click the Data Source you would like to manipulate.

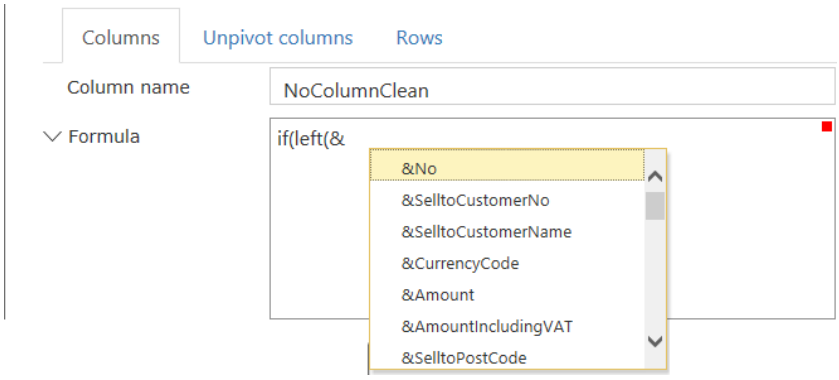


The new Format must be given a proper name. To add a new column to the Format, click one of the '+' buttons.

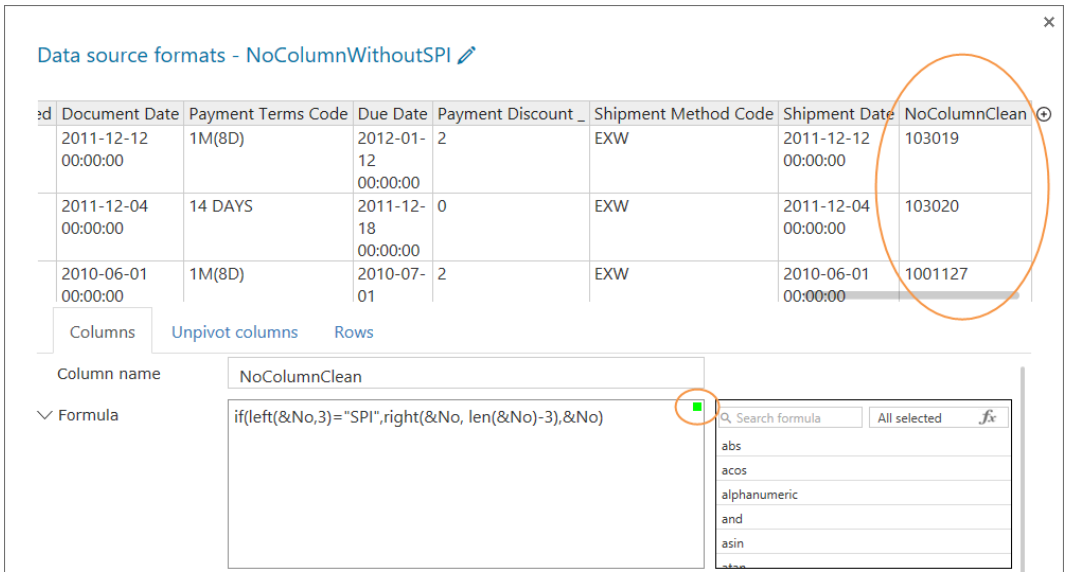


The new column must also be named properly. Expand the Formula field to start entering the text manipulating functions. From the panel on the left hand side, you can search for available functions. And once found, drag them into the formula field.

References to other columns in the Source Data uses the '&' syntax. You can search for the column names in search field on the left hand side, or, by typing '&' and then pressing **CTRL + Space**, you will get a list of all columns and functions matching the partial string you have written.



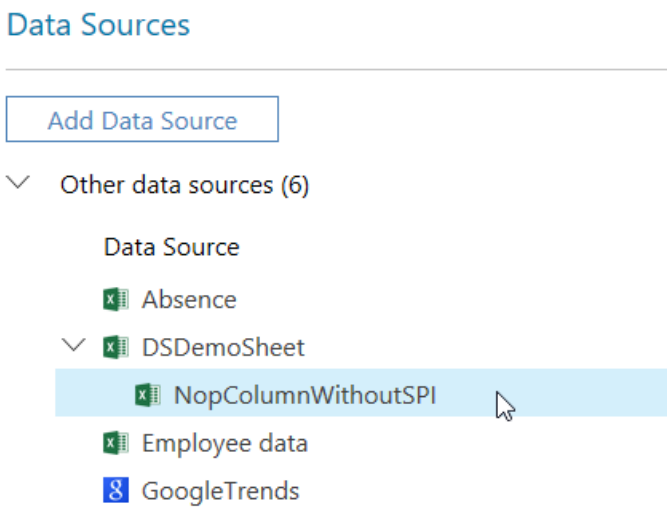
Once you have completed your formula, a small green indicator in top right corner of the formula field indicates that the formula has been correctly syntax checked. Also, after clicking 'Save', the new column should appear in the preview on top.



The formula syntax "translated": *If the first three characters of the No. column equals "SPI" then keep the number of characters, counted from the right, that equals the total length of the string minus 3; if not, then just keep No. as it is.*

So, now we have a new column with transaction ID numbers, where any prefixed “SPI” have been removed.

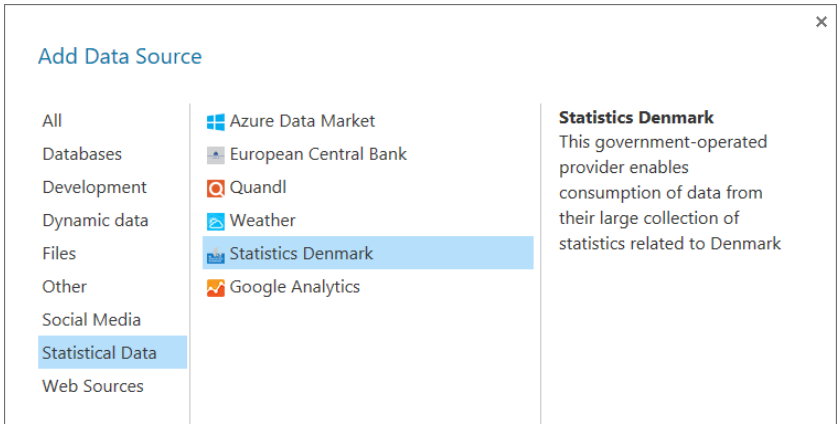
When clicking ‘Save’ a second time, the new Format will be saved. A saved format will be available as any other Data Source, while keeping the original Data Source.



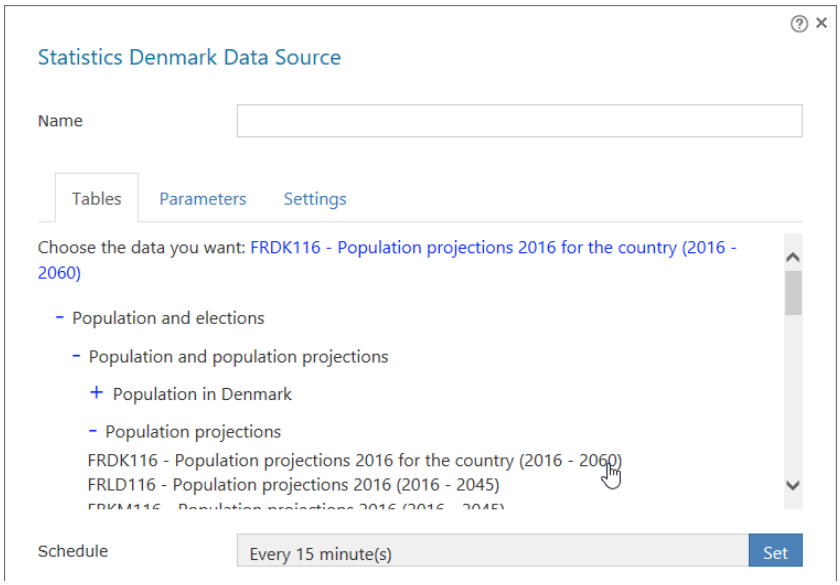
Exercises Lesson 4

Task 1 – Population projection

Statistics Denmark provides a large quantity of free and publicly available statistics data related to Denmark.



Pick the FRDK116 statistic, which is Population projections until 2060.



When you preview the data, notice the ALDER (age) column. It has been made up from a number plus text, where the text is either ‘year’ or ‘years’.

Preview data for PopulationProjectionsDenmark

HERKOMST	KØN	ALDER	TID	INDHOLD
string	string	string	integer	integer
Immigrants from western countries	Women	23 years	2044	2748
Immigrants from western countries	Men	23 years	2044	2538
Immigrants from western countries	Women	24 years	2044	3037
Immigrants from western countries	Men	24 years	2044	2929
Immigrants from western countries	Women	25 years	2044	3167
Immigrants from western countries	Men	25 years	2044	3149

We will translate the columns in a bit, but just for reference:

- Herkomst = Origin
- Køn = Gender
- Alder = Age
- Tid = Time
- Indhold = Content (actually, this is the Population measure)

The combined number and text will create sorting problems when we want to use that column as a dimension. To fix it, we will create a new Format to this data source, and in this Format we will add an extra column with corrected age as seen below. Notice the Destination Type.

Data source formats - CorrectedAge

HERKOMST	KØN	ALDER	TID	INDHOLD	CorrectAge
Immigrants from western countries	Women	23 years	2044	2748	23
Immigrants from western countries	Men	23 years	2044	2538	23
Immigrants from western countries	Women	24 years	2044	3037	24
Immigrants from western countries	Men	24 years	2044	2929	24
Immigrants from western countries	Women	25 years	2044	3167	25
Immigrants from western countries	Men	25 years	2044	3149	25

Columns | Unpivot columns | Rows

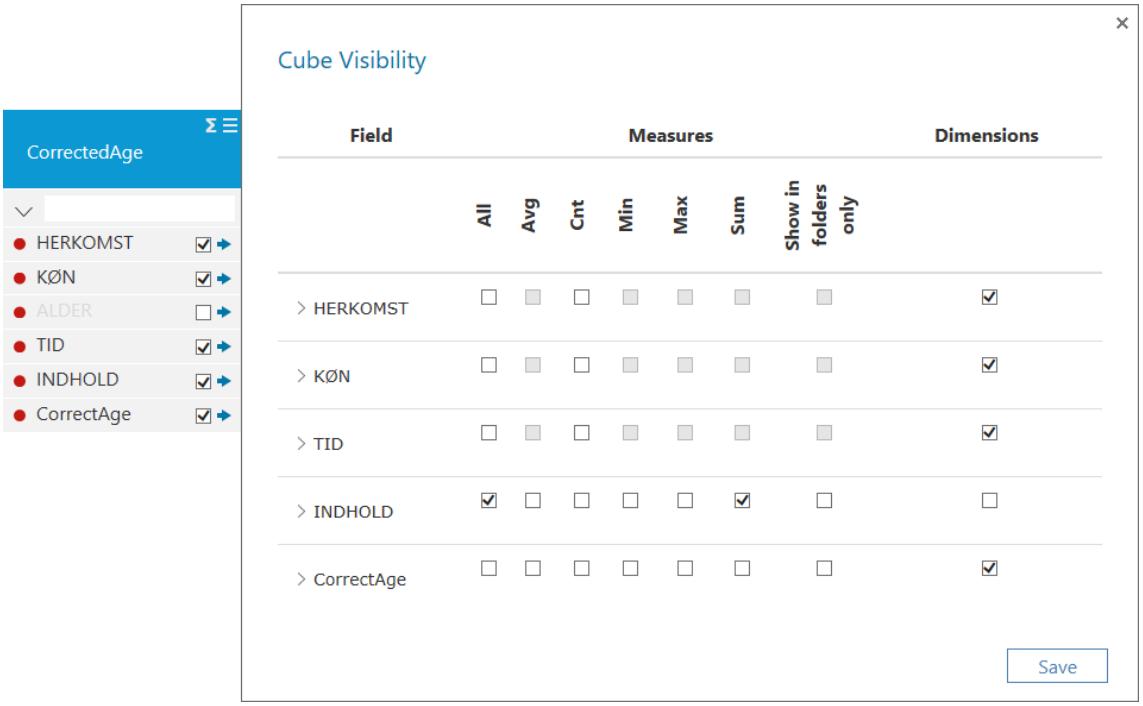
Column name: CorrectAge

Formula: left(&ALDER, find(" year",&ALDER)-1)

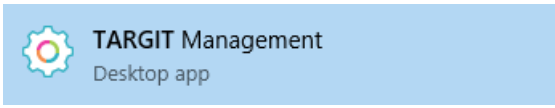
Destination type: integer

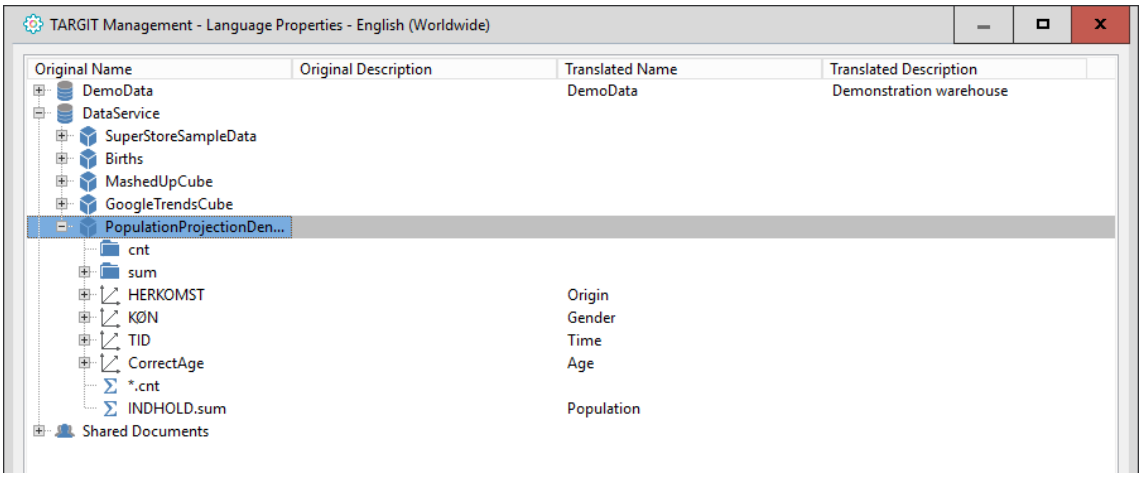
Advanced Settings

Once we have created a cube on top of this Data Source Format, we can simplify the complexity by changing some of the Attribute settings options.

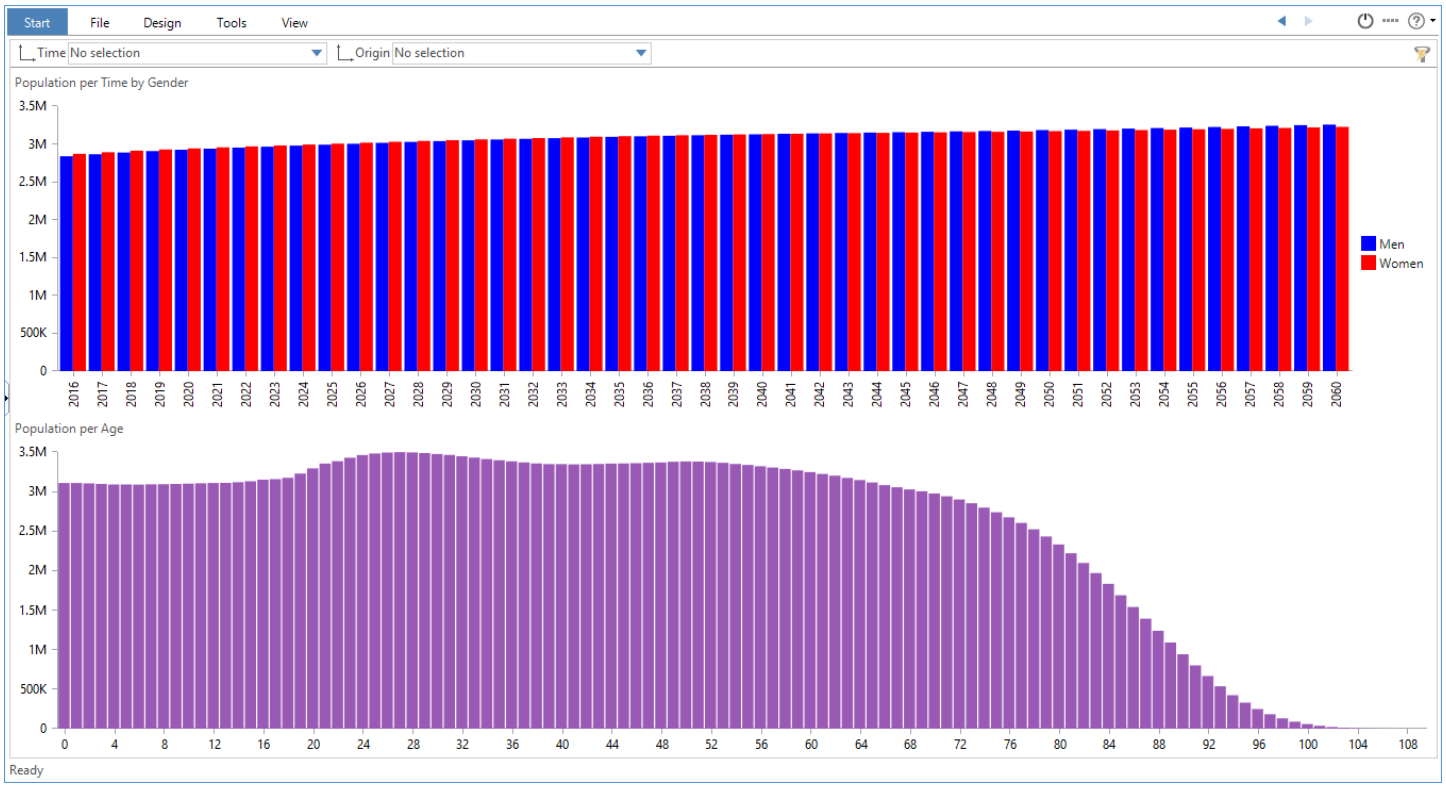


The Danish labelled columns can then be translated to proper English terms, if necessary. This is done through the Language option in the TARGIT Management client.





Finally, you can produce an analysis like this:



Save the analysis as **Lesson 4 Population Projection Analysis**.

Questions

How many 100-year-old Women and Men are projected for 2040?

Answer: Women: _____, Men: _____

Looking at descendants from non-western countries, what is the age (approximately) of the projected largest age group in 2030?

Answer: _____

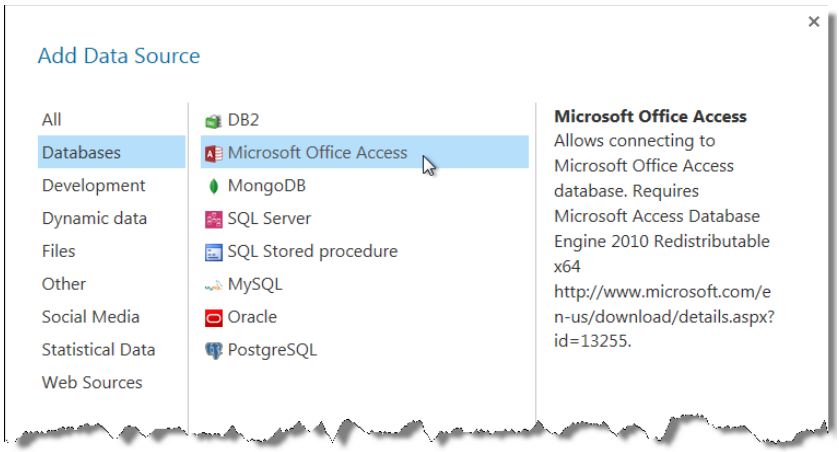
Task 2 – Sales and Budget mash up

Source Data file location: C:\TU\DataDiscovery\AccessDB.

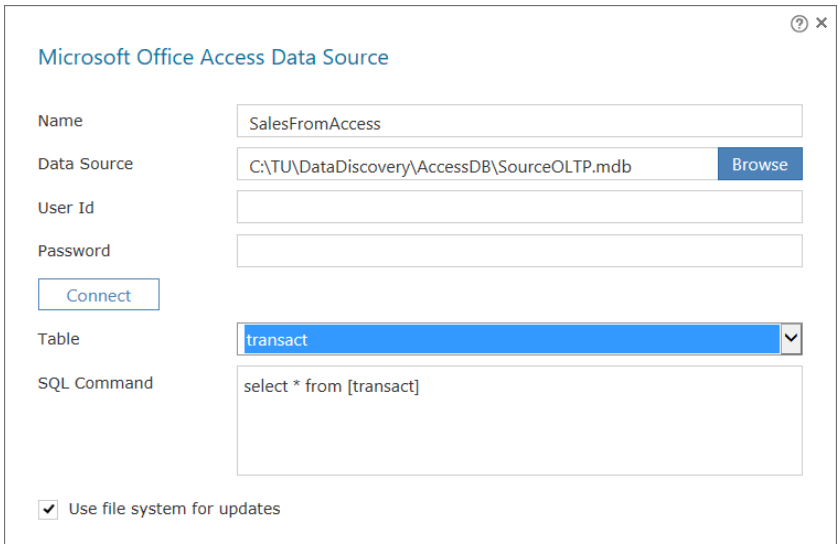
TARGIT Data Discovery offers a number of different databases to be used as data sources.

In this exercise we want to compare our sales numbers in a small Access database with some budget data in an excel sheet.

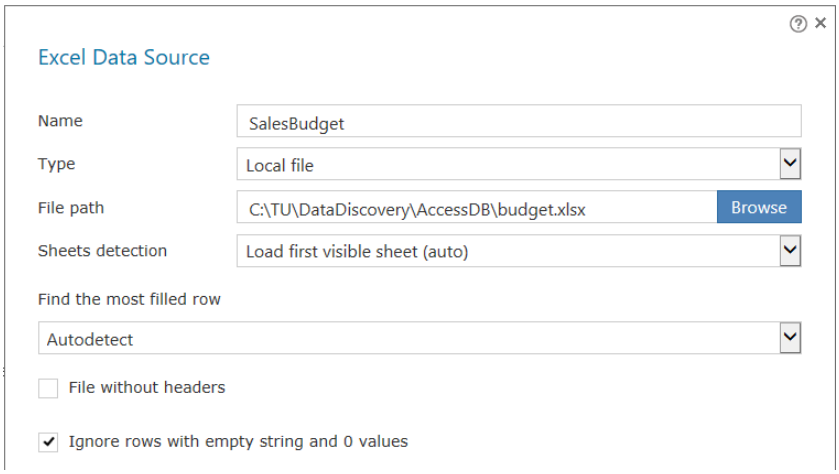
From the list of database plug-ins, we choose Microsoft Office Access and click next.



Next, we navigate to the data source, enter log-on information. We are able to select the table from the dropdown list. Select transact and click save.

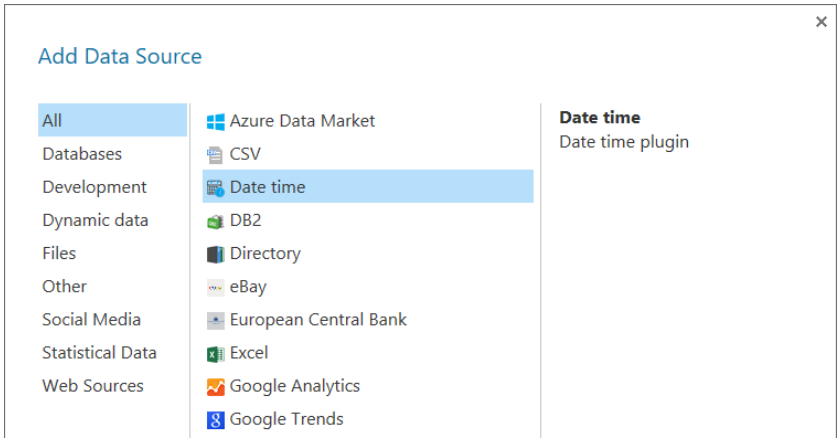


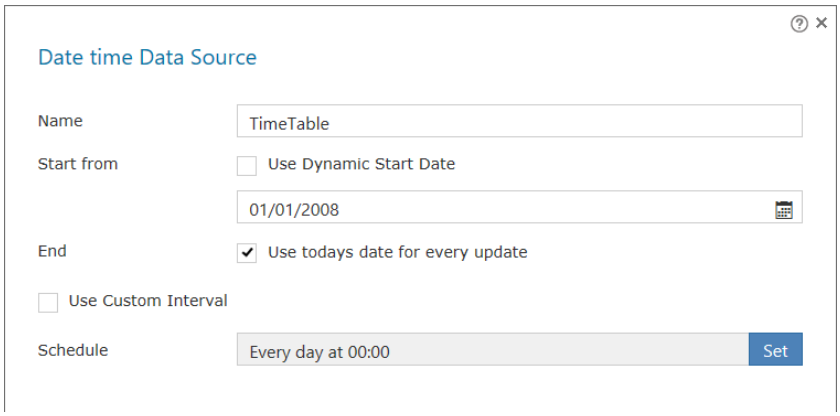
The other set of data we need to add, is our budget data from the excel sheet budget.xlsx, which is also located in the AccessDB folder.



Finally, we will create a common time dimension table that these two fact tables can be related to.

TARGIT Data Discovery offers a plugin to create a simple Time Dimension table.



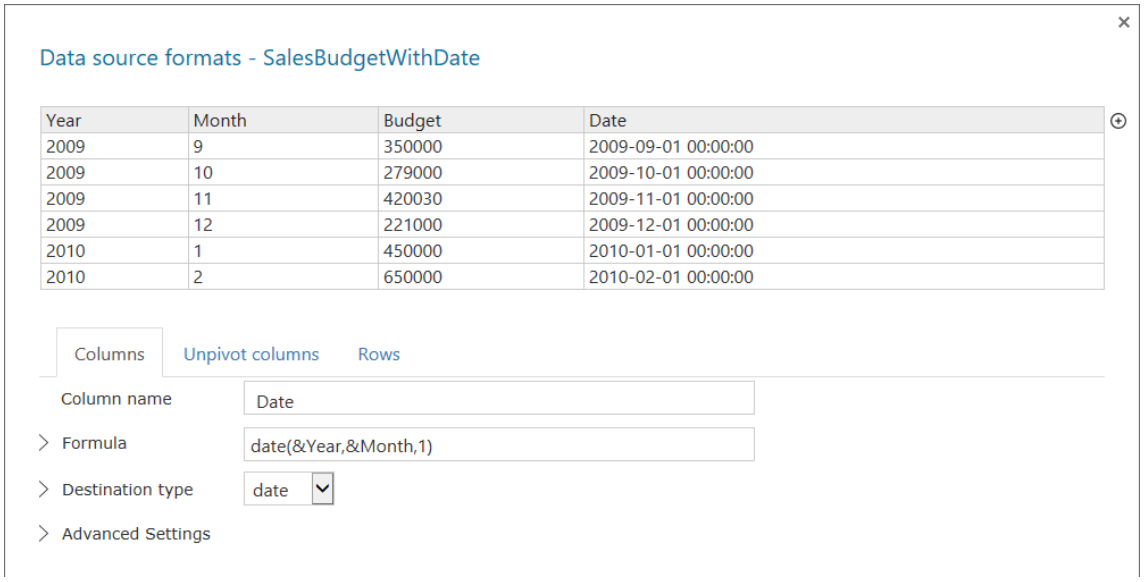


Before we can make relations from both fact tables to the Time table, at least the SalesBudget data is missing a proper Date key.

Currently it only contains a Year and a Month column.

To create a proper Date key, you must create a new Format for the SalesBudget DataSource and add a new Date column to it.

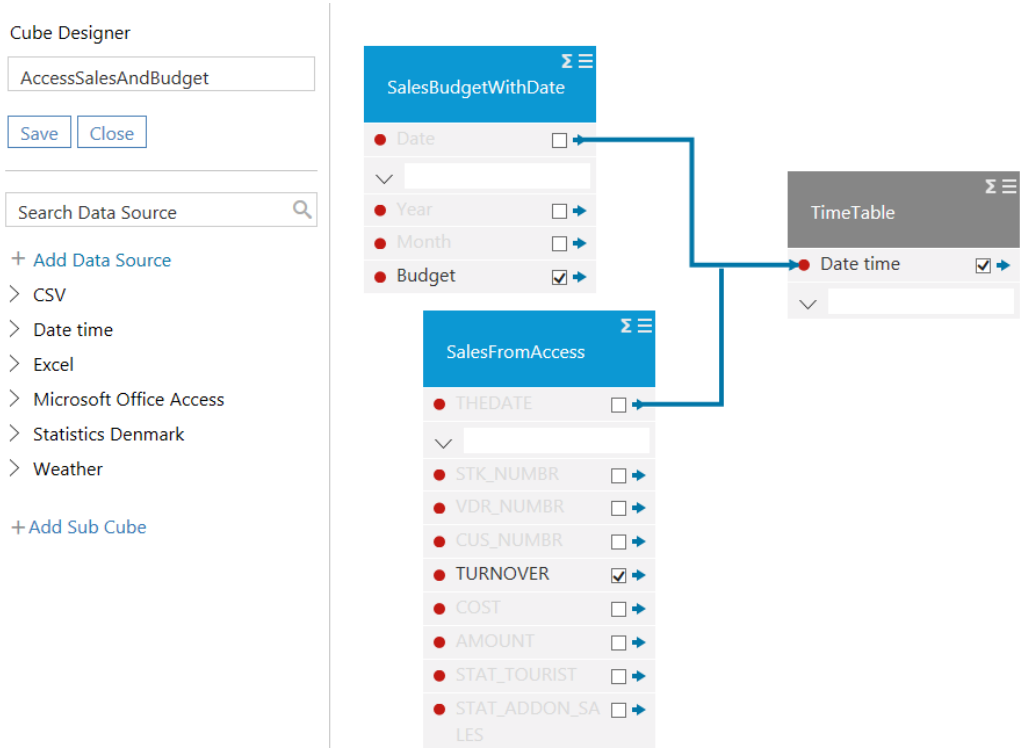
The ‘date()’ function to be used, will extract Year and Month information from the original source. By adding ‘1’ as the day level to the function, we actually create monthly budgets – all posted on the first of a month.



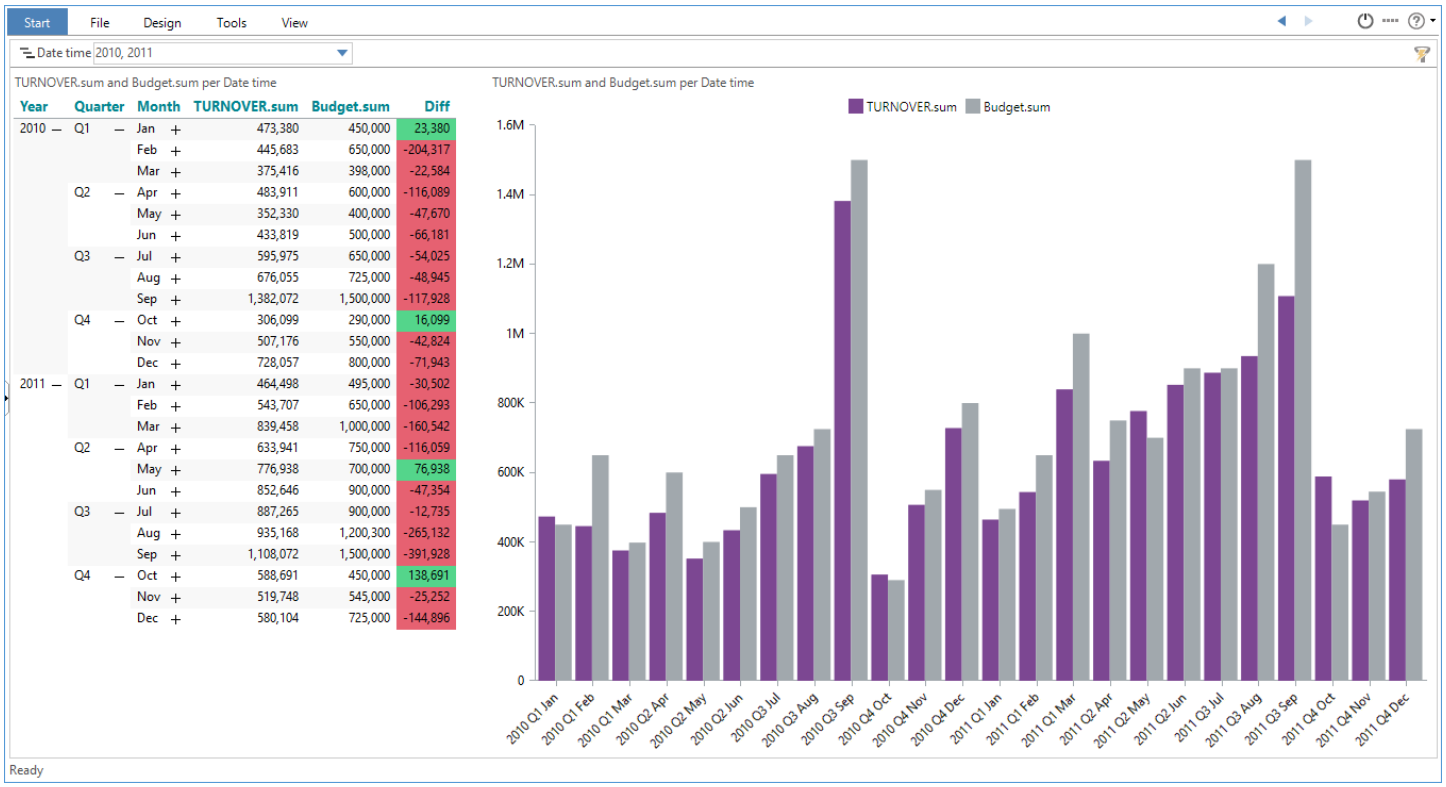
Year	Month	Budget	Date
2009	9	350000	2009-09-01 00:00:00
2009	10	279000	2009-10-01 00:00:00
2009	11	420030	2009-11-01 00:00:00
2009	12	221000	2009-12-01 00:00:00
2010	1	450000	2010-01-01 00:00:00
2010	2	650000	2010-02-01 00:00:00

Note: When previewing Date columns from different data sources you may sometimes notice that the date pattern varies – and you may then think that the date keys are inconsistent. Often this is simply an effect of different behavior of regional settings on different data sources, and the inconsistency should in most cases simply be ignored.

Now, create a cube on these three data sources.



Create an analysis on your Sales and Budget data:



Year	Quarter	Month	TURNOVER.sum	Budget.sum	Diff
2010	Q1	Jan	473,380	450,000	23,380
		Feb	445,683	650,000	-204,317
		Mar	375,416	398,000	-22,584
	Q2	Apr	483,911	600,000	-116,089
		May	352,330	400,000	-47,670
		Jun	433,819	500,000	-66,181
	Q3	Jul	595,975	650,000	-54,025
		Aug	676,055	725,000	-48,945
		Sep	1,382,072	1,500,000	-117,928
	Q4	Oct	306,099	290,000	16,099
		Nov	507,176	550,000	-42,824
		Dec	728,057	800,000	-71,943
2011	Q1	Jan	464,498	495,000	-30,502
		Feb	543,707	650,000	-106,293
		Mar	839,458	1,000,000	-160,542
	Q2	Apr	633,941	750,000	-116,059
		May	776,938	700,000	76,938
		Jun	852,646	900,000	-47,354
	Q3	Jul	887,265	900,000	-12,735
		Aug	935,168	1,200,300	-265,132
		Sep	1,108,072	1,500,000	-391,928
	Q4	Oct	588,691	450,000	138,691
		Nov	519,748	545,000	-25,252
		Dec	580,104	725,000	-144,896

Save the analysis as **Lesson 4 Sales And Budget.**

Lesson 5: Accessing Data Warehouse data

A very common request among TARGIT users is to be able to mash up external ad-hoc data with data in the Enterprise Data Warehouse – without having to involve the Enterprise Data Warehouse ETL and Data Modelling procedures.

While this, on paper, may seem like a simple task, it is in fact not possible to simply extract all Enterprise Data Warehouse data as a data source for TARGIT Data Discovery.

Instead, you will need to define a specific data extract from the Enterprise Data Warehouse needed for mash up with the external data.

TARGIT Analyses as Data Sources

When you create a TARGIT analysis upon data from your Enterprise Data Warehouse, you actually make small well-defined data extracts for each object in the analysis.

In other words, if you define a crosstab in a TARGIT analysis to show the necessary Enterprise Data Warehouse data for your data mash up, you can then use this TARGIT analysis as a data source in TARIT Data Discovery.

Before you start creating analyses to be used as Data Discovery data source input, please remember that TARGIT Data Discovery would like to see data as each row representing a transaction and each column representing a dimension attribute or a measure.

A TARGIT analysis crosstab, suitable to be used as a Data Discovery data source, might look like this:

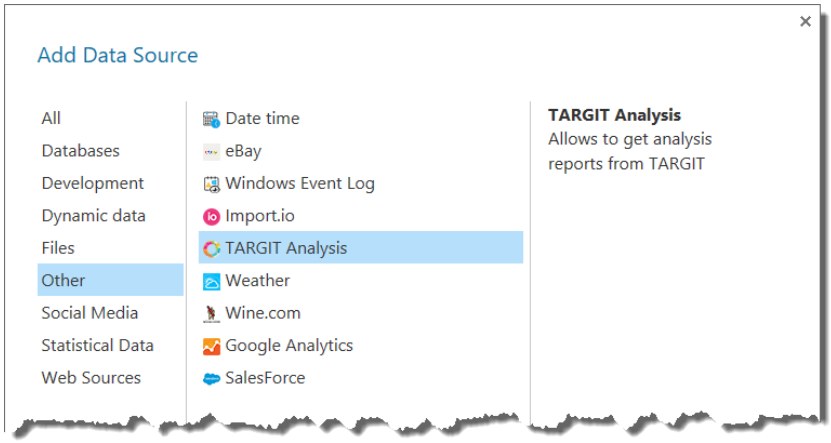
No of Sales per Time and Product Hierarchy

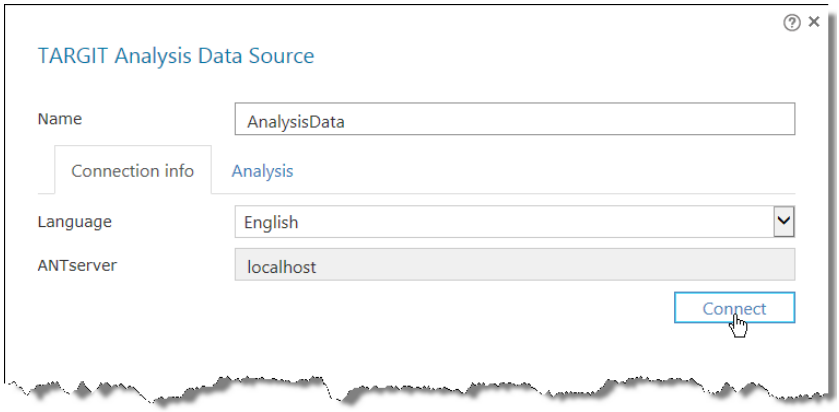
Year	Quarter	Product Group	No of Sales
2014 –	Q1 +	JEANS +	1,333
		SHIRTS +	312
		T-SHIRTS +	1,373
		UNDERWEAR +	43
	Q2 +	JEANS +	1,086
		SHIRTS +	282
		T-SHIRTS +	1,032
		UNDERWEAR +	11
	Q3 +	JEANS +	651
		SHIRTS +	190
		T-SHIRTS +	783
		UNDERWEAR +	8
	Q4 +	JEANS +	983
		SHIRTS +	232
		T-SHIRTS +	854
		UNDERWEAR +	11
2015 –	Q1 +	JEANS +	454
		SHIRTS +	141
		T-SHIRTS +	467
		UNDERWEAR +	2
		JEANS	

When you create a crosstab to be used as a Data Discovery data source, make sure that you consider this:

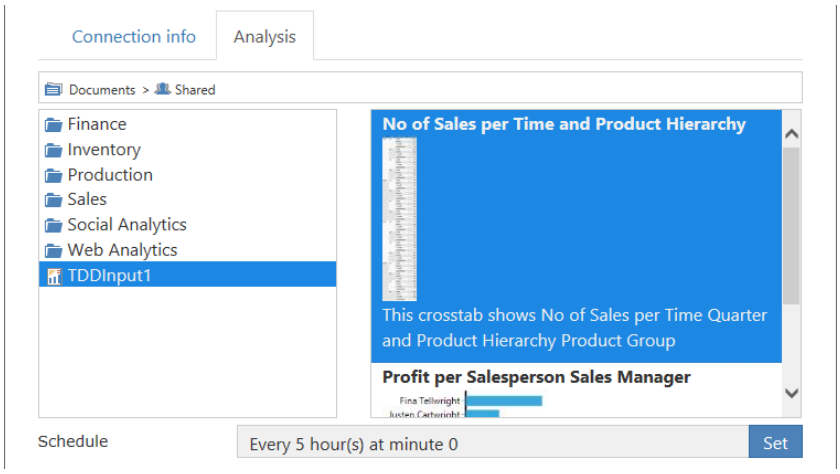
- Add all dimension to the vertical axis.
- Hide all Totals and Subtotals.

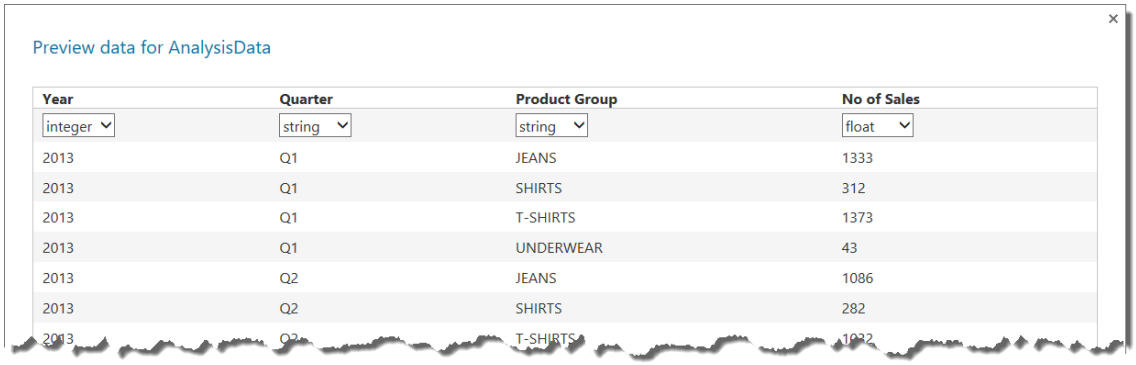
When the analysis has been saved, it can be used as a data source in Data Discovery.





...and then select the analysis and object of your choice:





Year	Quarter	Product Group	No of Sales
integer	string	string	float
2013	Q1	JEANS	1333
2013	Q1	SHIRTS	312
2013	Q1	T-SHIRTS	1373
2013	Q1	UNDERWEAR	43
2013	Q2	JEANS	1086
2013	Q2	SHIRTS	282
2013	Q2	T-SHIRTS	1032

Un-pivoting pivoted data

Now, let us assume that we have a set of source data that hasn't been so nicely formatted as we have seen so far. If possible, we would like to have data delivered in the "rows equals transactions and columns equals dimension attributes or measures" fashion, but sometimes this is not the case.

Especially when working with data coming from Excel sheets, you may experience data in the pivoted format, where dimension values have been added to both axes to form a *grid* of transactions rather than just a *list* of transactions.

Before such data can be useful to TARGIT Data Discovery, we will need to *un-pivot* the data.

To illustrate this, we can modify the previous crosstab to work with the two dimensions on separate axes:

No of Sales per Time by Product Hierarchy

		JEANS	SHIRTS	T-SHIRTS	UNDERWEAR
2014	Q1	1,333	312	1,373	43
	Q2	1,086	282	1,032	11
	Q3	651	190	783	8
	Q4	983	232	854	11
2015	Q1	454	141	467	2
	Q2	428	144	388	3
	Q3	549	172	634	2
	Q4	733	221	699	8
2016	Q1	700	180	875	5
	Q2	711	197	838	5
	Q3	585	168	742	1
	Q4	940	301	1,065	3
2017	Q1	581	235	690	3
	Q2	555	145	493	
	Q3	643	213	786	1
	Q4	1,070	264	888	3

These are the exact same data as we looked at previously – now just in a pivoted mode.

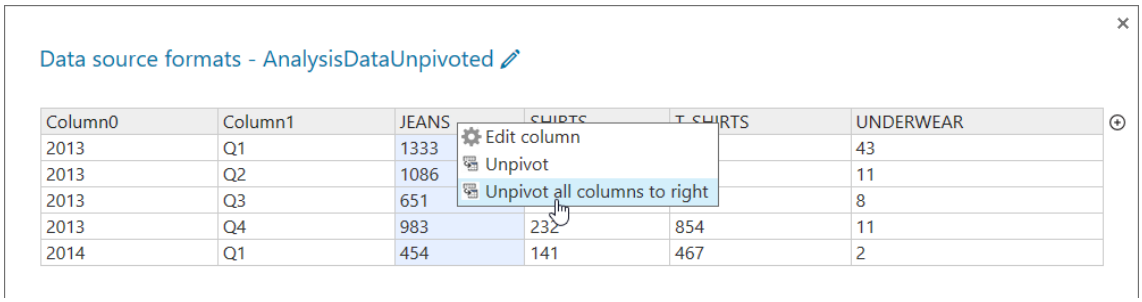
If we save this analysis with a new name and add it as a data source to Data Discovery we can see that the current format is not useful. If we continued from here, Jeans, Shorts, T-Shirts and Underwear would be treated as four separate dimensions or measures rather than just one dimension with four different dimension members.

Preview data for AnalysisDataPivoted

Column0	Column1	JEANS	SHIRTS	T-SHIRTS	UNDERWEAR
string	string	float	float	float	float
2013	Q1	1333	312	1373	43
2013	Q2	1086	282	1032	11
2013	Q3	651	190	783	8
2013	Q4	983	232	854	11
2014	Q1	454	141	467	2
2014	Q2	428	144	388	3
2014	Q3	549	172	634	2

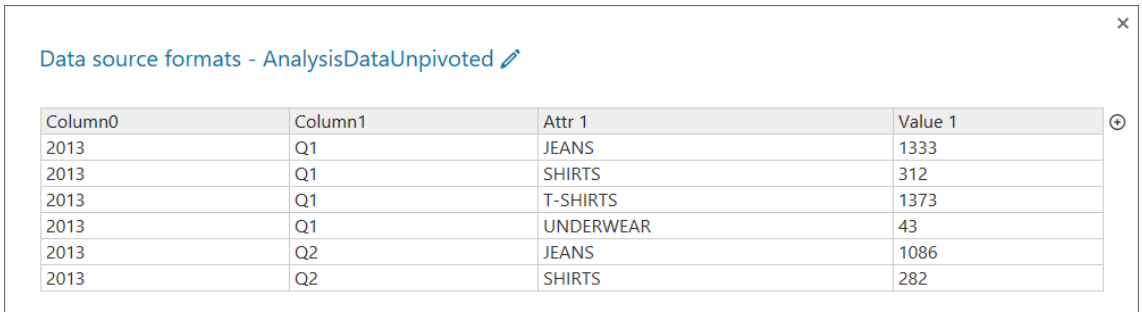
Important: In this example, we will need to change the data type for years to *string* before proceeding.

To un-pivot these data, we will have to create a new Format for the data source. All the columns Jeans, Shirts, T-Shirts and Underwear should be un-pivoted to the vertical axis. The easiest way to do this, is to right click the first column that should be un-pivoted, and then choose to un-pivot all columns to the right.



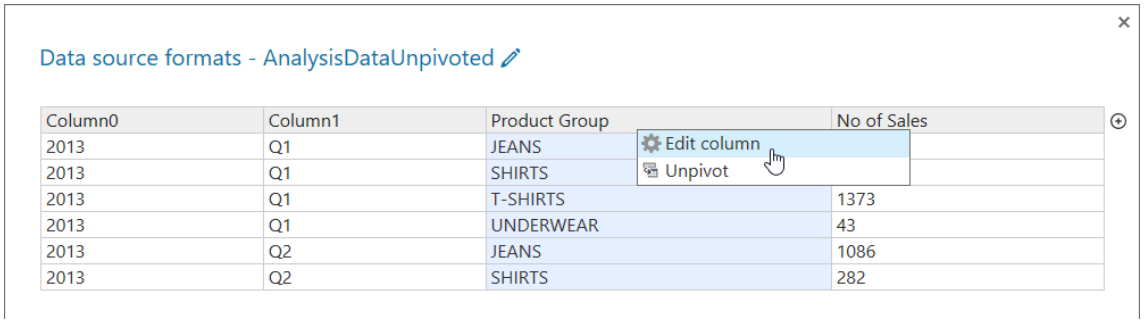
Column0	Column1	JEANS	SHIRTS	T-SHIRTS	UNDERWEAR
2013	Q1	1333			43
2013	Q2	1086			11
2013	Q3	651			8
2013	Q4	983	232	854	11
2014	Q1	454	141	467	2

This should produce an un-pivoted result like this:



Column0	Column1	Attr 1	Value 1
2013	Q1	JEANS	1333
2013	Q1	SHIRTS	312
2013	Q1	T-SHIRTS	1373
2013	Q1	UNDERWEAR	43
2013	Q2	JEANS	1086
2013	Q2	SHIRTS	282

Before saving the Format, we may want to rename the last two columns. Right click to rename.



Column0	Column1	Product Group	No of Sales
2013	Q1	JEANS	
2013	Q1	SHIRTS	
2013	Q1	T-SHIRTS	1373
2013	Q1	UNDERWEAR	43
2013	Q2	JEANS	1086
2013	Q2	SHIRTS	282

Exercises Lesson 5

In this exercise you will use TARGIT analysis as data sources for looking at Actuals for Salespersons in YTD 2015.

Furthermore, we will produce and modify Budget data for *all* of 2015.

Task 1 – preparing data sources

Create a new TARGIT analysis with a new crosstab: **Revenue per Salesperson by Month**. These data are located in the DemoData\Sales cube.

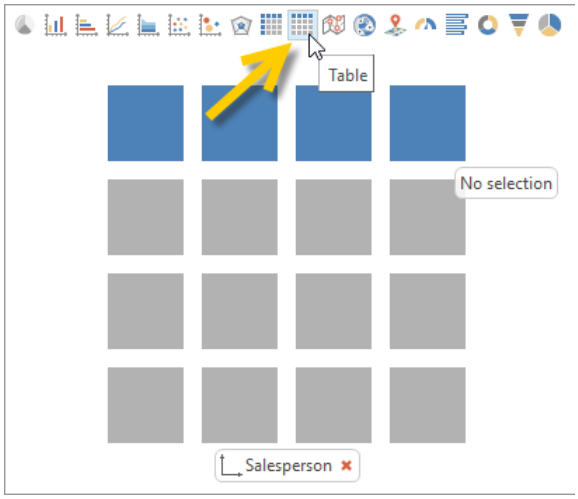
Time 2016 Q1, 2016 Q2, 2016 Q3

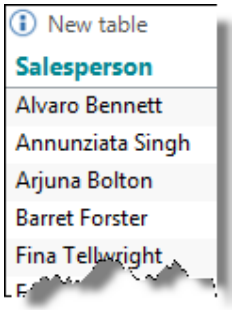
Revenue per Salesperson by Time Month

	Total	January	February	March	April	May	June	July	August	September
Total	**$32,741,316.01**	**$2,187,810.68**	**$3,350,159.46**	**$4,686,659.65**	**$4,657,380.01**	**$3,135,432.45**	**$3,803,719.24**	**$2,310,369.93**	**$4,107,738.44**	**$4,502,046.16**
Alvaro Bennett	**$697,736.96**	$51,719.20	$136,793.02	$83,070.00	$123,831.24	$85,534.80	$66,892.80	$36,628.80	$66,248.00	$47,019.10
Annunziata Singh	**$44,648.76**		($8,548.80)	$25,917.32			$11,544.00	$21,378.24		($5,642.00)
Arjuna Bolton	**$1,504,544.63**	$100,715.16		$258,825.24	$596,326.82	$57,671.98	$201,362.20		$122,271.17	$167,372.06
Barret Forster	**$4,603,058.80**	$381,810.00	$532,660.86	$555,428.64	$358,542.60	$510,872.70	$461,182.80	$354,393.00	$909,711.58	$538,456.62
Fina Tellwright	**$98,032.90**	$18,383.04		$7,787.52	$7,207.20		$30,371.33		$34,283.81	
Fortunato Crawford	**$3,726,826.43**	$63,585.60	$666,933.53	$935,264.60	$370,316.01	$117,295.63	$479,155.69	$137,997.60	$559,370.26	$396,907.51
Jessika Thornton	**$245,070.80**	$37,736.40		$7,800.00	$78,384.80	$83,553.60	$16,224.00			$21,372.00
Juniper Peabody	**$115,385.40**	$26,488.80				$7,784.40	$32,572.80		$37,159.20	$11,380.20
Justen Cartwright	**$1,168,482.94**	$75,237.76	$101,423.40	$75,301.20	$188,371.56	$192,577.38	$27,190.80	($8,653.01)	$259,857.34	$257,176.50
Keren Rose	**$236,908.10**	$39,347.10	$58,000.80	($23,405.20)	$47,252.40	$17,550.00	$37,432.20		$32,370.00	$28,360.80
Luitpold Whyman	**$5,318,497.14**	$206,909.56	$561,120.12	$540,732.97	$861,985.80	$562,811.60	$700,715.61	$489,880.30	$511,106.23	$883,234.95
Madelina Hewitt	**$139,172.28**		$8,611.20			$30,388.80	$34,125.00		$11,540.88	$54,506.40
Maggie Warren	**$570,180.00**	$26,410.80	$45,427.20	$79,185.60	$113,950.20	$30,108.00	$60,512.40	$34,226.40	$109,067.40	$71,292.00
Nicolle Bramble	**$78,276.90**	$1,487.77	$8,819.93	$14,804.40	$24,804.00		$28,360.80			
Opaline Webster	**$237,439.80**	$4,739.80		$53,851.20	$52,353.60	$53,362.40	$47,658.00	$7,644.00	$7,644.00	$10,186.80
Regena Wilder	**$204,877.40**	$30,513.60	$19,908.20	$46,394.40		$54,568.80	$23,384.40	$17,893.20	$12,214.80	
Rhetta Parker	**$259,033.32**	$10,686.00	$41,184.00	$24,128.52	$34,086.00	$49,498.80	$53,508.00		$11,684.40	$34,257.60
Sanjeev Walton	**$6,637,671.84**	$537,263.29	$401,627.20	$1,241,294.76	$1,011,660.00	$371,940.40	$845,798.20	$632,504.60	$747,732.49	$847,850.91
Savannah Morell	**$159,057.60**			$78,795.60	$22,698.00		$9,734.40	$11,544.00		$36,285.60
Shukriyya Burrows	**$241,167.68**	($40,760.20)		$68,870.88	$56,004.00	$58,188.00	$38,274.60	$32,713.20	$7,924.80	$19,952.40
Verda Heath	**$322,288.20**			$76,369.80	$84,848.40	$51,838.80	$37,455.60	$21,637.20	$27,877.20	$22,261.20
Vern Ferguson	**$6,132,958.13**	$615,537.00	$776,198.80	$536,242.20	$624,757.38	$799,886.36	$560,263.60	$520,582.40	$639,674.88	$1,059,815.51

For the cross tab above, make sure that you set a criterion, **Time = 2016 Q1, 2016 Q2, 2016 Q3**.

Furthermore, add a simple table with just the Salesperson dimension. This table will serve as a common Salesperson dimension in Data Discovery.





Save the analysis as **SalespersonsActuals**.

Add these two tables as ***TARGIT Analysis*** data sources to Data Discovery.

Create a new TARGIT analysis with a new crosstab: **Revenue Goal per Salesperson by Month**.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Salesperson	January	February	March	April	May	June	July	August	September	October	November	December
2	Alvaro Bennett	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000
3	Annunziata Singh	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000
4	Arjuna Bolton	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000
5	Barret Forster	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000
6	Charity Carmichael												10,218
7	Fina Tellwright	20,263		5,438	6,503		21,288		13,780		28,729		11,676
8	Fortunato Crawford	204,252	857,382	704,341	129,208	166,856	915,113	179,432	334,521	359,895	1,030,586	360,899	662,764
9	Jessika Thornton	41,569		5,465	67,777	107,181	16,856	0		19,240	42,347	8,970	32,759
10	Juniper Peabody	29,131				8,540	22,811		14,837	10,233	7,147	20,460	13,204
11	Justen Cartwright	39,895	43,629	88,294	111,561	52,328	28,587	0	74,151	259,948	78,958	257,654	103,408
12	Keren Rose	43,297	52,256	15,096	42,466	19,339	26,223		12,983	23,455	30,616	46,530	18,784
13	Luitpold Whyman	191,032	115,388	415,316	512,928	515,027	546,982	880,614	145,803	1,148,226	812,368	592,188	881,000
14	Madelina Hewitt		7,800			33,470	23,895		4,680	46,123	22,922		24,300
15	Maggie Warren	29,086	40,820	34,049	95,796	33,125	42,320	51,350	43,659	64,221	17,206	39,546	42,993
16	Mechtilde Watts											33,891	34,385
17	Nicolle Bramble	13,814	7,930	218	31,734		14,418					14,186	2,319
18	Opaline Webster	14,549		37,670	47,107	65,256	33,376	11,440	3,120	9,172	34,190	30,167	10,990
19	Regena Wilder	33,571	10,270	32,487		60,049	16,369	26,780	4,917		9,234	11,050	1,428
20	Rhetta Parker	11,700	37,054	16,878	30,683	48,754	37,469		4,703	30,810	17,604	68,749	23,786
21	Sanjeev Walton	312,318	389,021	675,059	827,003	519,355	278,263	611,570	161,400	440,190	706,094	700,229	519,949
22	Savannah Morell			55,167	20,394		6,813	17,290		32,686	7,930	24,166	9,692
23	Shukriyya Burrows	8,720		48,207	50,418	60,819	26,816	49,016	3,176	17,914			6,852
24	Verda Heath			53,448	76,292	44,078	26,173	32,500	11,144	20,057	11,134	10,384	26,161
25	Vern Ferguson	809,817	483,837	425,028	872,622	807,036	142,149	523,007	249,157	655,746	682,529	430,842	515,698

Make sure that you:

- Hide Totals and Subtotals.
- Format numbers: Numeric, No decimals.
- Set a criterion, Time = 2016.

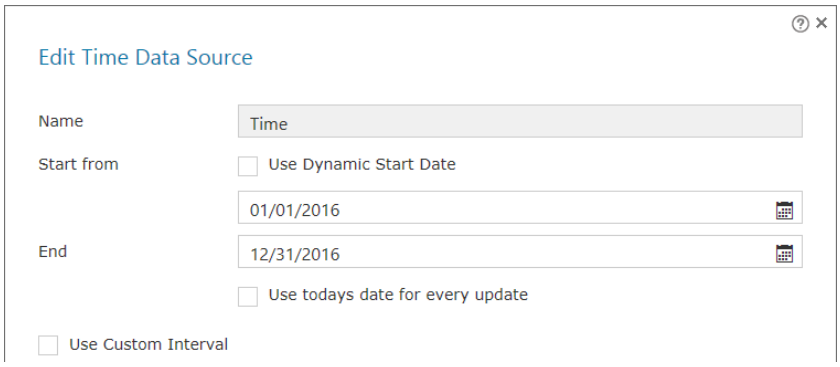
Now, **export the crosstab to Excel** (from TARGIT's File menu) and…

- Change the first four Salespersons' Budgets to fixed monthly budget: 100000, 10000 200000 and 500000 respectively.
- **Important**: Insert the text string 'Salesperson' in cell A1.

Save the Excel file as **SalespersonsBudget**.

Add the Excel file as a data source to Data Discovery.

Finally, create a ***Date time*** data source in Data Discovery:

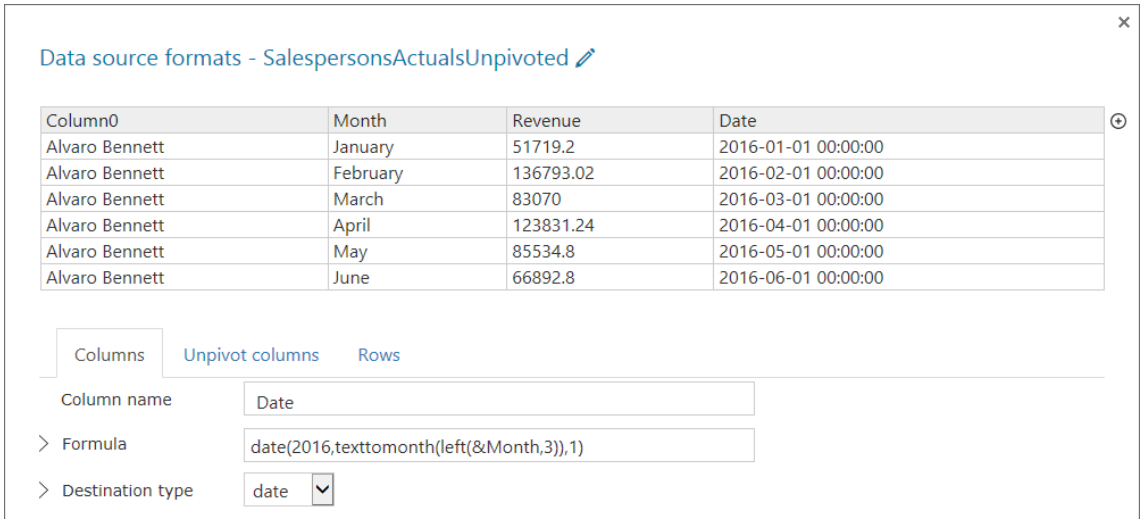


Task 2 – modifying data sources

The two data sources SalespersonActuals and SalespersonBudget will need to be un-pivoted and modified to get a Date key column added to them.

For each of those two files, do the following:

- Create a ***Format***.
- Un-pivot all date columns.
- Rename columns properly.
- Add a new column to the format.
- Use the date() function to create a proper date key in this column.

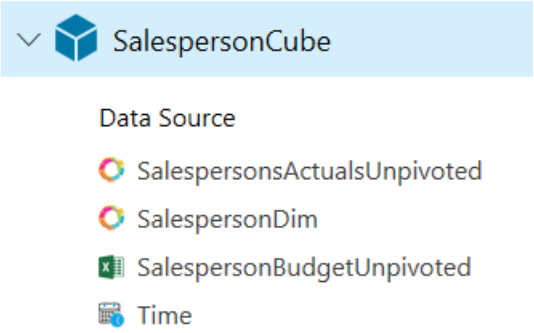


Notice: Depending on your regional settings, you may need to experiment a bit with the exact formula syntax to get a proper date key. The date key result should hold the pattern YYYY-MM-DD.

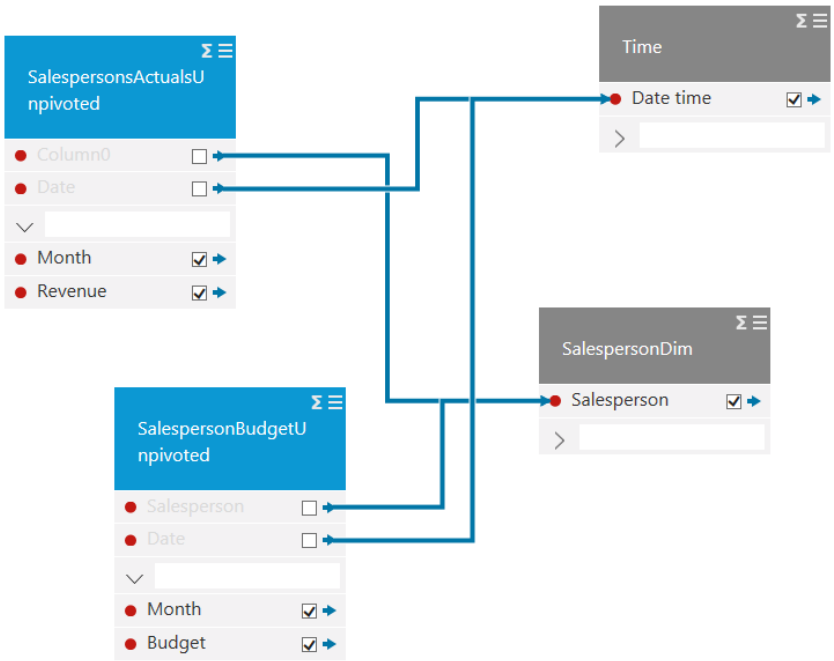
If the above formula produces a wrong result, try switching month and day input to the date() function like this:

Task 3 – creating the cube

At this point we should be able to create a cube on basis of the four available data source:

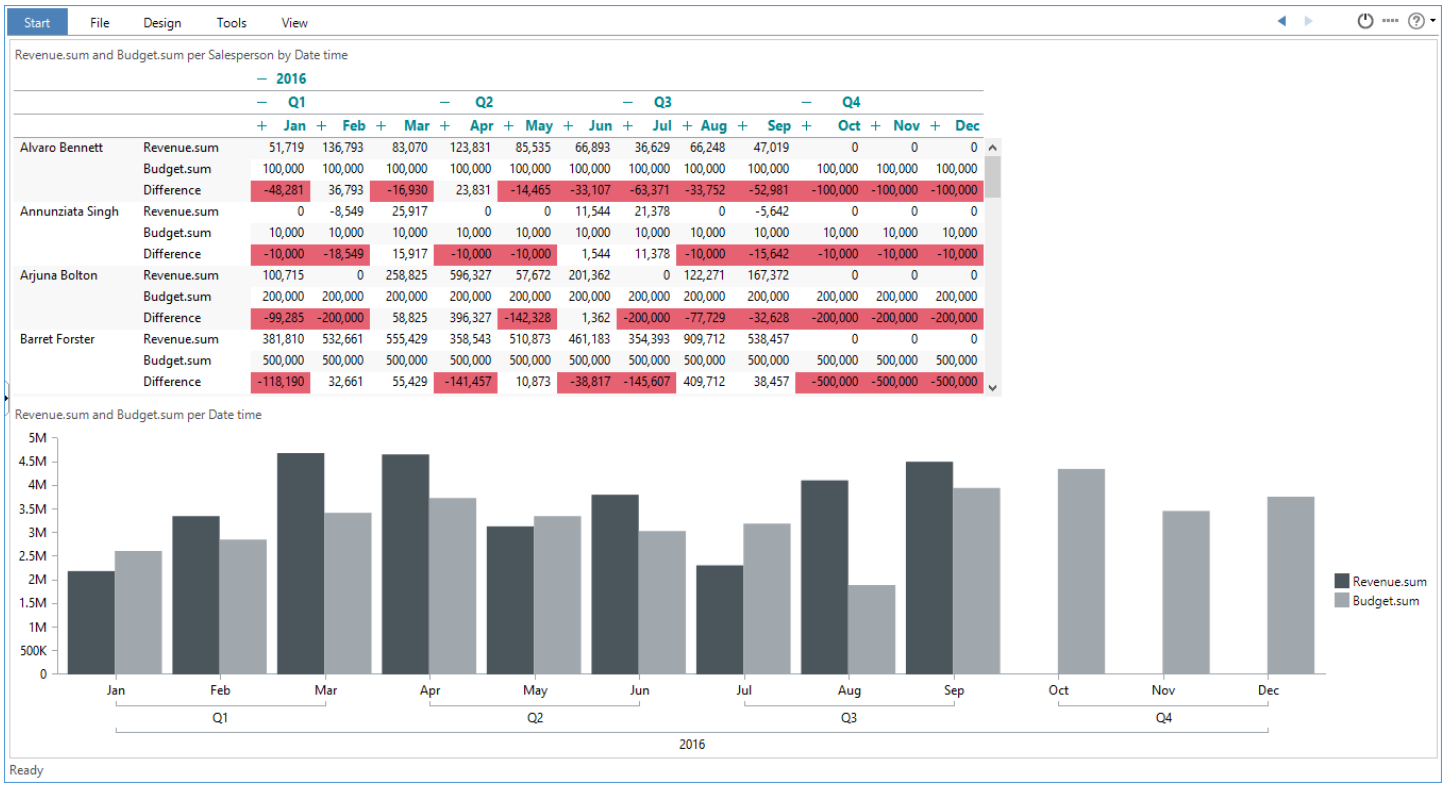


The four data sources should be related in this way. Also, make sure that you work with proper settings for Attribute settings.



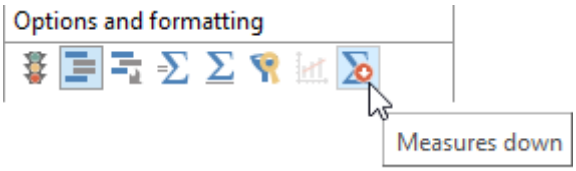
Task 4 – create an analysis

At this point you should be able to create an analysis like this:



		2016											
		Q1			Q2			Q3			Q4		
		Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Alvaro Bennett	Revenue.sum	51,719	136,793	83,070	123,831	85,535	66,893	36,629	66,248	47,019	0	0	0
	Budget.sum	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000	100,000
	Difference	-48,281	36,793	-16,930	23,831	-14,465	-33,107	-63,371	-33,752	-52,981	-100,000	-100,000	-100,000
Annunziata Singh	Revenue.sum	0	-8,549	25,917	0	0	11,544	21,378	0	-5,642	0	0	0
	Budget.sum	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000
	Difference	-10,000	-18,549	15,917	-10,000	-10,000	1,544	11,378	-10,000	-15,642	-10,000	-10,000	-10,000
Arjuna Bolton	Revenue.sum	100,715	0	258,825	596,327	57,672	201,362	0	122,271	167,372	0	0	0
	Budget.sum	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000	200,000
	Difference	-99,285	-200,000	58,825	396,327	-142,328	1,362	-200,000	-77,729	-32,628	-200,000	-200,000	-200,000
Barret Forster	Revenue.sum	381,810	532,661	555,429	358,543	510,873	461,183	354,393	909,712	538,457	0	0	0
	Budget.sum	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000	500,000
	Difference	-118,190	32,661	55,429	-141,457	10,873	-38,817	-145,607	409,712	38,457	-500,000	-500,000	-500,000

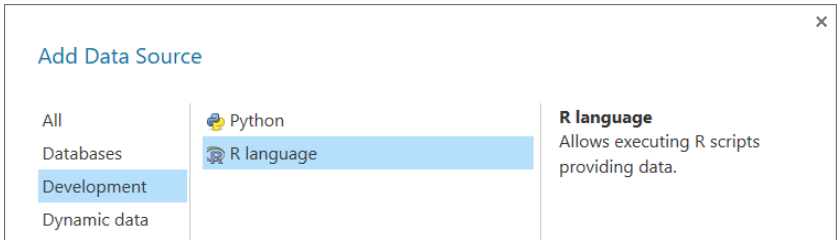
Notice: The formatting option *Measures down* has been used to format the crosstab as seen above.



Save the analysis as **Lesson 5 Salesperson Actuals and Budget**.

Lesson 6: Merging data with R

TARGIT Data Discoverer includes an option to work with the “R” language.



R is a powerful statistical programming language that allow you to run scripts for extracting data sets for statistical purposes.

The purpose of this class is not to teach you details on how to use R – many online resources are available to give you a head start on that.

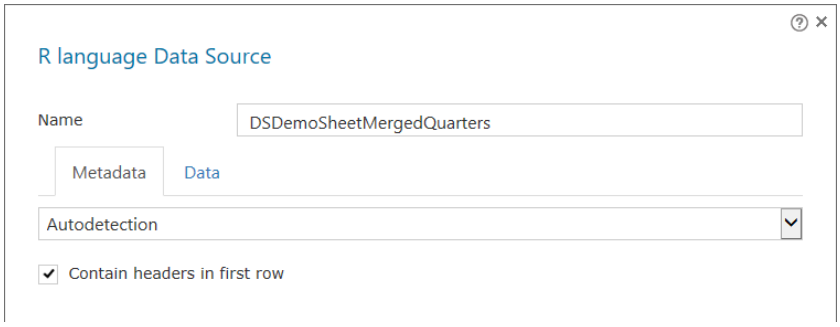
In this class you will simply learn how to work with one specific and useful script that allows you to merge data of multiple files into one data source.

Imagine this scenario: You have a source data system that is capable of delivering a daily data dump. This data dump might be e.g. a comma separated file (CSV) that goes into a specific folder. The system automatically creates file names with a continuous date stamp as part of the file name.

Now, you can create a script with one of R’s functions that will merge all files in the folder into one data source.

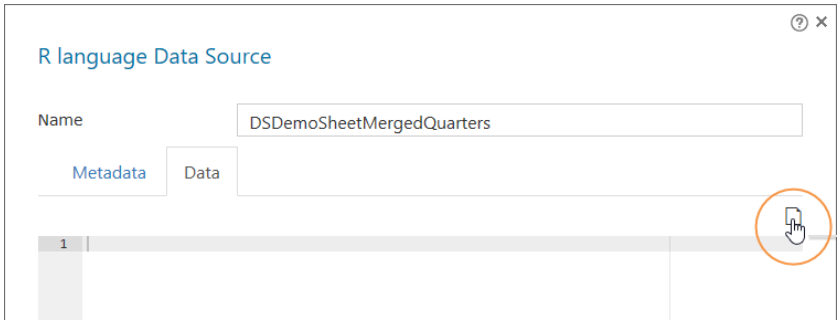
Creating an R language data source

Once you have added an R language data source, make sure that you name it properly on the first tab.



On the second tab you will initially see a default script. It is in fact not a working script as all lines have been marked as comments due to the hashtag (#) sign.

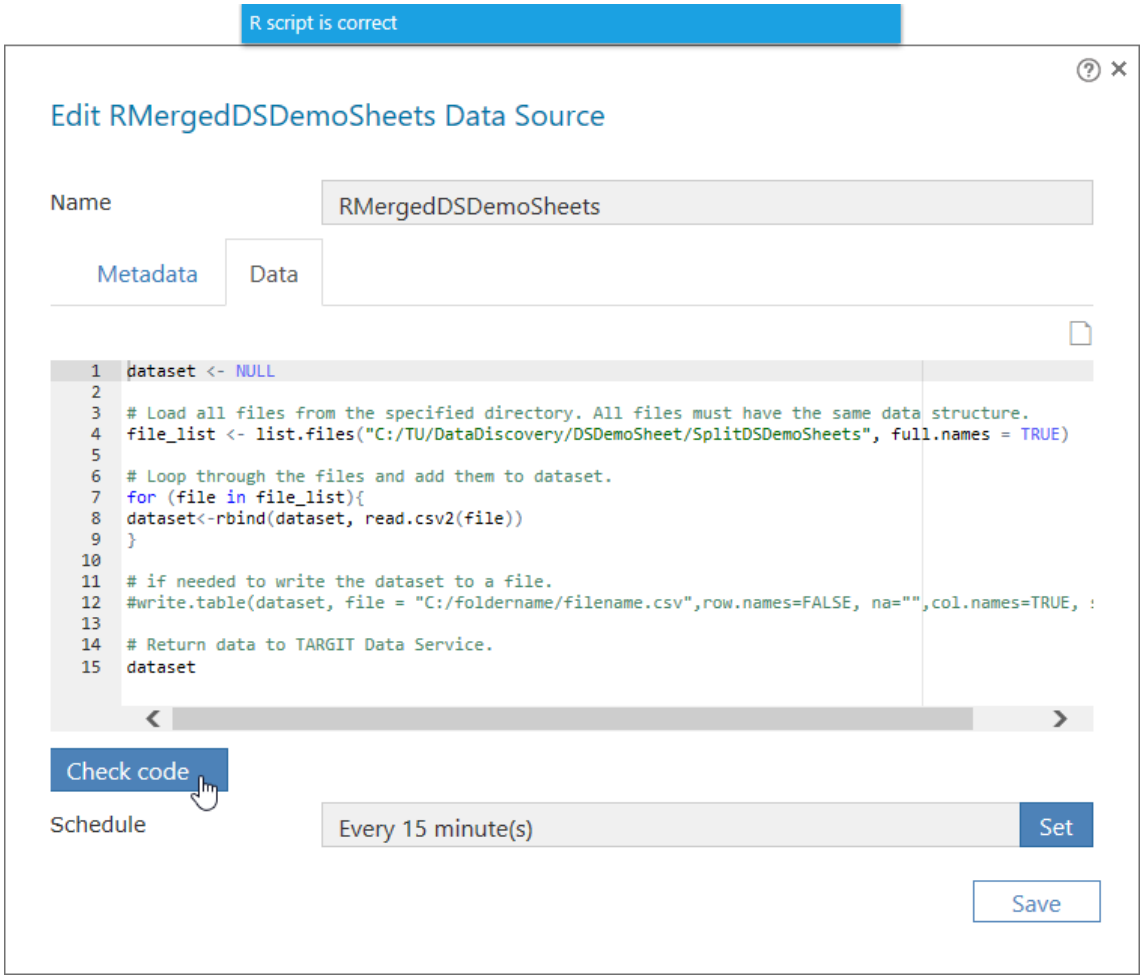
You may now either write your own R script from scratch, or you may import an existing script from a file.



If you open the R script file, MergeFiles.R, from the C:\TU\DataDiscovery\R\Script folder, you will have a pre-cooked script that just needs slight modifications to merge your own set of data.

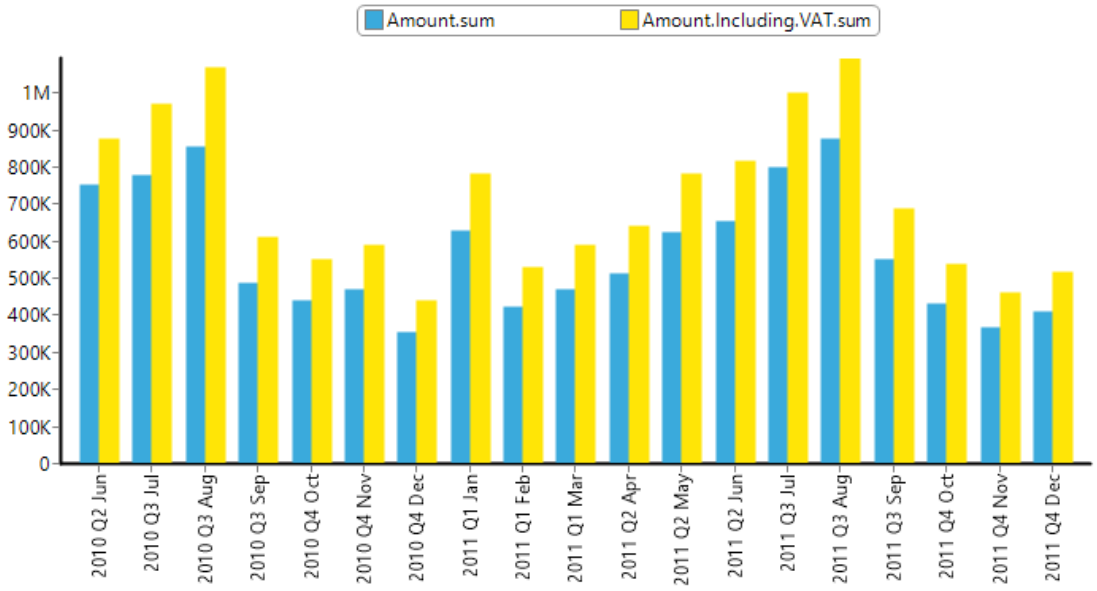
One thing you probably will need to change, is the directory path to where your files to be merged are located.

Important: Notice the *forward* slashes (/) in the directory path!



Once you save the script, it will run and create one merged data set to be used as a data source for TARGIT Data Discovery.

If we then create a cube on top of that data source, we can verify the merge result by creating a TARGIT Analysis like this:

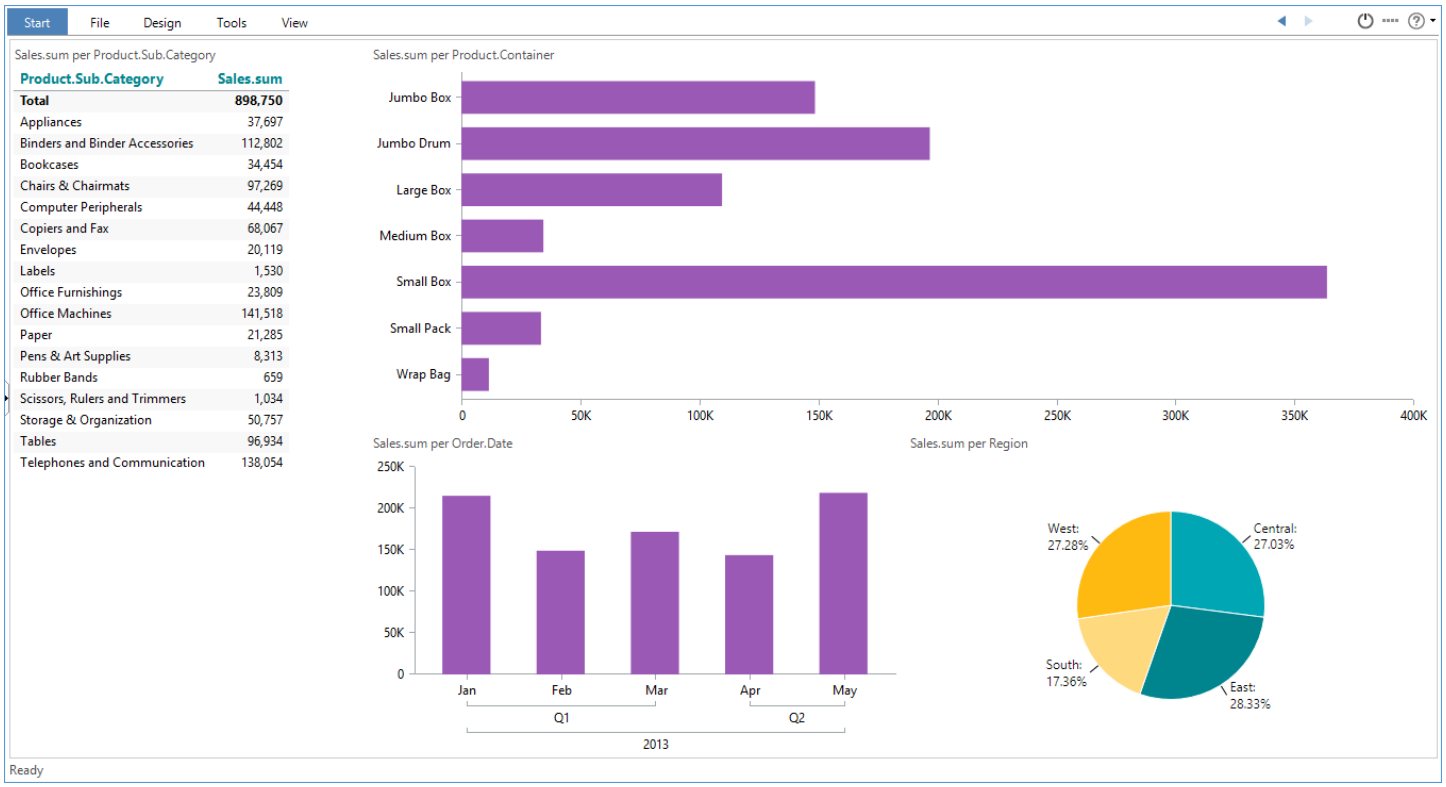


Exercises Lesson 6

Task 1

The directory C:\TU\DataDiscovery\R\Data contains five months of data from the SuperStore – each month's data in a separate CSV file.

Use the R script to merge these files into one data set and create a cube and an analysis like this:



Save the analysis as **Lesson 6 Merged Data Analysis**.

Lesson 7: Data Discovery Collaboration

An important thing to know about the Data Sources and the Cubes you create in TARGIT Data Discovery is that the things you create are by default available only to yourself!

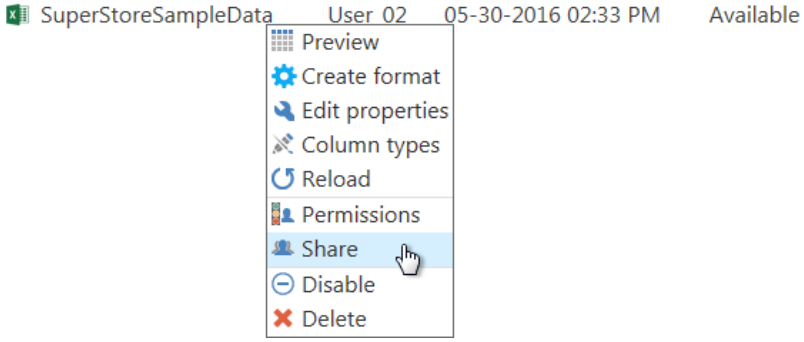
In other words, even though many people within an organization are working on the same central TARGIT Data Discovery installation, their data are by default not shared. We think this is the best way of handling data governance: That, by default, you do not need to fear that the wrong people might accidently be able to see your uploaded data.

Only when you actively *want* to or *need* to share your data with other people within your organization, you can of course do that – and only with the people you select.

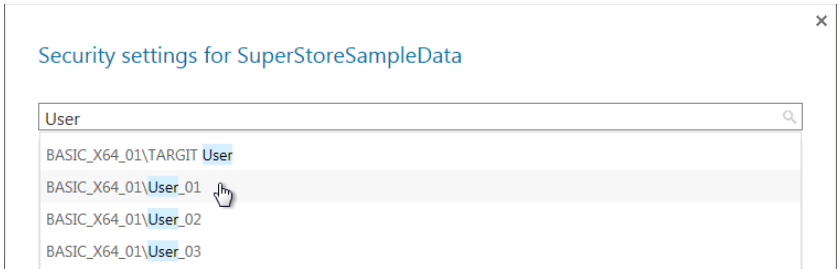
Sharing Data Sources and Cubes

Imagine that one of your colleagues have discovered a valuable data source on the Internet that she used in a cube for one of her internal projects. In fact, you need the same data source for one of your projects as well, but rather than duplicating the data source (and taking up additional memory on the server), you ask your colleague to simply share the data source with you.

To share one of your data sources, you should right click the data source and select “Share”.

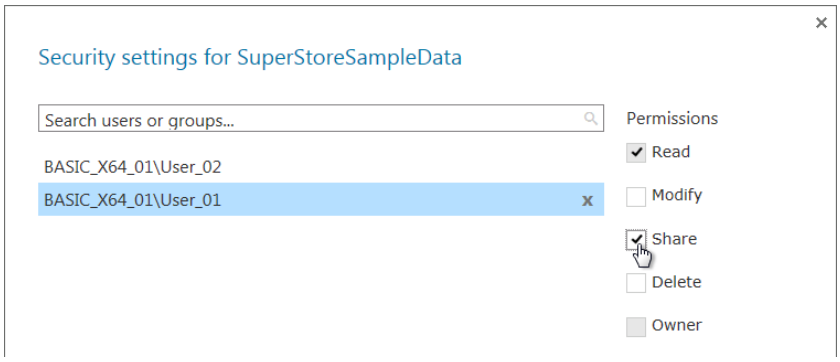


With whom you would like to share it, is selected from your list of Active Directory users within your domain.

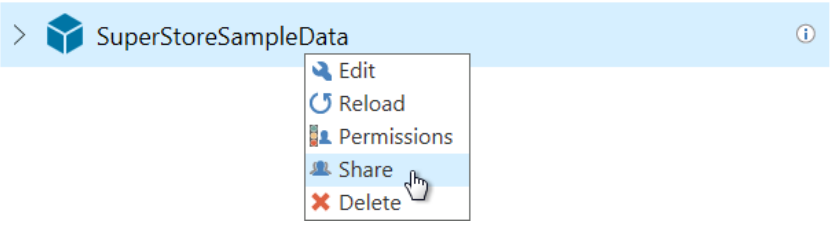


Setting Permissions

A user, with whom you have shared a Data Source, will by default only have permissions to *Read* that data source. Right click the Data Source again and select “Permission” to change these permissions. You can even transfer ownership of a Data Source in this way.



Sharing and setting permissions for Cubes works in an identical manner.



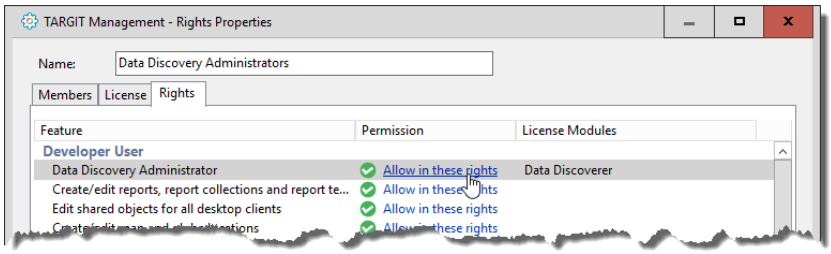
Exercises Lesson 7

No exercises available for this lesson.

Lesson 8: Administrator options

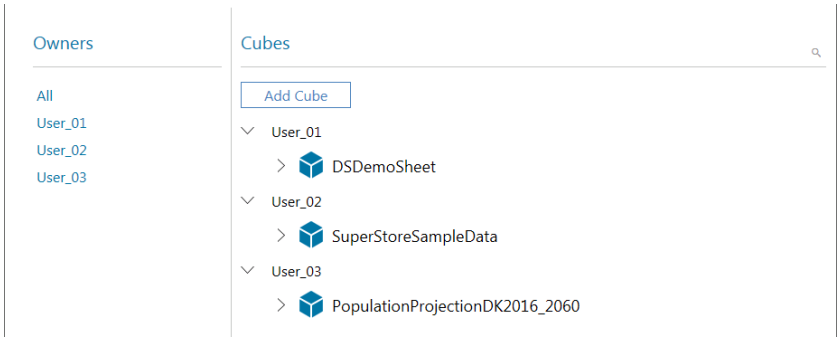
Within the organization at least one employee will probably need to become a Data Discovery Administrator. While the individual Data Discovery users can see their own Data Sources and Cubes – whether these have been created by themselves or shared by someone else – the Administrator will at any time be able to get an overview of *all* existing Data Sources and Cubes.

To obtain these rights, you must be appointed a Data Discovery Administrator, and this is done by the Rights settings in the TARGIT Management client.



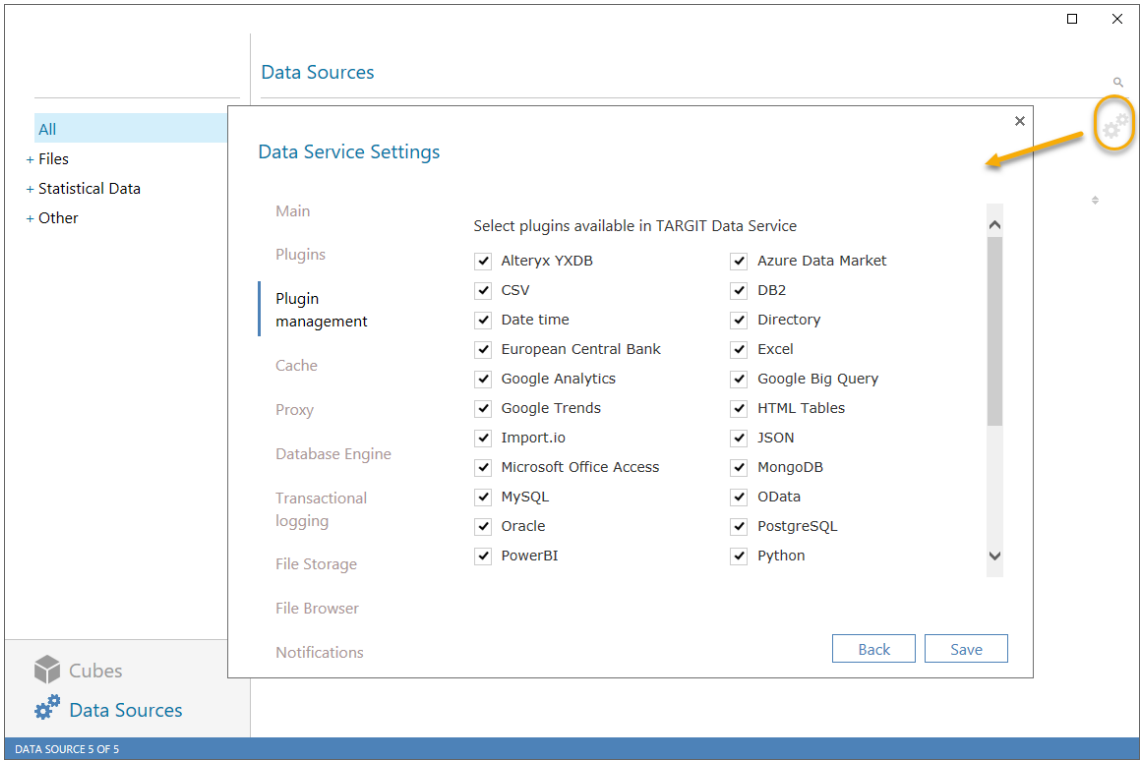
The Data Discovery Administrator has the power to:

- View and use all Data Sources and Cubes created by all users.
- Share and set permissions on all Data Sources and Cubes.
- Edit and Delete all Data Sources and Cubes.



Data Service Settings

Furthermore, the Administrator has rights to change some of the more advanced settings of the Data Service installation.



Exercises Lesson 8

No exercises available for this lesson.