MYOB EXO

Clarity User Guide

2019.4

Table of Contents

MYOB Exo Clarity

Exo Clarity is the MYOB Exo Business report writer especially written for use with Exo Business. The Report Designer is the application you will use to build Exo Business reports.

This manual assumes a basic understanding of databases, and a general understanding of the operation and table structure of Exo Business.

The Report Creation Process

The Exo Clarity interface is split into tabs, where each tab controls part of the report creation process.

The basic process of creating a report is:

1. Set up one or more data sources (or "data pipelines") for the report on the Data tab. This tab offers a Query Wizard, which guides you through the process of setting up a data source, and a Query Designer, which presents all options on a tabbed interface, so that users who are more familiar with Exo Clarity can set up a data source quickly.
2. As a more advanced step, complex queries and calculations can be set up on the Calc Tab. This requires a good degree of familiarity with Exo Business database structures and a knowledge of Delphi.
3. Build the report on the Design tab by adding report components to the main canvas, where they can be edited as necessary. A wide range of components are available, including simple graphical components and data-aware components that display data from one of the data sources you set up. A Report Wizard is available for setting up the basic layout of the report.
4. Check the report on the Preview tab to ensure that it's producing the results you want.
5. You can go back to the Data and Design tabs at any time to make changes as necessary. Once the report is finished, you can save it as one of the available report types.

Note: Video-based Help is available for this topic on the **Exo Business Education Centre**.

Exo Clarity Report Types

There are several kinds of report types in MYOB Exo Clarity - the functionality of each type differs depending on what is required of them.

- **Clarity Reports (.CLR)** - Most commonly used report type. Each CLR can have its own set of runtime parameters stored as a .CLS file and can be run from a menu, the command line, custom buttons, and so on.
- **Clarity Business Forms (.CLF)** - Specific Clarity Forms are identified with specific functions within MYOB Exo Clarity. These are essential business forms that are part of the usual business workflow. In contrast Clarity Reports (CLRs) can be placed on the MYOB Exo Business menu or run via the command line, and can in fact be given any name and there can be any number of them.

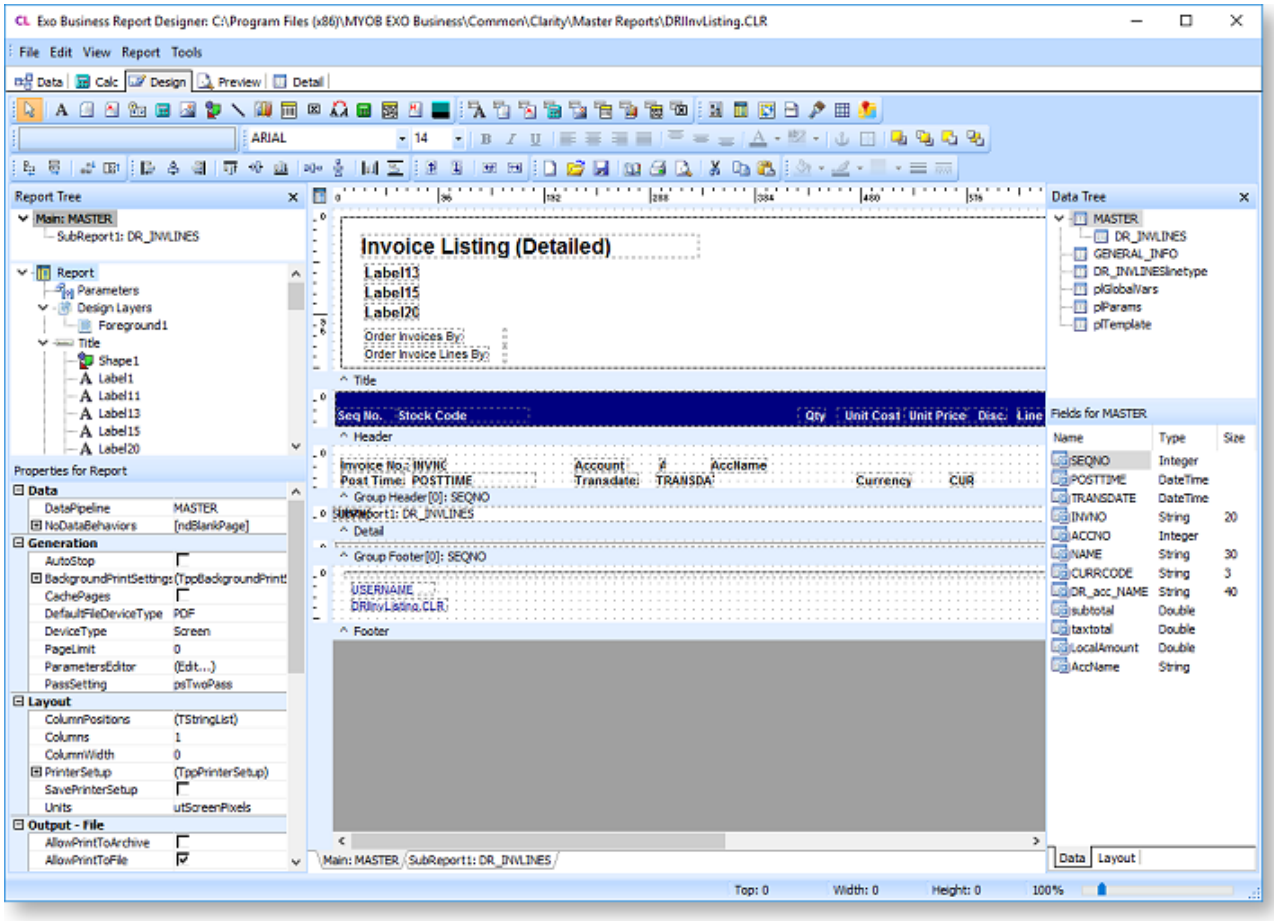
  A few notable exceptions exist however. Examples include where you are using a variety of invoice forms and they are specified by on the Debtor Account Details > Details 2 tab - Invoice file number – e.g. Invoice1.CLF, Invoice2.CLF - or if you have set up you system to prompt you to select from a list of formats when you save an invoice or sales order.

  Another characteristic of a CLF file is that it is automatically passed a single key value such as a transaction record number or an account number to the data pipeline named "Master" in your CLF definition. For example, Exo Business will pass the DR_Trans Seqno to the file named Invoice.CLF. Therefore the DR_Trans.Seqno field must be in the data pipeline named "Master" on the data tab in Clarity.

- **Clarity Merge Files (.CLM)** - CLM files are used to produce mail merge files.
- **Clarity Email Templates (.CLE)** - Clarity email templates are the HTML equivalent of a .CLM merge file for bulk-sent emails. They build the HTML that is inserted into the body of an email.

The Exo Clarity Interface

The Exo Clarity interface is divided into tabs. Each tab has its own menus and toolbars, containing functions specific to the tab.



Data Tab

The Data tab is where you select the database tables and fields that will be used by the report.

Calc Tab

The Calc tab lets you perform more sophisticated calculations on the report's data. It is an advanced feature, which requires knowledge of Delphi to use.

Design Tab

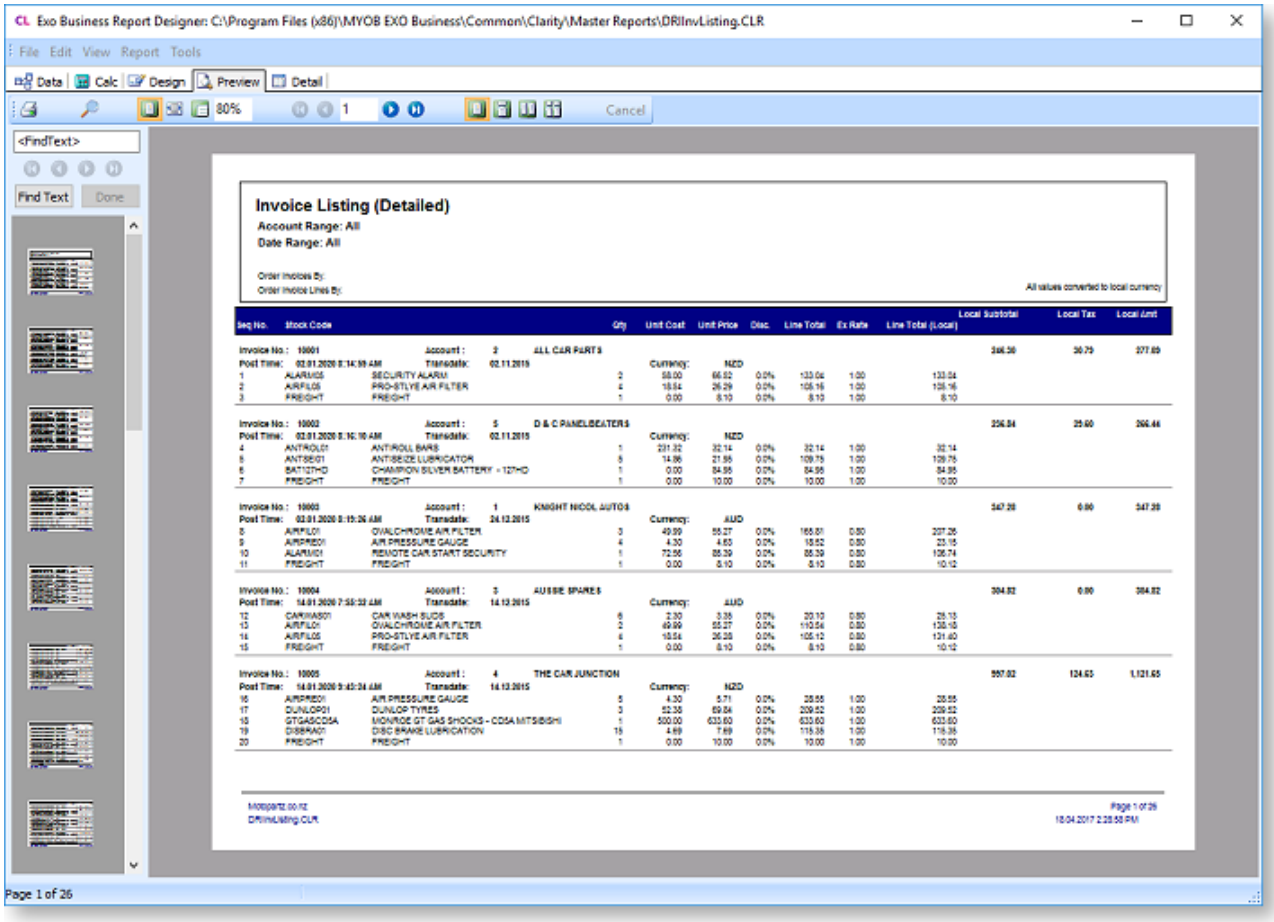
The Design tab is displayed when MYOB Exo Clarity is first opened. It is where the majority of the work of building the report is done.

Preview Tab

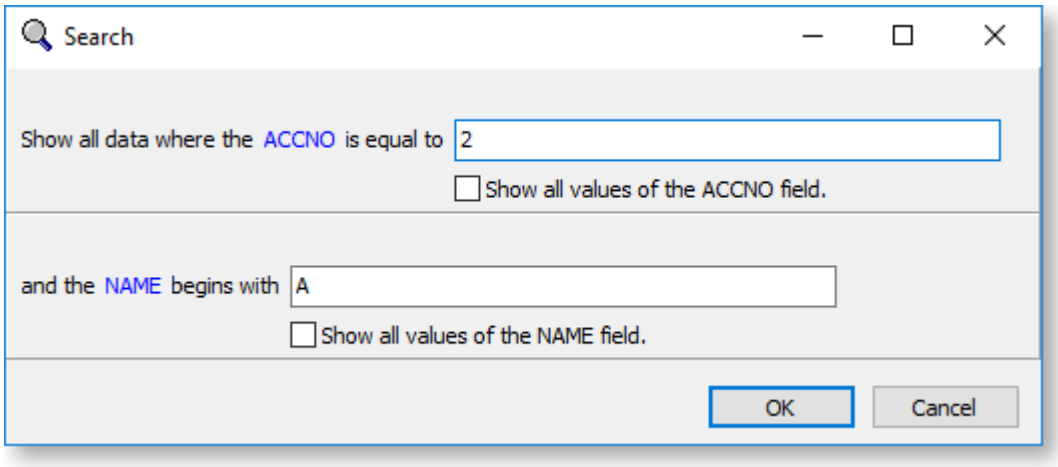
The Preview tab displays the report as it will look when printed. Referring to this tab regularly will let you see how the report is taking shape, and will highlight any changes or corrections that need to be made.

Toolbar buttons and thumbnails on this tab let you view each page of the report, zoom in to get a better look at a specific page and search for text in the report. You can browse on screen or you can print the report from the Preview tab.

Note: You can enable and configure the thumbnail and text search controls by editing the report's properties in the **Report Tree** - edit the **TextSearchSettings** and **ThumbnailSettings** properties under **User Interface**. You can also edit the text search settings by selecting **Find Text Settings** on the **Report menu of the Design tab**.



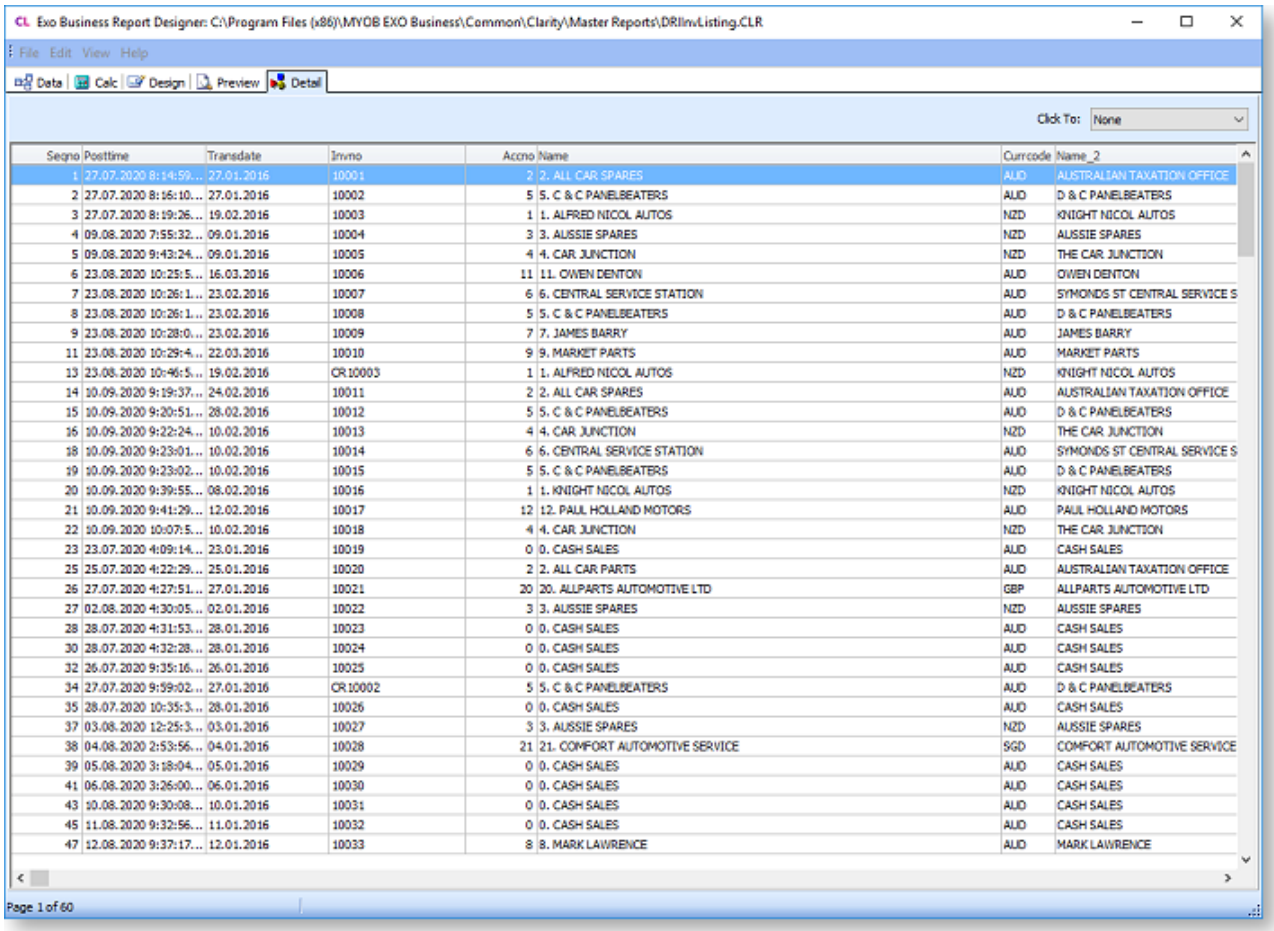
If the report has any search criteria with the **AutoSearch** option enabled, a Search button () is available on the Preview tab's toolbar. Clicking this button opens a window where you can filter the data displayed on the preview using the available search criteria:



Detail Tab

This tab displays the data from the data source that has been selected for the primary detail. You can use it to ensure that your report is returning the data you want it to.

Select an option from the **Click To** dropdown and double-click on a line to open the selected record for that line, e.g. in the report below, if you selected "Debtor Account" from the dropdown, then double-clicking on a line would open the Debtor Account Details window, showing the Debtor account from that line.



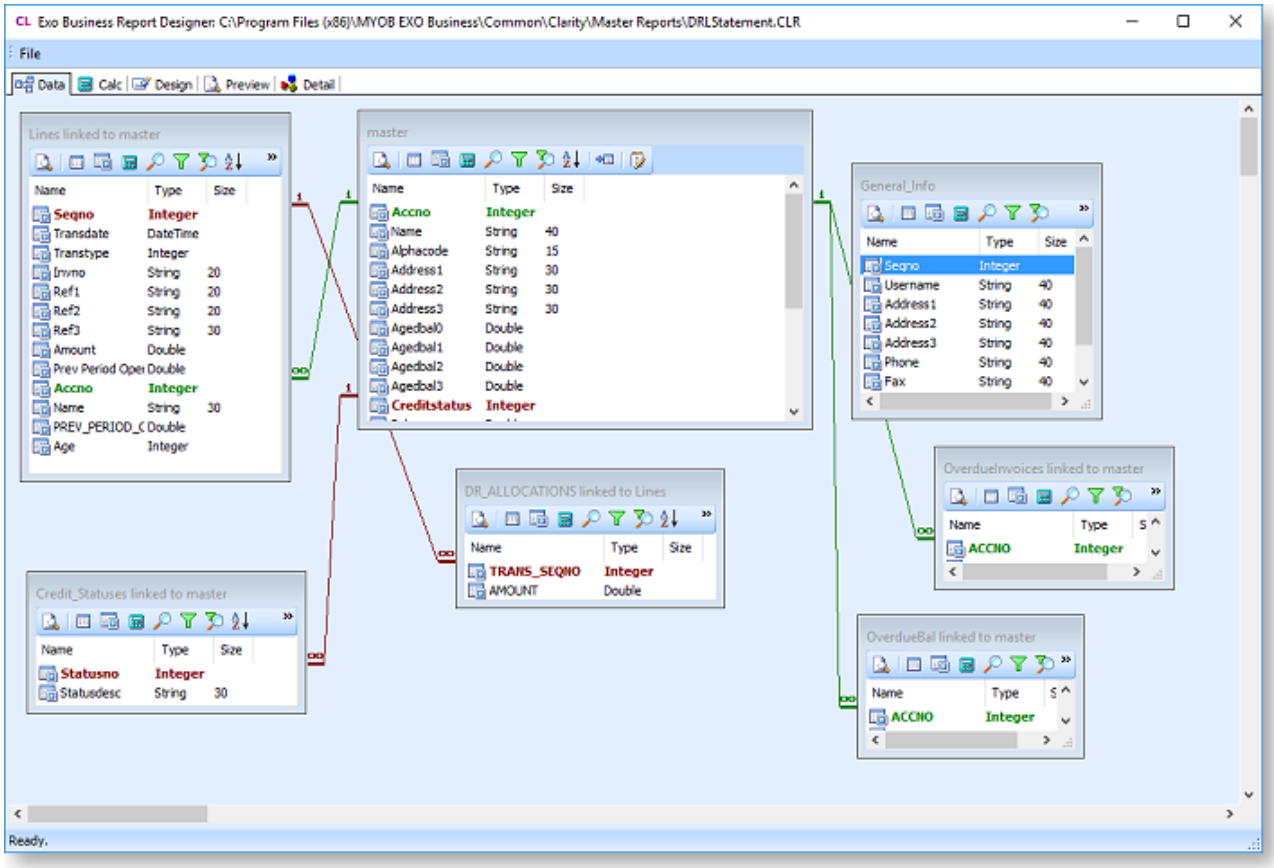
Segno	Posttime	Transdate	Invno	Accno	Name	Currcode	Name_2
1	27.07.2020 8:14:59...	27.01.2016	10001	2	2. ALL CAR SPARES	AUD	AUSTRALIAN TAXATION OFFICE
2	27.07.2020 8:16:10...	27.01.2016	10002	5	5. C & C PANELBEATERS	AUD	D & C PANELBEATERS
3	27.07.2020 8:19:26...	19.02.2016	10003	1	1. ALFRED NICOL AUTOS	NZD	KNIGHT NICOL AUTOS
4	09.08.2020 7:55:32...	09.01.2016	10004	3	3. AUSSIE SPARES	NZD	AUSSIE SPARES
5	09.08.2020 9:43:24...	09.01.2016	10005	4	4. CAR JUNCTION	NZD	THE CAR JUNCTION
6	23.08.2020 10:25:5...	16.03.2016	10006	11	11. OWEN DENTON	AUD	OWEN DENTON
7	23.08.2020 10:26:1...	23.02.2016	10007	6	6. CENTRAL SERVICE STATION	AUD	SYMONDS ST CENTRAL SERVICE S
8	23.08.2020 10:26:1...	23.02.2016	10008	5	5. C & C PANELBEATERS	AUD	D & C PANELBEATERS
9	23.08.2020 10:28:0...	23.02.2016	10009	7	7. JAMES BARRY	AUD	JAMES BARRY
11	23.08.2020 10:29:4...	22.03.2016	10010	9	9. MARKET PARTS	AUD	MARKET PARTS
13	23.08.2020 10:46:5...	19.02.2016	CR10003	1	1. ALFRED NICOL AUTOS	NZD	KNIGHT NICOL AUTOS
14	10.09.2020 9:19:37...	24.02.2016	10011	2	2. ALL CAR SPARES	AUD	AUSTRALIAN TAXATION OFFICE
15	10.09.2020 9:20:51...	28.02.2016	10012	5	5. C & C PANELBEATERS	AUD	D & C PANELBEATERS
16	10.09.2020 9:22:24...	10.02.2016	10013	4	4. CAR JUNCTION	NZD	THE CAR JUNCTION
18	10.09.2020 9:23:01...	10.02.2016	10014	6	6. CENTRAL SERVICE STATION	AUD	SYMONDS ST CENTRAL SERVICE S
19	10.09.2020 9:23:02...	10.02.2016	10015	5	5. C & C PANELBEATERS	AUD	D & C PANELBEATERS
20	10.09.2020 9:39:55...	08.02.2016	10016	1	1. KNIGHT NICOL AUTOS	NZD	KNIGHT NICOL AUTOS
21	10.09.2020 9:41:29...	12.02.2016	10017	12	12. PAUL HOLLAND MOTORS	AUD	PAUL HOLLAND MOTORS
22	10.09.2020 10:07:5...	10.02.2016	10018	4	4. CAR JUNCTION	NZD	THE CAR JUNCTION
23	23.07.2020 4:09:14...	23.01.2016	10019	0	0. CASH SALES	AUD	CASH SALES
25	25.07.2020 4:22:29...	25.01.2016	10020	2	2. ALL CAR PARTS	AUD	AUSTRALIAN TAXATION OFFICE
26	27.07.2020 4:27:51...	27.01.2016	10021	20	20. ALLPARTS AUTOMOTIVE LTD	GBP	ALLPARTS AUTOMOTIVE LTD
27	02.08.2020 4:30:05...	02.01.2016	10022	3	3. AUSSIE SPARES	NZD	AUSSIE SPARES
28	28.07.2020 4:31:53...	28.01.2016	10023	0	0. CASH SALES	AUD	CASH SALES
30	28.07.2020 4:32:28...	28.01.2016	10024	0	0. CASH SALES	AUD	CASH SALES
32	26.07.2020 9:35:16...	26.01.2016	10025	0	0. CASH SALES	AUD	CASH SALES
34	27.07.2020 9:59:02...	27.01.2016	CR10002	5	5. C & C PANELBEATERS	AUD	D & C PANELBEATERS
35	28.07.2020 10:35:3...	28.01.2016	10026	0	0. CASH SALES	AUD	CASH SALES
37	03.08.2020 12:25:3...	03.01.2016	10027	3	3. AUSSIE SPARES	NZD	AUSSIE SPARES
38	04.08.2020 2:53:56...	04.01.2016	10028	21	21. COMFORT AUTOMOTIVE SERVICE	SGD	COMFORT AUTOMOTIVE SERVICE
39	05.08.2020 3:18:04...	05.01.2016	10029	0	0. CASH SALES	AUD	CASH SALES
41	06.08.2020 3:26:00...	06.01.2016	10030	0	0. CASH SALES	AUD	CASH SALES
43	10.08.2020 9:30:08...	10.01.2016	10031	0	0. CASH SALES	AUD	CASH SALES
45	11.08.2020 9:32:56...	11.01.2016	10032	0	0. CASH SALES	AUD	CASH SALES
47	12.08.2020 9:37:17...	12.01.2016	10033	8	8. MARK LAWRENCE	AUD	MARK LAWRENCE

Data Tab

The Data tab is where you select the data sources (or "pipelines") that will be used by the report. Data sources consist of database tables and fields from the Exo Business database - when setting up a data source, you can select one or more tables and joining them where needed, creating a source that contains only the data required in your report. This gives the benefit of speed, as the subset of data is smaller and any heavy processing is performed on the server.

Each data source appears in a separate pane on the Data tab, with any links between sources appearing as lines between the panes. You can resize and reposition these panes as necessary to make the layout clearer to read.

You can access the Query Designer to edit a data source using the toolbar buttons at the top of each pane. To remove a data source from the Data tab, right-click on it and select **Delete**.

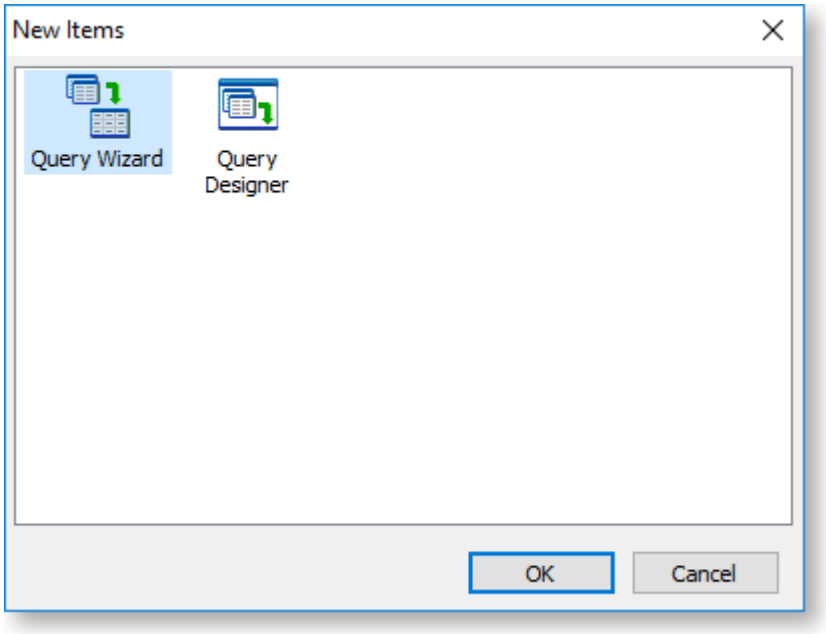


Data Tab Menus

The Data tab has one menu: File. This menu provides the following options:

New

Select this option to create a new data source, using the Query Wizard or the Query Designer.



Close

Closes the Exo Clarity application.

Import

Imports saved data source settings from a .DTM file into the Data tab, replacing the current setup.

Merge

Imports saved data source settings from a .DTM file into the Data tab, adding them to the current setup.

Export

Exports the current setup of the Data tab to a .DTM file. This file can be imported or merged into another report, to save you having to set up the same data source multiple times.

Data Source Toolbars

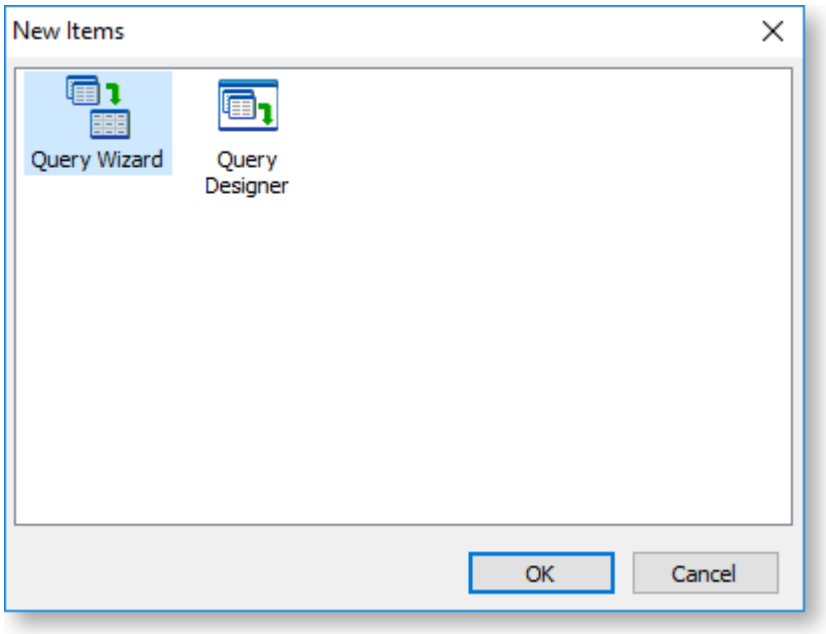
Each data source pane has a toolbar at the top, which let you edit the data source's properties.

Button	Name	Description
	**Preview**	Opens a window showing the data that will be selected by the data source. **Note:** You can also open the preview window by holding SHIFT and clicking anywhere in the data source's pane.
	**Tables**	Opens the Tables tab of the Query Designer.
	**Fields**	Opens the Fields tab of the Query Designer.
	**Calcs**	Opens the Calcs tab of the Query Designer.
	**Search**	Opens the Search tab of the Query Designer.

	Group	Opens the Group tab of the Query Designer.
	Group Search	Opens the Group Search tab of the Query Designer.
	Sort	Opens the Sort tab of the Query Designer.
	Link	Opens the Link window, where you can set up links between data sources.
	SQL	Opens the SQL tab of the Query Designer.

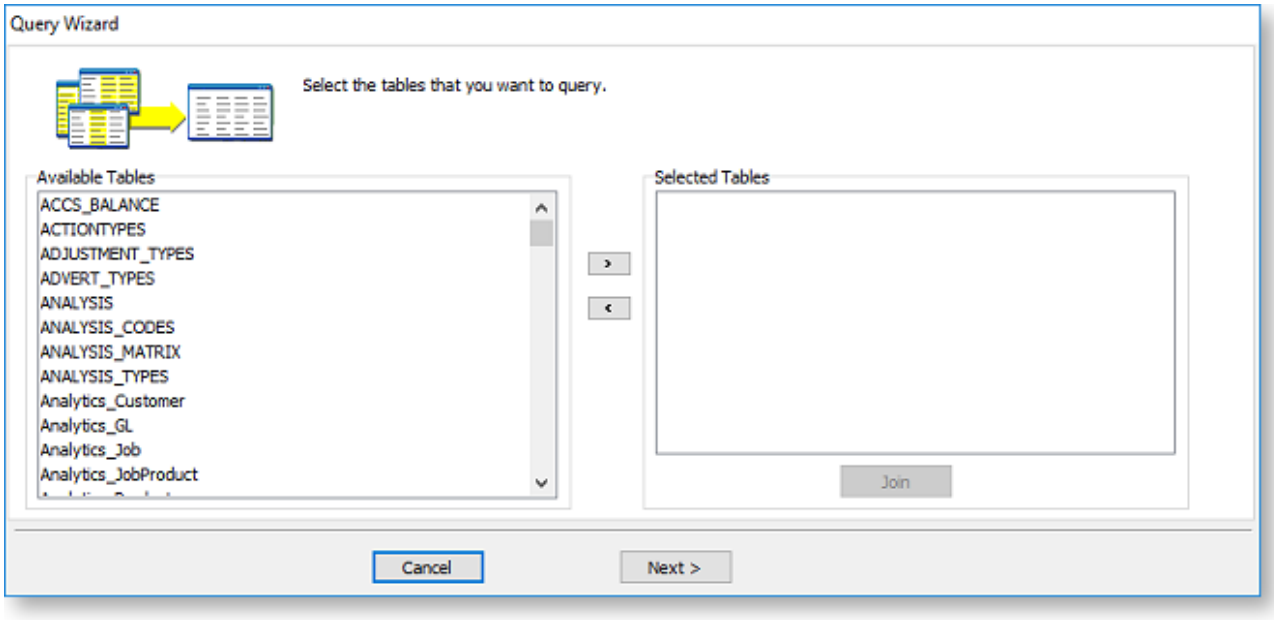
The Query Wizard

The Query Wizard steps you through the process of setting up a new data source. To create a new data source with the Query Wizard, select **New** from the File menu on the Data tab, then select **Query Wizard** from the window that appears.



Select Tables

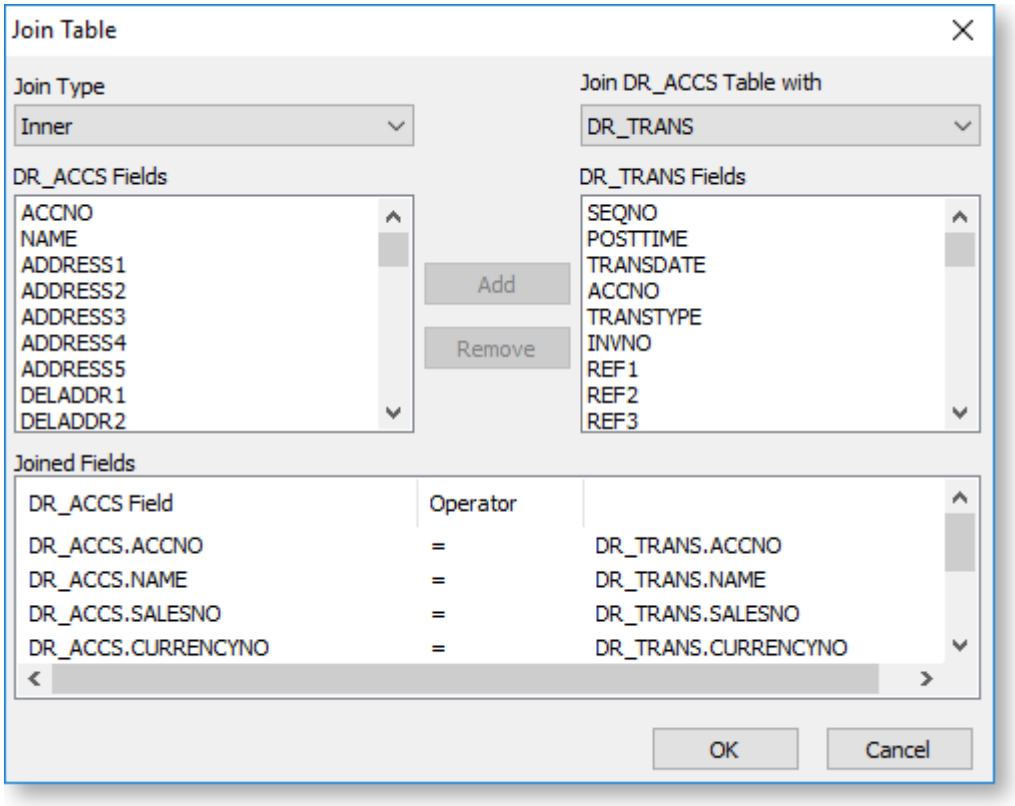
The first screen of the Query Wizard lets you select the tables to include in the data source:



To add a table, double-click on it in the Available Tables pane, or select it and click the left arrow button. The table appears in the Selected Tables pane.

Note: Any **views** set up in the Exo Business database are also available for section here.

When you add a second or subsequent table to the Selected Tables section, the Join Table window appears so you can create a join between the new table and one of the existing tables:



To create the join:

1. Select the **Join Type**: Inner, Left Outer, Right Outer, or Full Outer.
2. Use the **Join Table with** dropdown to select the table that the new table will be joined to. Only tables that are already part of the data source are available in the dropdown.

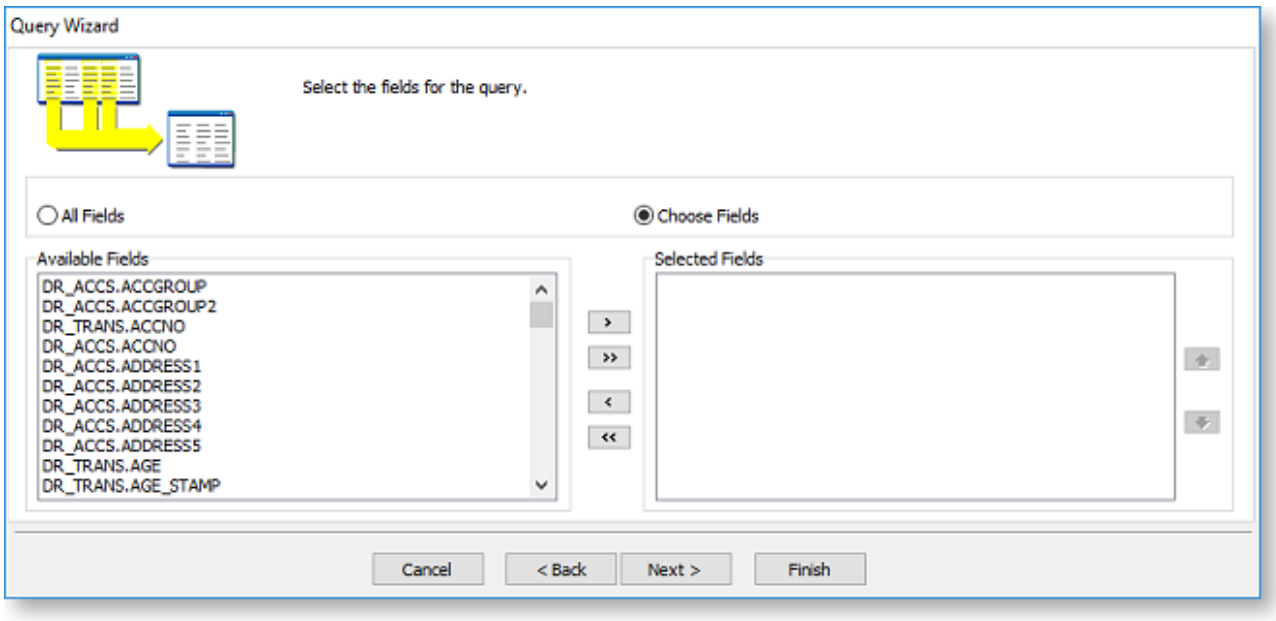
3. Exo Clarity automatically adds joins between fields that have the same name in both tables. You can then add or remove joins as necessary:
    - To add a new join, select a field from the Fields box on the left, select the field to join it to from the Fields box on the right, then click **Add**.
    - To remove a join, highlight it in the Joined Fields section and click **Remove**.
4. Click **OK** to create the join and close the Join Table window.

You can edit the join by selecting the joined table in the Selected Tables section and clicking the **Join** button.

Once you have added all tables, click **Next** to go to the next wizard screen.

Select Fields

The next wizard screen lets you select which fields from the table(s) you selected on the previous screen should be included in the data source. By including only a subset of the available fields, you can reduce the amount of data that has to be processed when the report is run.



Either select the **All Fields** option, or select **Choose Fields**, then move fields between the Available Fields and Selected Fields sections using the following methods:

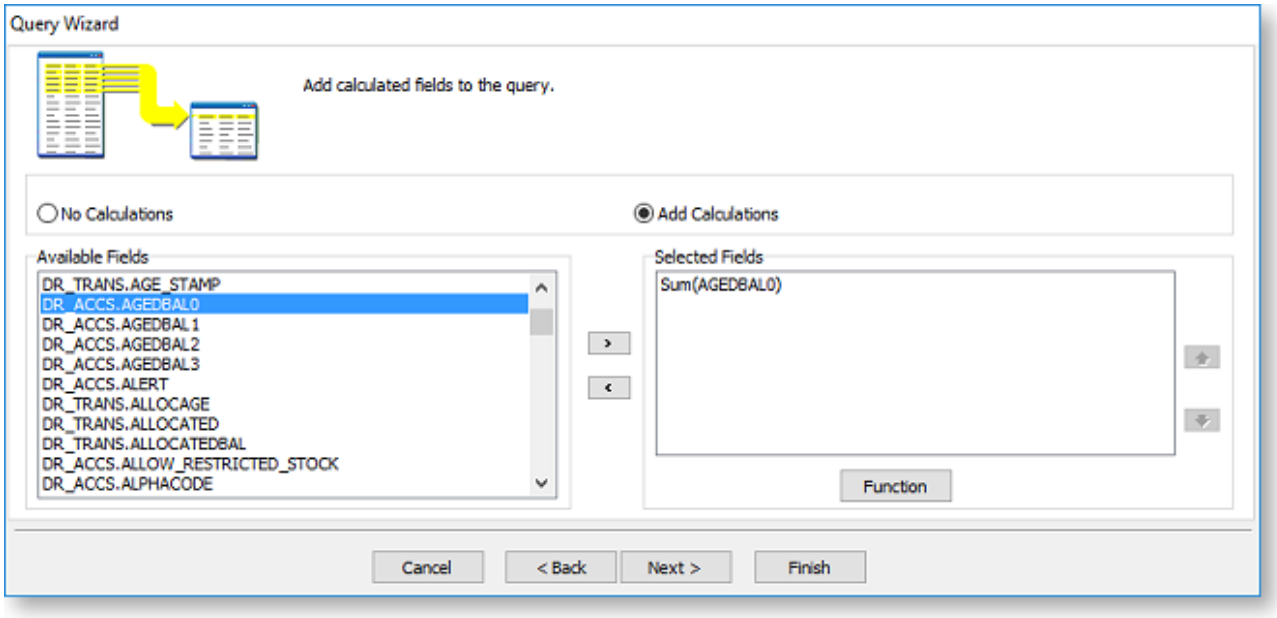
- Double-click on a field in the Available Fields section to move it to the Selected Fields section (and vice versa).
- Select one or more fields, then drag them from one section to the other. You can select multiple fields using the CTRL and SHIFT keys.
- Select one or more fields, then click the left and right arrow buttons to move them from one section to the other.
- Click the double arrow buttons to move all fields from one section to the other.

Once you have added all fields, click **Next** to go to the next wizard screen.

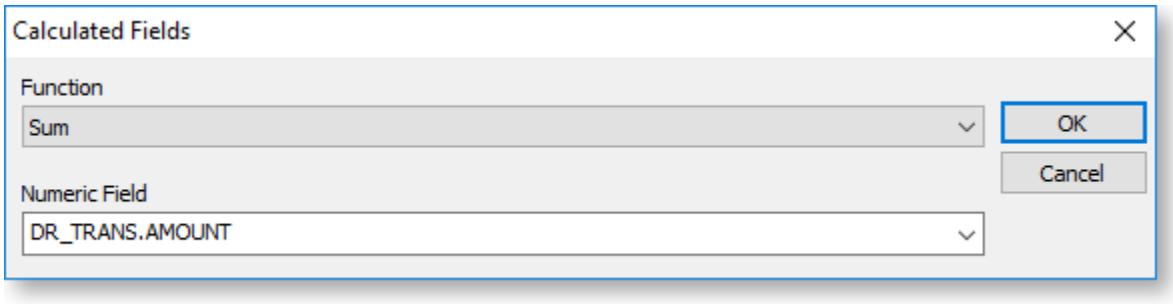
Add Calculations

The next wizard screen lets you add fields that perform basic mathematical calculations on the data.

Note: Advanced calculations are set up on the **Exo Clarity Calc tab**.



To add a calculation, select the Add Calculations option, then add a field to the Selected Fields section by double-clicking on it or selecting it and using the left arrow button. When a field is added, the Calculated Fields window opens, which lets you set up the calculation:



Select a calculation function to perform from the **Functions** dropdown. The following functions are available on this tab:

- Avg - Average the values in the selected field.
- Count - Return a count of the values in the selected field.
- Max - Return the maximum value out of the values in the selected field.
- Min - Return the maximum value out of the values in the selected field.
- Sum - Sum the values in the selected field.

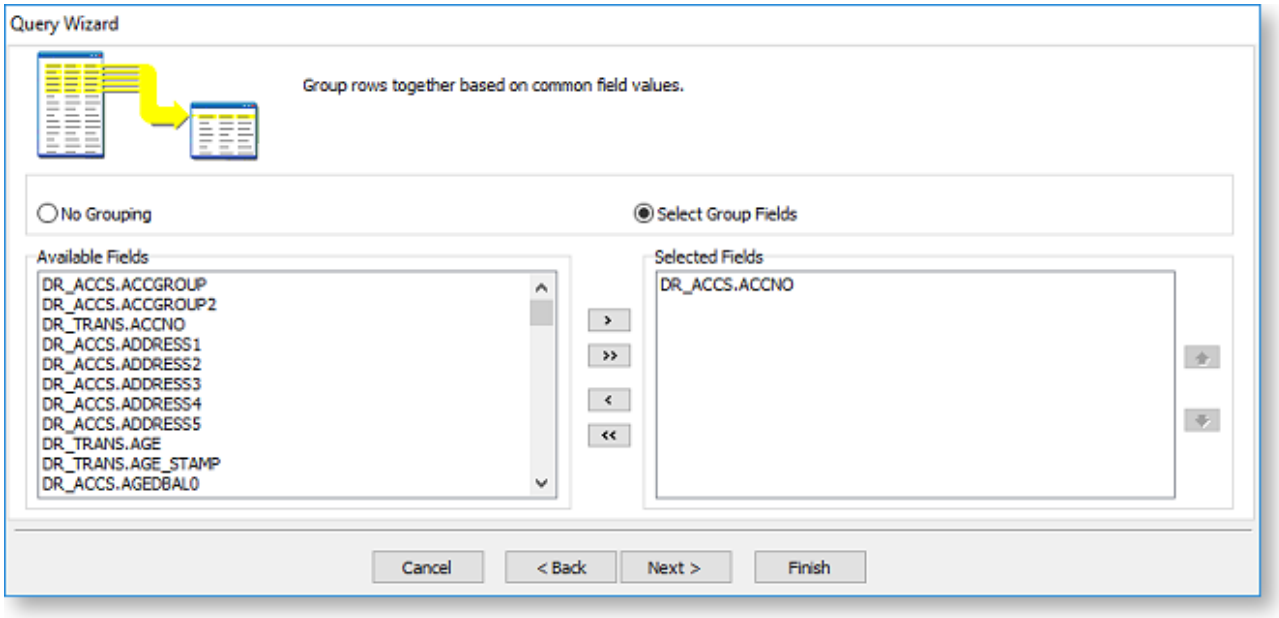
You can change the field that the calculation is based on by selecting a different one from the **Numeric Field** dropdown. Click **OK** to add the calculated field.

Once you have finished adding calculated fields, click **Next** to go to the next wizard screen.

Add Groups

The next wizard screen lets you group the data in the data source by common field values.

Note: More advanced grouping options are available on the **Groups window**, which is available from the **Design tab**.



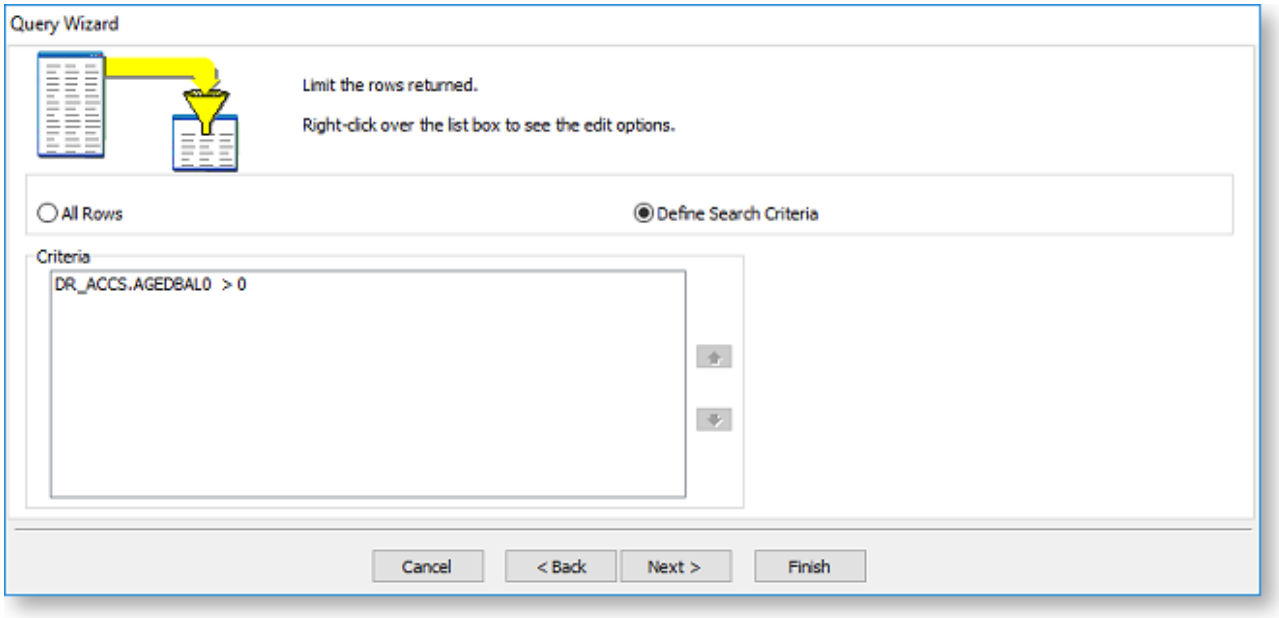
Either select the **No Grouping** option, or select **Select Group Fields**, then move fields between the Available Fields and Selected Fields sections using the following methods:

- Double-click on a field in the Available Fields section to move it to the Selected Fields section (and vice versa).
- Select one or more fields, then drag them from one section to the other. You can select multiple fields using the CTRL and SHIFT keys.
- Select one or more fields, then click the left and right arrow buttons to move them from one section to the other.
- Click the double arrow buttons to move all fields from one section to the other.

Once you have finished adding groups, click **Next** to go to the next wizard screen.

Add Search Criteria

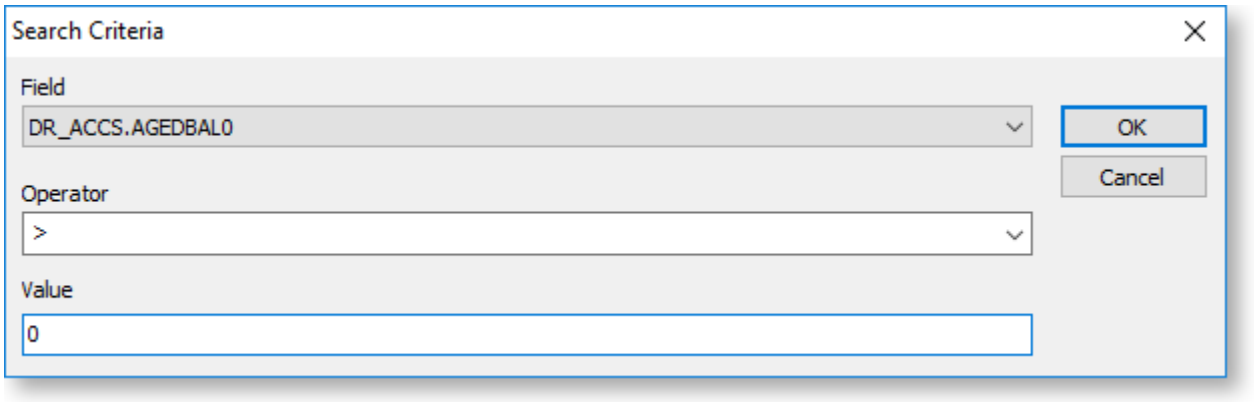
The next wizard screen lets you perform searches on the data to narrow down the selected data so the data set is kept as small as possible.



To add search criteria, select the **Define Search Criteria** option, then right-click on the Criteria section to add search criteria. The right-click menu offers the following options:

- Insert Criteria - adds new search criteria
- Insert OR - adds an OR operator between criteria
- Insert NOT - adds a NOT operator before the selected criteria
- Edit Criteria - edits the details of the selected criteria
- Remove - deletes the selected criteria
- Add Parentheses - adds "begin" and "end" statements around the selected criteria, so that you can group criteria together when they are evaluated.
- Remove Parentheses - removes any parentheses from around the selected criteria.

Selecting **Insert Criteria** or **Edit Criteria** opens the Search Criteria window, where you can set up the search:



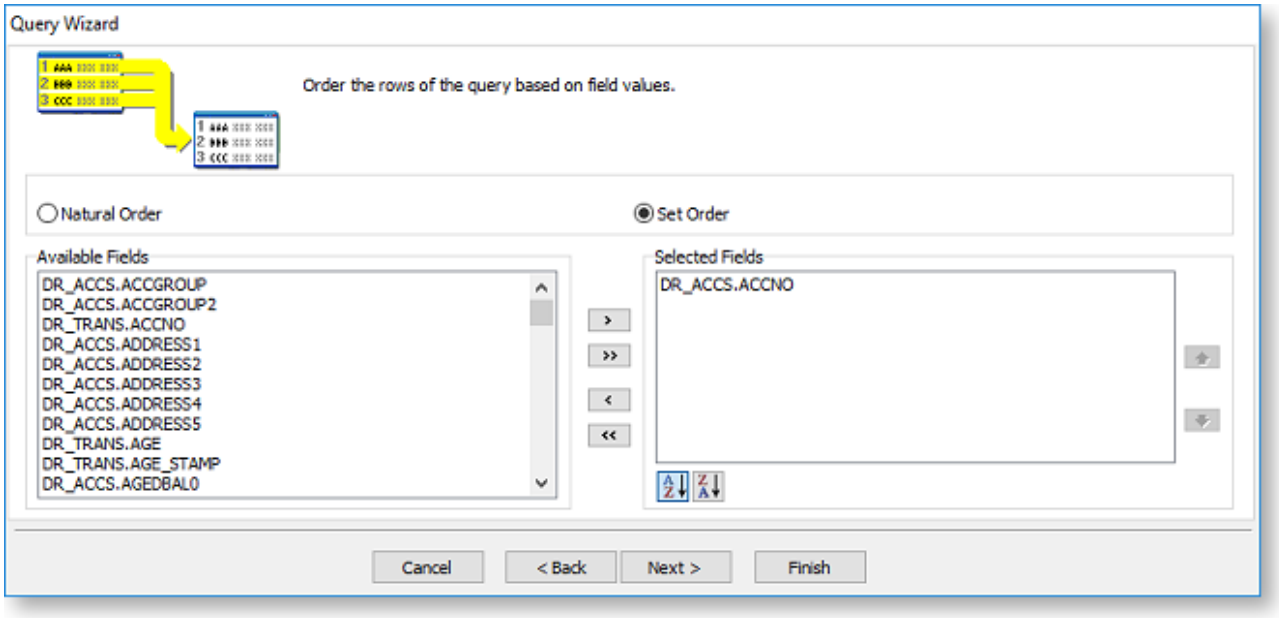
To set up the search:

1. Select the field to search on from the **Field** dropdown.
2. Select the search operator, e.g. less than, greater than or equal to, not equal to, from the **Operator** dropdown.
3. Enter the search value into the **Value** field.
4. Click **OK**.

Once you have finished adding search criteria, click **Next** to go to the next wizard screen.

Add Sorting

The next wizard screen lets you control how the data in the data source will be sorted.



Either select the **Natural Order** option, or select **Set Order**, then move fields between the Available Fields and Selected Fields sections using the following methods:

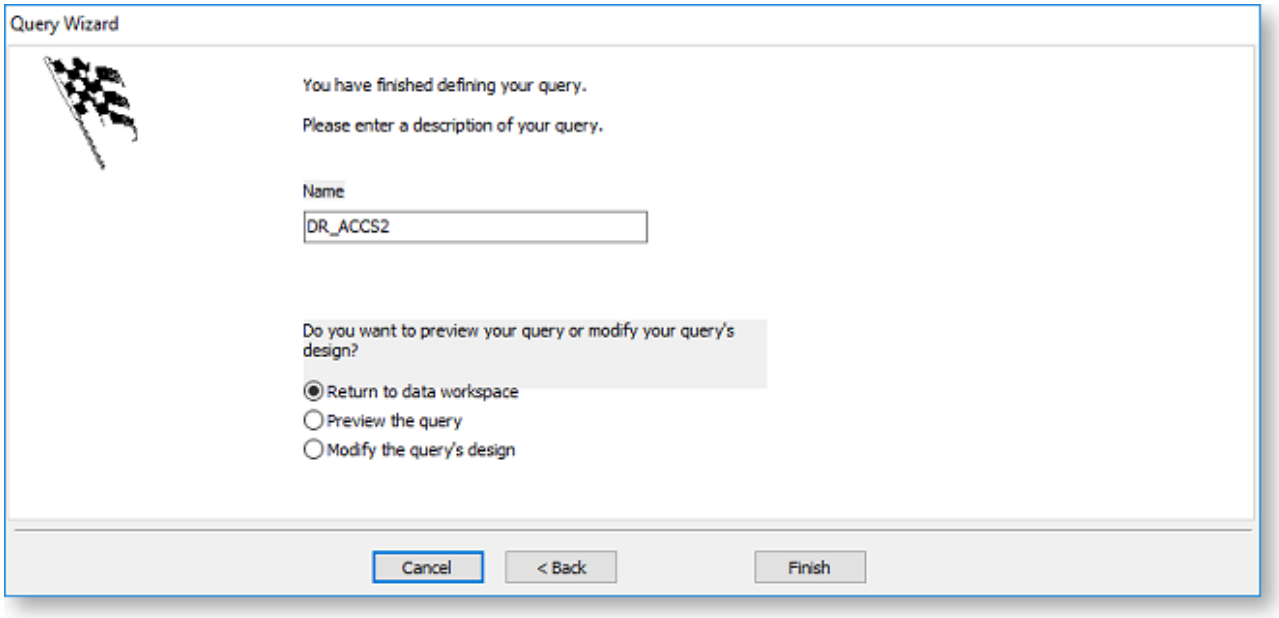
- Double-click on a field in the Available Fields section to move it to the Selected Fields section (and vice versa).
- Select one or more fields, then drag them from one section to the other. You can select multiple fields using the CTRL and SHIFT keys.
- Select one or more fields, then click the left and right arrow buttons to move them from one section to the other.
- Click the double arrow buttons to move all fields from one section to the other.

Use the  buttons to specify whether data should be sorted in ascending or descending order for each sort field.

Once you have finished adding sorting, click **Next** to go to the next wizard screen.

Name the Data Source and Finish

The final wizard screen lets you enter a name for the new data source.



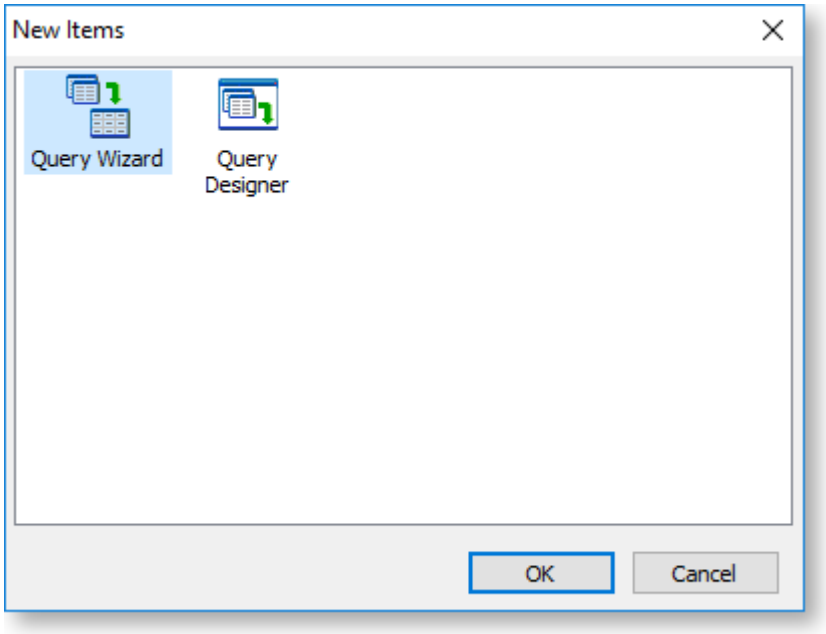
Click **Finish** to create the new data source and perform one of the following actions:

- Return to data workspace - closes the wizard and returns to the Data tab.
- Preview the query - closes the wizard and displays a preview window showing the data that will be selected by the new data source.
- Modify the query's design - closes the wizard and opens the new data source in the Query Designer, where you can edit it further.

Once a data source has been added, it appears in its own pane on the Data tab. You can access the Query Designer to edit the data source using the toolbar buttons at the top of each pane. To remove a data source from the Data tab, right-click on it and select **Delete**.

The Query Designer

The Query Designer lets you set up a new data source. Unlike the Query Wizard, the Query Designer present all options to you from the start, providing a quicker way for an advanced user to create data sources. To create a new data source with the Query Designer, select **New** from the File menu on the Data tab, then select **Query Designer** from the window that appears.

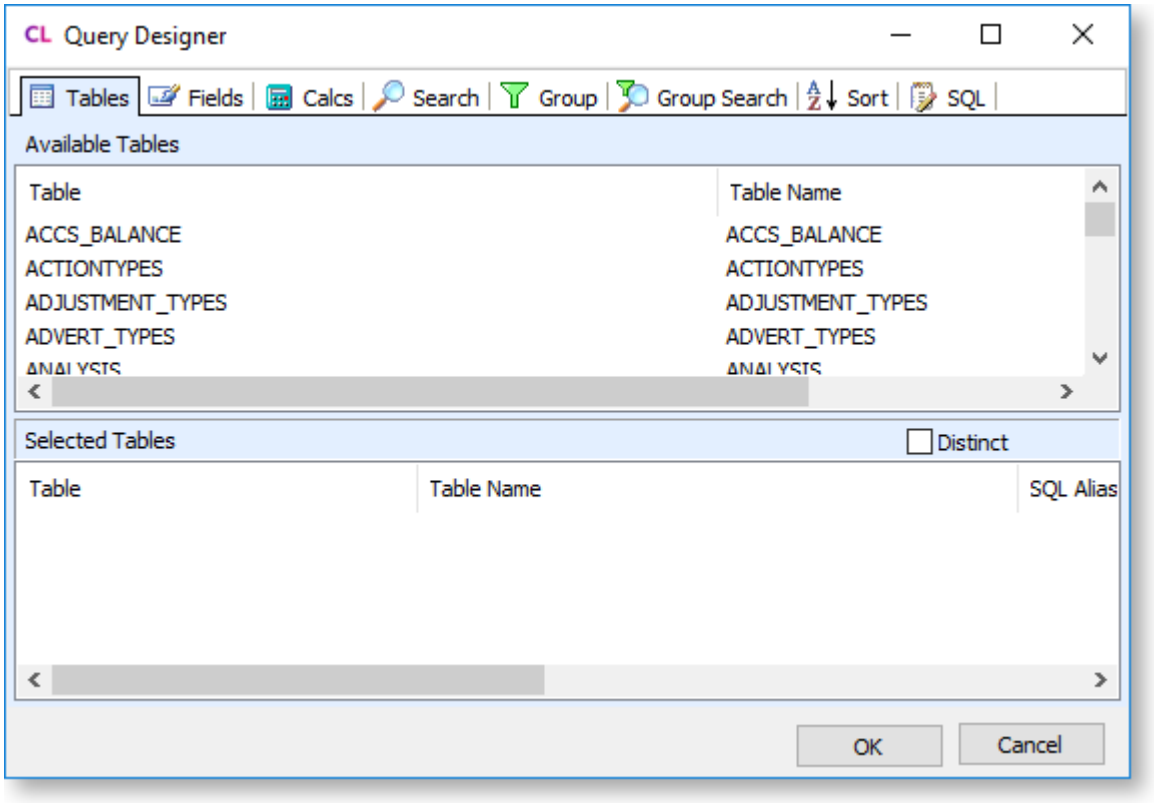


The Query Designer presents options for setting up a data source on a single window, divided up into tabs:

- Tables
- Fields
- Calcs
- Search
- Group
- Group Search
- Sort
- SQL

Once a data source has been added, it appears in its own pane on the Data tab. You can access the Query Designer tabs to edit the data source using the toolbar buttons at the top of each pane. To remove a data source from the Data tab, right-click on it and select **Delete**.

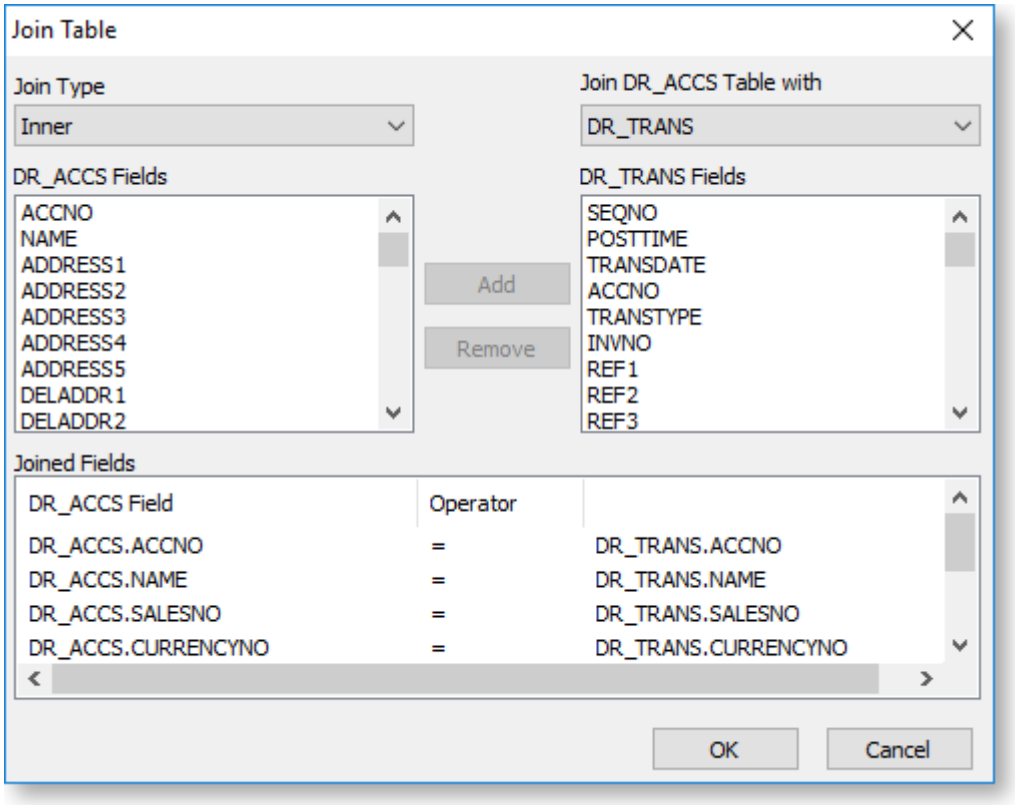
Tables Tab



This tab specifies the tables that will be included in the data source. The Available Tables section lists all tables in the Exo Business database. Double-click on a table to add it to the Selected Tables section. (You can remove a table from the Selected Tables section by double-clicking on it again.)

Note: Any **views** set up in the Exo Business database are also available for section here.

When you add a second or subsequent table to the Selected Tables section, the Join Table window appears so you can create a join between the new table and one of the existing tables:

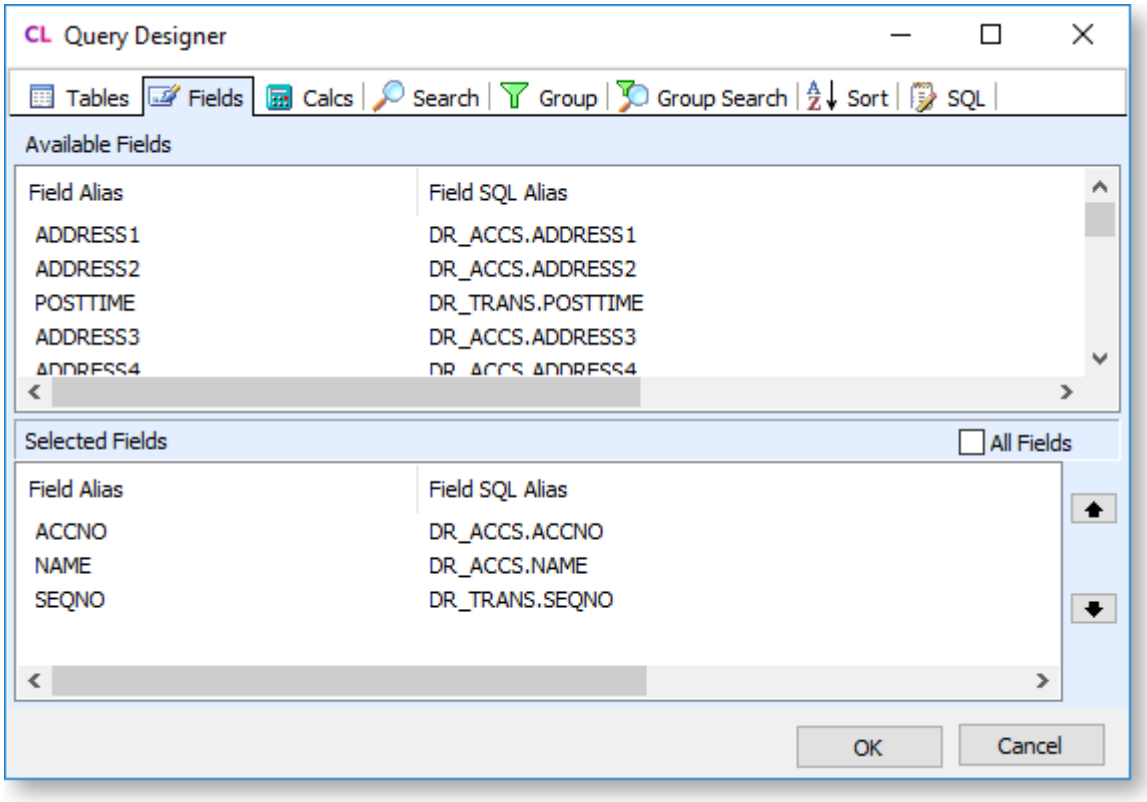


To create the join:

1. Select the **Join Type**: Inner, Left Outer, Right Outer, or Full Outer.
2. Use the **Join Table with** dropdown to select the table that the new table will be joined to. Only tables that are already part of the data source are available in the dropdown.
3. Exo Clarity automatically adds joins between fields that have the same name in both tables. You can then add or remove joins as necessary:
    - To add a new join, select a field from the Fields box on the left, select the field to join it to from the Fields box on the right, then click **Add**.
    - To remove a join, highlight it in the Joined Fields section and click **Remove**.
4. Click **OK** to create the join and close the Join Table window.

Fields Tab

This tab specifies which fields from the table(s) you selected on the Tables tab should be included in the data source. By including only a subset of the available fields, you can reduce the amount of data that has to be processed when the report is run. Ideally, you should include only those fields you need to run the report - you can always return to this tab later if you need to add more fields.



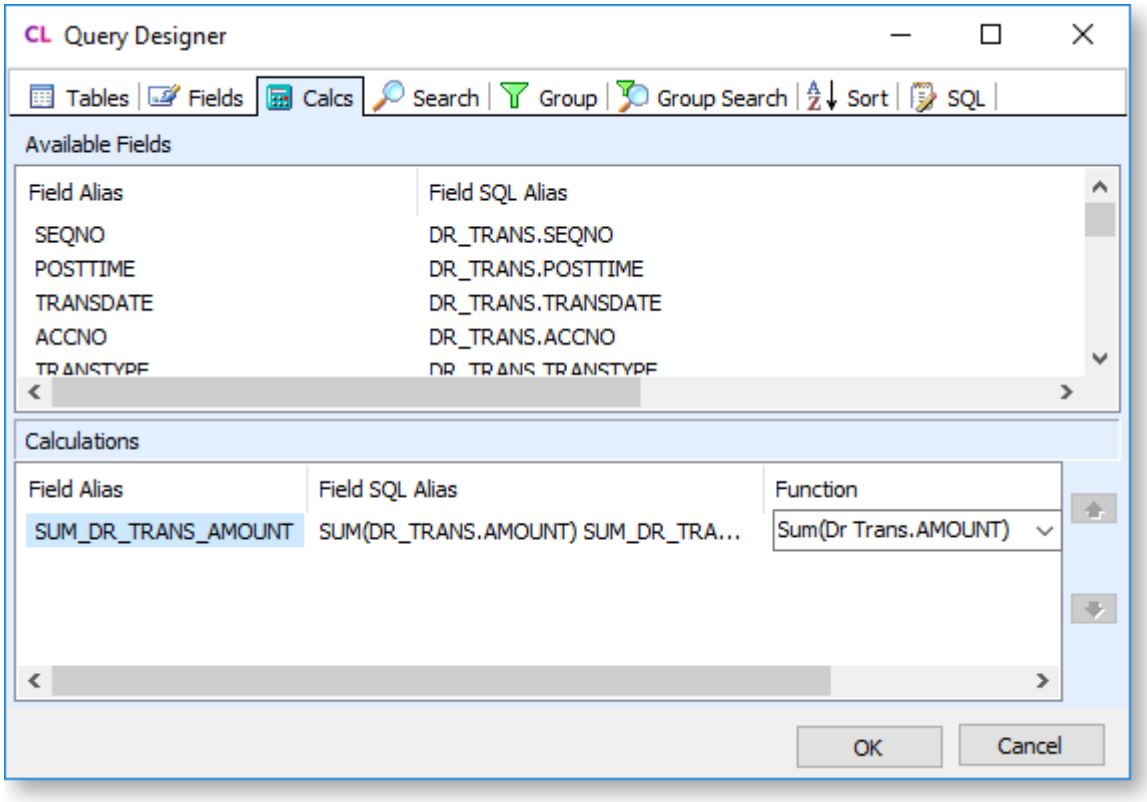
The Available Fields section lists all fields from the selected table(s). Move fields between this section and the Selected Fields section using the following methods:

- Double-click on a field in the Available Fields section to move it to the Selected Fields section (and vice versa).
- Select one or more fields, then drag them from one section to the other. You can select multiple fields using the CTRL and SHIFT keys.
- Tick the **All Fields** check box to move all fields into the Selected Fields section. Clear it to move all fields back to the Available Fields section.

You can click on a selected field's **Field Alias** to rename it (or right-click and select **Rename**).

Calcs Tab

This tab lets you add calculated fields to the data source. These fields perform simple mathematical calculations on the data - advanced calculations are set up on the Exo Clarity Calc tab.



To add calculations, double-click on the field to perform the calculation on, or select multiple fields using CTRL or SHIFT, then drag them into the Calculations section. For each field, select a calculation function from the dropdown in the **Function** column. The following functions are available on this tab:

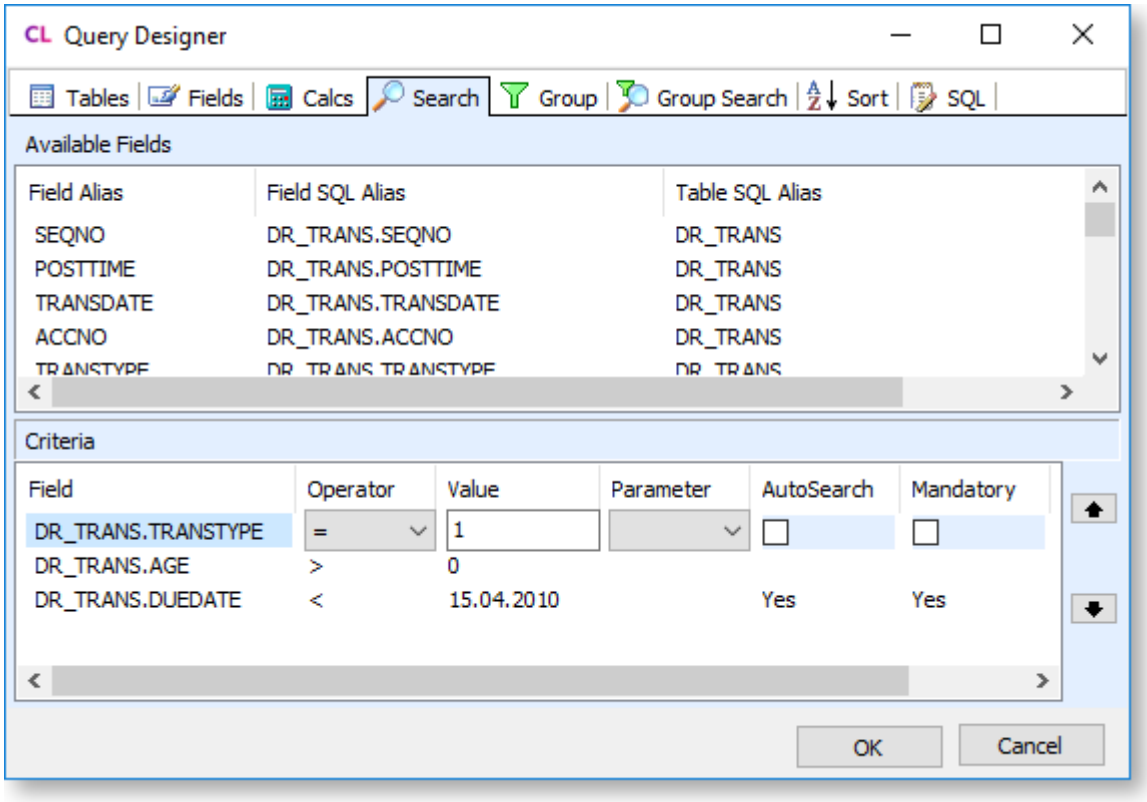
- Avg() - Average the values in the selected field.
- Count() - Return a count of the values in the selected field.
- Max() - Return the maximum value out of the values in the selected field.
- Min() - Return the maximum value out of the values in the selected field.
- Sum() - Sum the values in the selected field.
- Expression - Enter the formula for a more complex calculation manually. You can type the formula directly into the **Function** field.

An alias for the calculation is automatically generated based on the function - you can click in the **Field Alias** column to enter a new alias.

Search Tab

This tab lets you narrow down the selection of data for the report to keep the data set as small as possible. You can set restrictions like only including transactions later than a certain date, or only accounts with a balance greater than zero. Searching adds a WHERE clause to the SQL statement that retrieves the data.

Note: When working on a new report, you can use this tab to restrict the sample size to a small number so that the report runs faster while you are testing it. Remember to remove this restriction once the report is complete and ready for use on a live site.



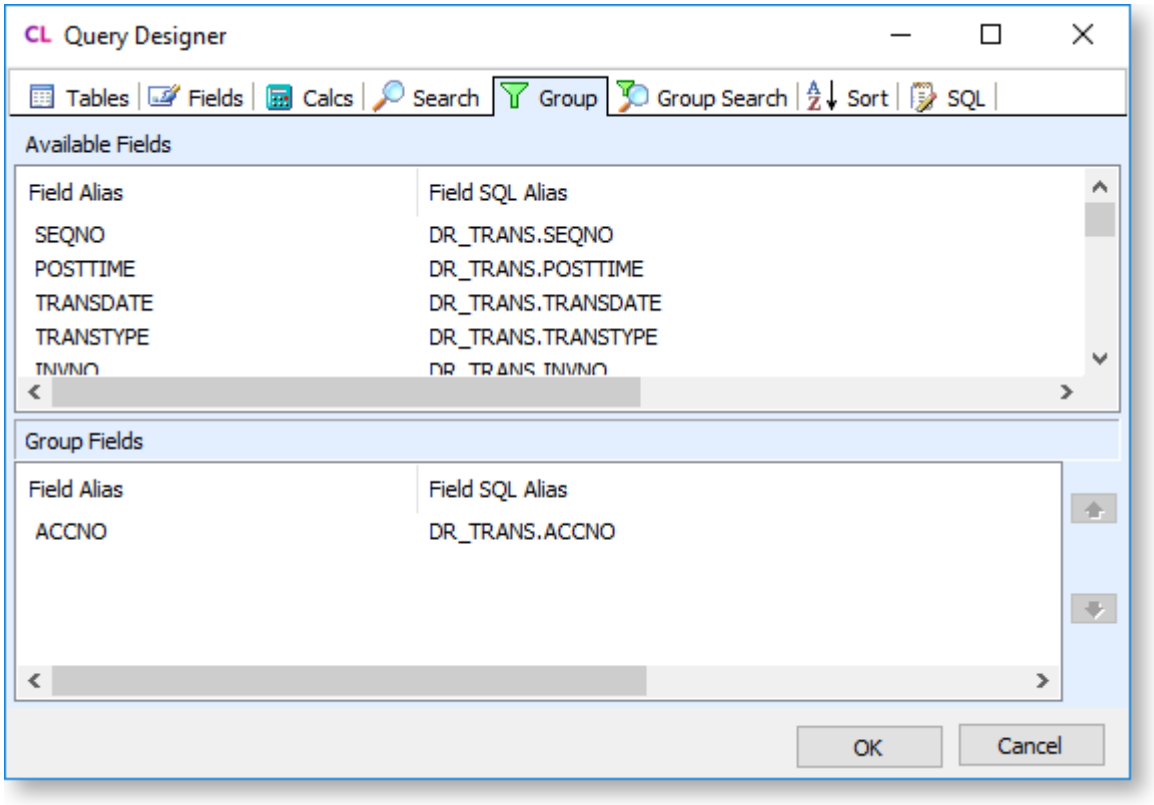
To add a search, double-click on the field to search on, or select multiple fields using CTRL or SHIFT, then drag them into the Criteria section. For each field in the Criteria section, set the search conditions using the controls in each column. You can set up more advanced search criteria using the options in the right-click menu, which include options to add NOT or OR conditions, and to add parentheses around groups of criteria.

Any search conditions that have the **AutoSearch** column set to "Yes" will appear in the Preview tab's Search window.

Group Tab

On this tab, you can select fields to group the data by. Grouping adds a GROUP BY clause to the SQL statement that retrieves the data - this is usually done when using calculated fields (set up on the Calcs tab) to aggregate data.

> **Note:** As well as grouping data at the data source, you can also display data in groups on the report by using the **Groups window**, which is available from the **Design tab**.

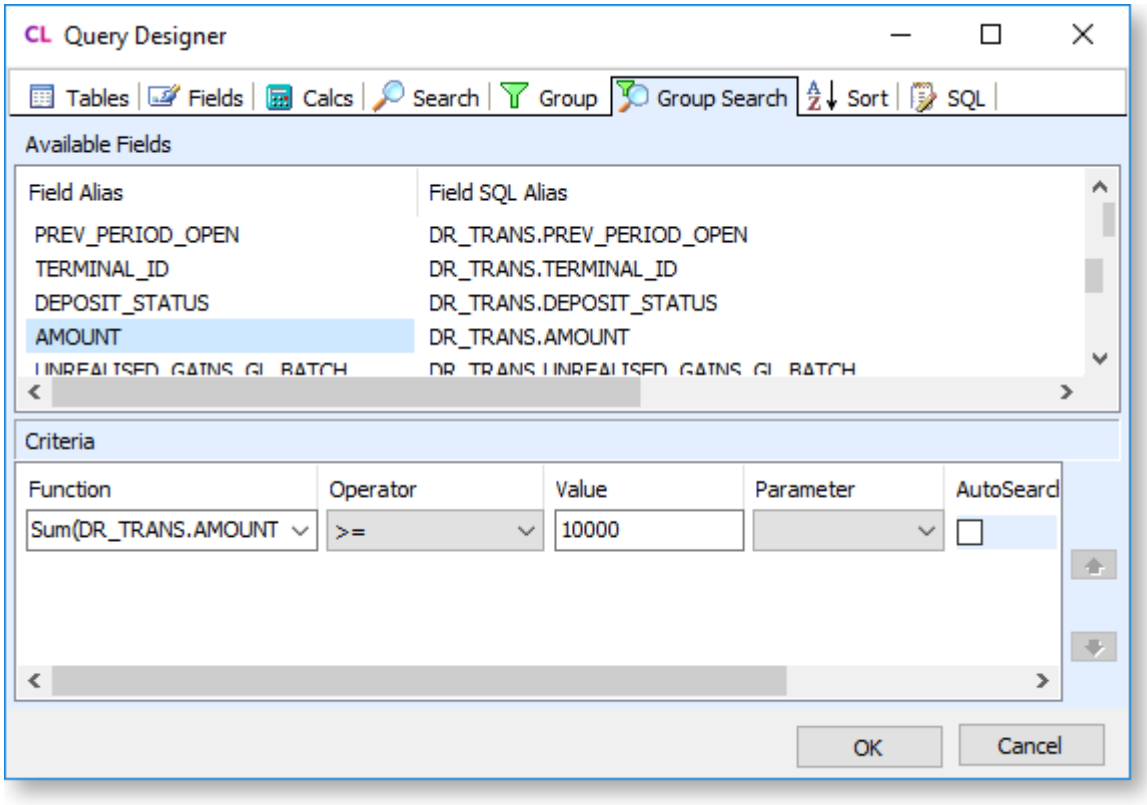


To add a grouping, double-click on the field to group by, or select multiple fields using CTRL or SHIFT, then drag them into the Group Fields section.

Group Search Tab

On this tab, you narrow down the selection of data for the report by searching grouped data.

Note: This is equivalent to using a HAVING clause in SQL, where a standard search uses a WHERE clause. In the screenshot below, data from the DR_TRANS table is being grouped by ACCNO - the search will restrict the data to accounts whose transactions total to $10,000 or more.

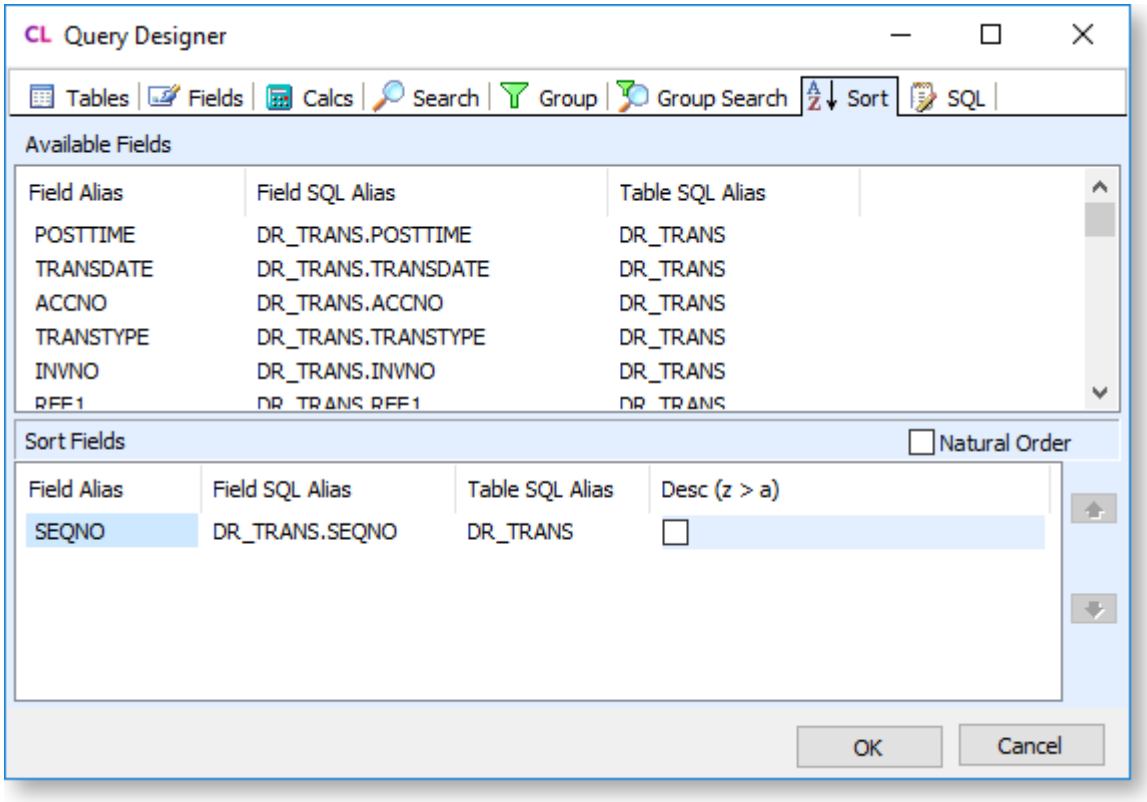


If one or more groups have been specified on the Group tab, this table will contain all available fields in the selected database table.

To add a search, double-click on the field to search on, or select multiple fields using CTRL or SHIFT, then drag them into the Criteria section. For each field in the Criteria section, set the search conditions using the controls in each column. You can set up more advanced search criteria using the options in the right-click menu, which include options to add NOT or OR conditions, and to add parentheses around groups of criteria.

Sort Tab

This tab lets you specify how the data should be sorted. This adds an ORDER BY clause to the SQL statement that retrieves the data.



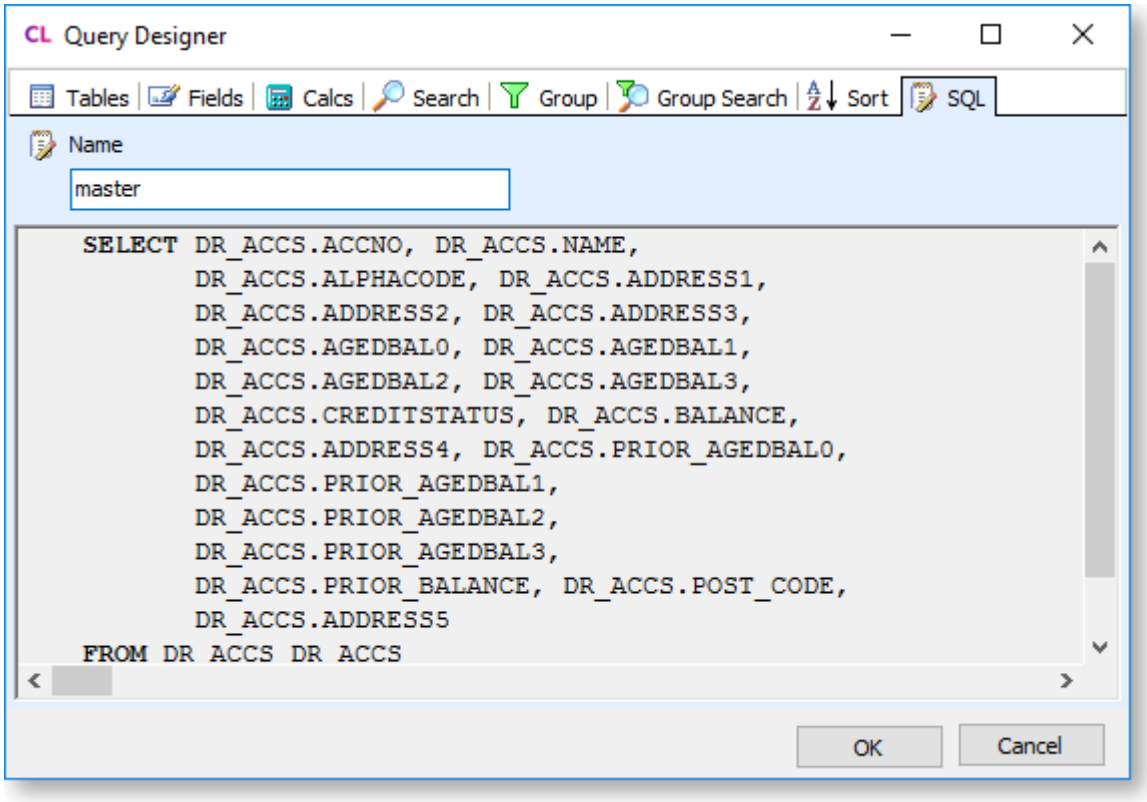
To add a sort field, double-click on the field to sort by, or select multiple fields using CTRL or SHIFT, then drag them into the Sort Fields section. Arrange the fields into the correct sort order using the arrow buttons on the right of the window. You can tick the box in the **Desc (z > a)** column to determine whether records should be sorted in descending (ticked) or ascending (unticked) order.

Ticking **Natural Order** clears the Sort Fields section, removing all sorting.

> **Note:** If you are grouping fields using the **Groups window**, it is essential that the fields that you sort by are the same ones that you group by and in the same order, or your report will not make sense. Once these fields have been added in the correct order, any further fields can be added to sort records within the group. For example, in order to correctly group records in a report by Debtor account you must have the records sorted by Debtor account. You could then add a sort field for the invoice number to sort by invoice number within the Debtor account.

SQL Tab

This tab displays the selections you have made on the other tabs as SQL statements. If you are familiar with SQL, this tab provides a useful check to see that you have made the selections you intended.



By default, the only editable part of the tab is the **Name** field, which you can use to give a new name to the data source. You can edit the SQL directly by right-clicking on the SQL statements and selecting **Edit SQL**. By editing the SQL manually, you can enter any SQL statement you want; for example, you could enter a query to take data from a function or stored procedure.

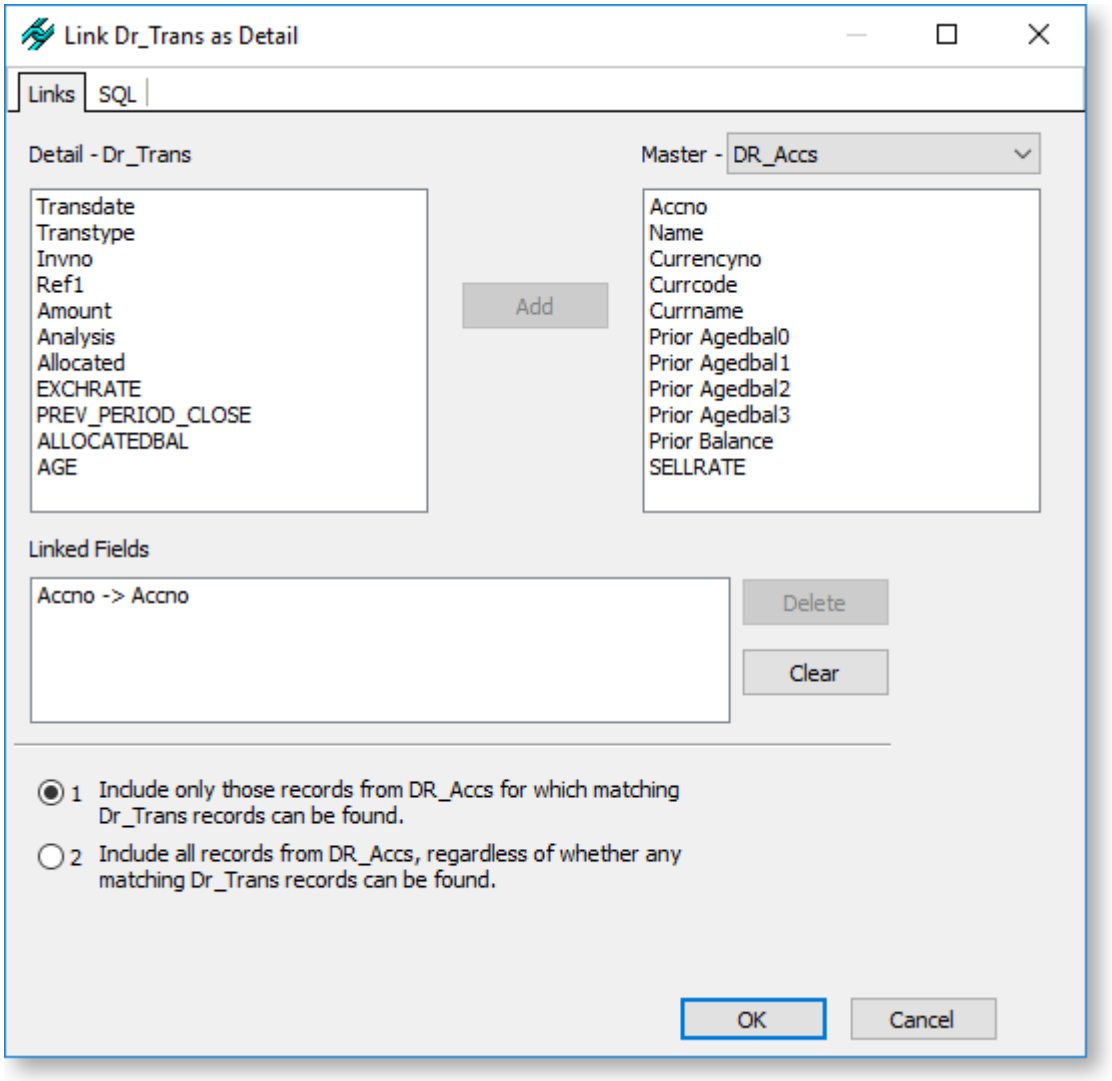
Note: Once you choose edit the SQL manually, this cannot be undone; the query can only be edited manually from now on. All of the other tabs on the Query Designer window are hidden.

Linking Fields

It is possible to create master-detail field links between data sources on the Data tab, which can be useful when setting up a report with sub reports.

You can set up a new link between data sources in one of the following ways:

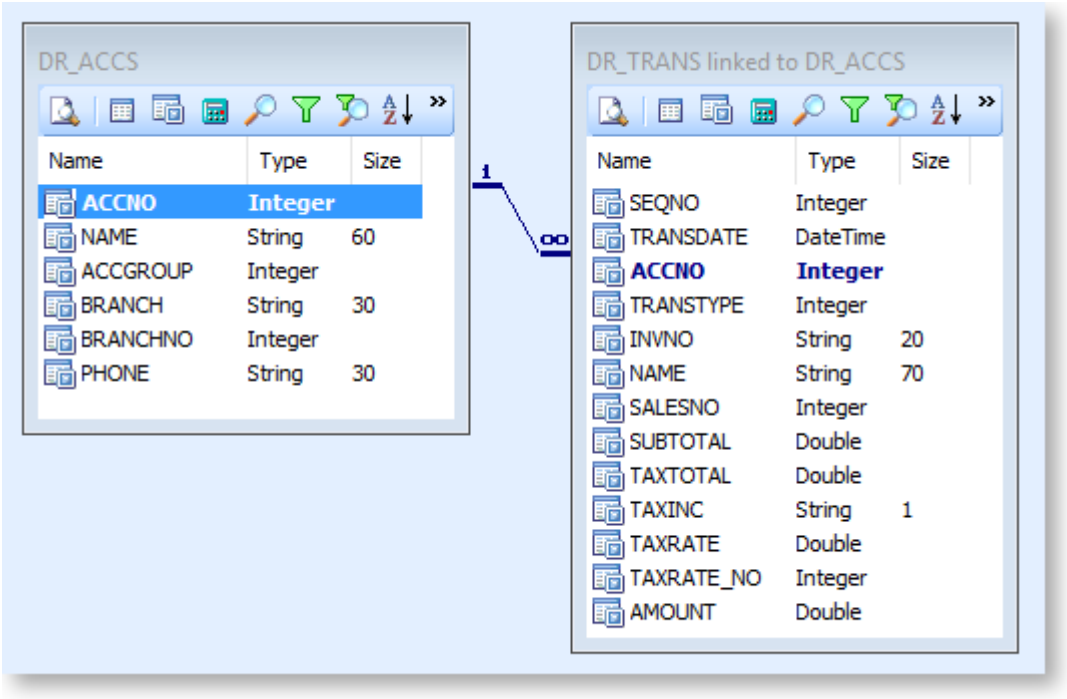
- Drag the field you want to link from the detail data source to the master data source and drop it on the field you want to link it to.
- Click the Link button ( ) on the toolbar of the detail data source to open the Link window:



The fields in the destination data source are displayed on the left. Select the master data source that should be linked to this source from the **Master** dropdown, and the fields in that source appear on the right. Any fields with the same name in both sources are automatically linked and appear in the Linked Fields section below - you can edit the links between the sources in several ways:

- Select a field in the Detail section and a field in the Master section, then click **Add** to link the two fields. The new link appears in the Linked Fields section.
- Select a link in the Linked Fields section and click **Delete** to remove that link.
- Click **Clear** to remove all links between the source.

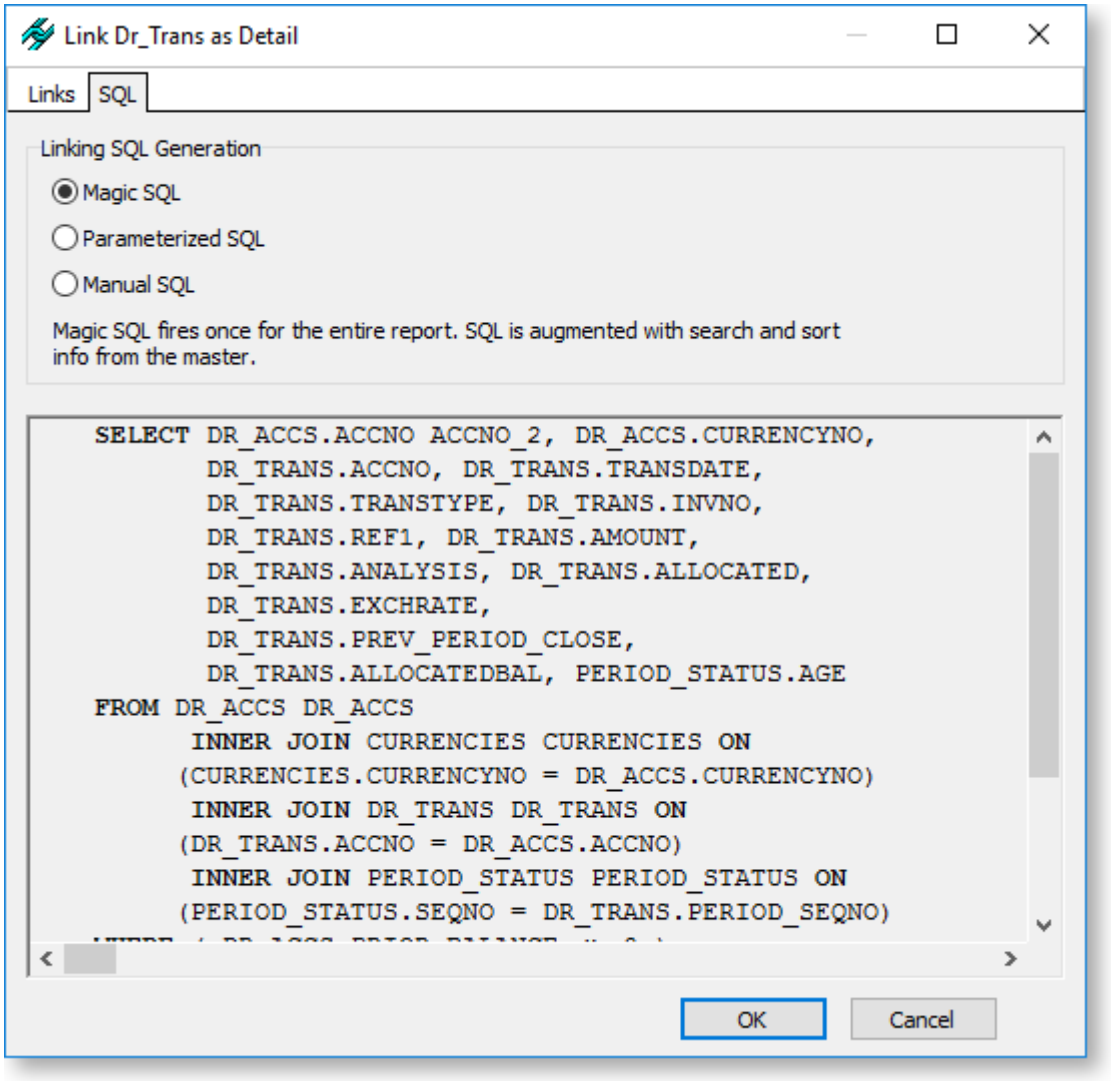
Links are displayed on the Data tab as lines between tables and the title of the detail data source updates to show that it is linked to the master:



You can double-click on a line to open the Link window and edit the links between the sources.

SQL Tab

The SQL tab of the Link window shows the linking SQL statements that are generated for the data source.



Note: You can hold CTRL and click on a data source's pane on the Data tab to view the linking SQL in a separate window.

Views

When selecting tables for a data source using the Query Wizard or Query Designer, you also have the option of selecting one or more of the views that have been defined in the Exo Business database.

A view is a virtual table in the database; the table does not actually exist in the database, so does not take up any space other than for the code describing it. When it includes calculations, it is a lot faster than doing the calculations in the report itself. If your data source requires a complex SQL statement that performs slowly when the report is run, it may be better to create a view in the Exo Business database that exposes the data you need and use that view as your data source.

Views can also provide a limited means of security, because they can provide users access to a subset of available data while hiding other related and sensitive data.

Fact Views

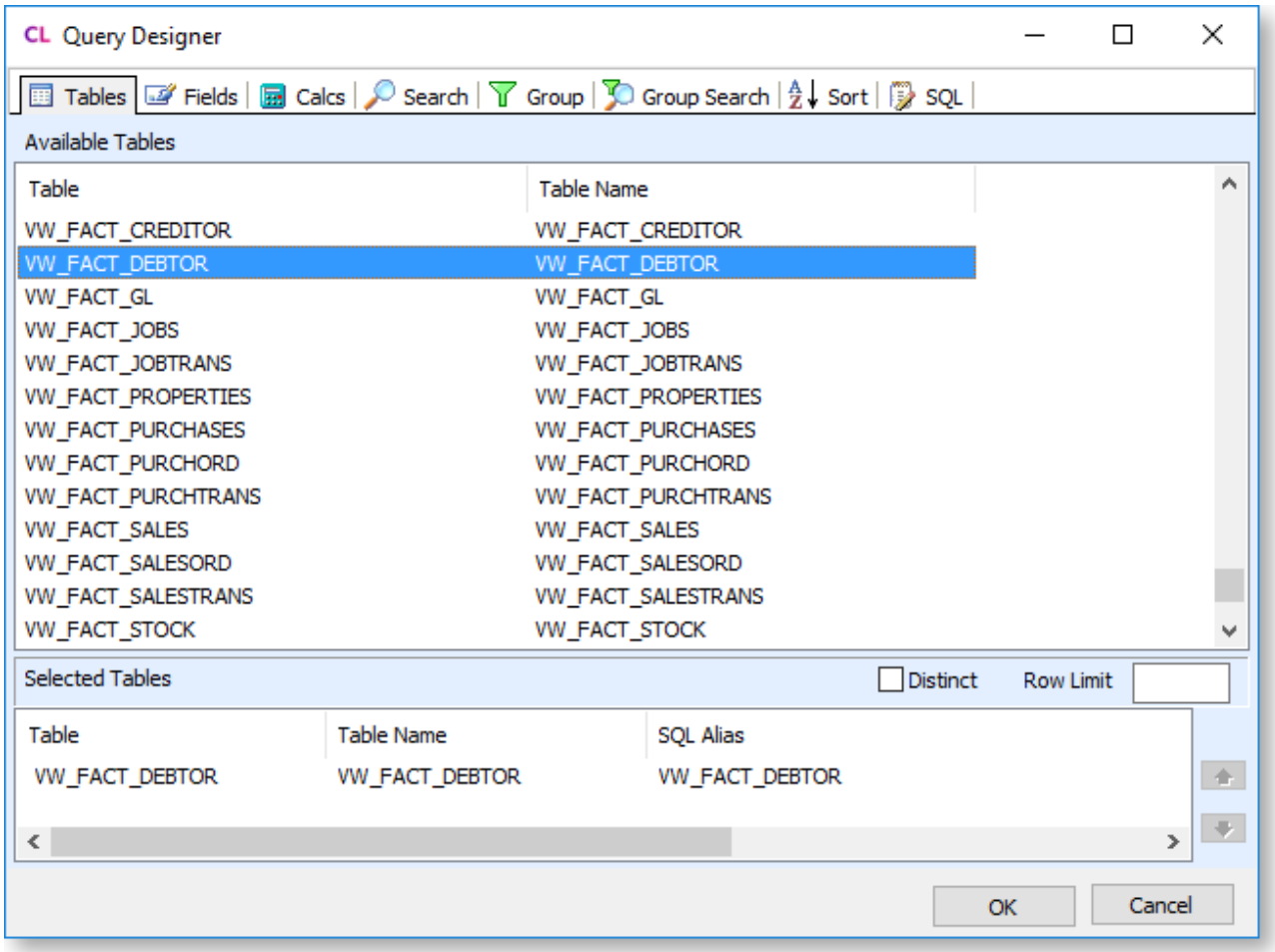
Over twenty pre-made data views are available in the Exo Business database, providing comprehensive, dependable, user-friendly abstract views of all major data objects. These "fact views" encapsulate the associated complexities of the underlying database and present simple pre-formatted flat-file facts requiring no join logic, making them useful in reporting, particularly in pivot table ad-hoc style reporting.

Two "layers" of fact views have been set up in the Exo Business database. The views that make up the first layer are identifiable in the Exo Business database by their names, which all start with "VW_FACT". These views draw in all data relevant to each area of the Exo Business. For example, the VW_FACT_SALESTRANS view, which contains data on sales transactions, includes three columns for the ACCNAME: ACCNAME_ENTERED, ACCNAME_SALES, and ACCNAME_BRANCH. This captures all cases where an account name can be associated with a transaction.

Note: Examine the VW_FACT_PROPERTIES view for schema information on all of the other fact views.

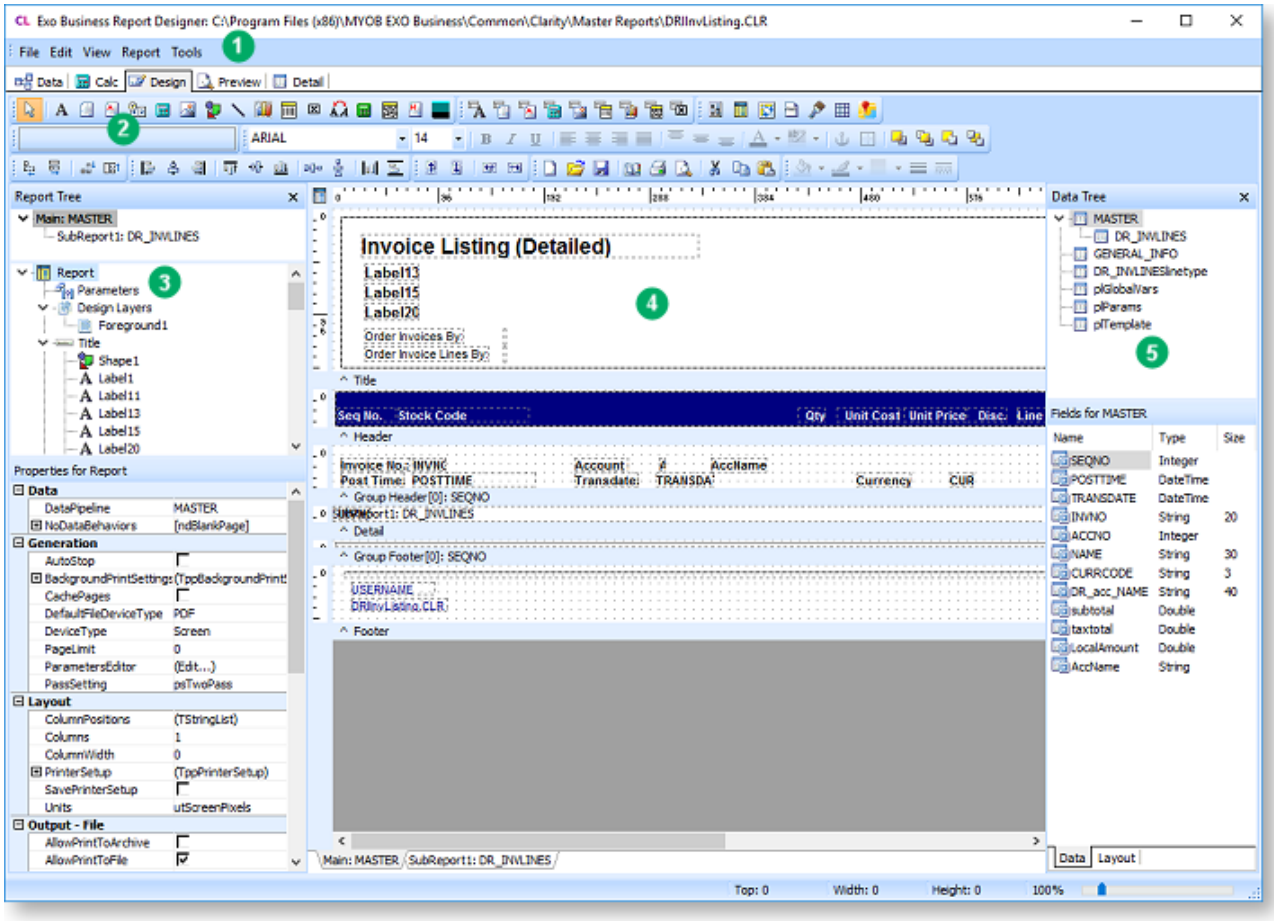
The second layer is a set of views that are based on the VW_FACT views, but have been set up with user-friendly column names for ease of use. These views are identified by names that start with "Analytics_" - you can see them in use in the Exo Business Analytics module.

All of these fact views (as well as all other views available in the Exo Business database) can be selected on the first screen of the Query Wizard or the Tables tab of the Query Designer:



Design Tab

Most of the work of building a report is done on the Design tab. After a data source has been set up on the Data tab, the content and appearance of the report is set on the Design tab.

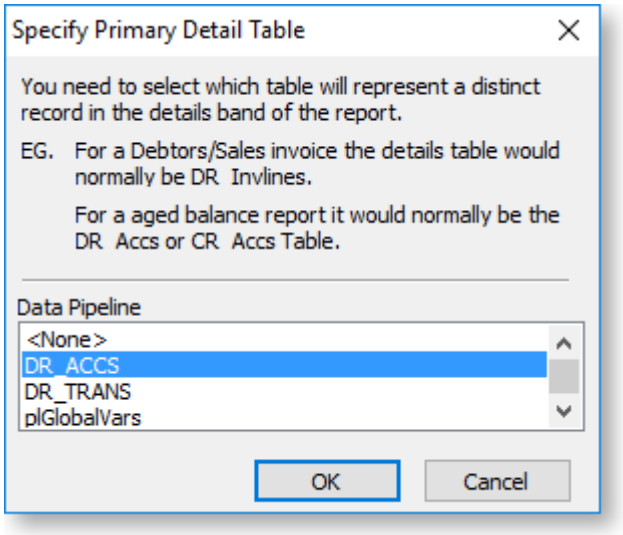


The Design tab is divided into the following main areas:

1. Dropdown menus contain options for creating reports, controlling the appearance of the Design tab, and opening report utilities.
2. A variety of toolbars contain the controls for placing and formatting content on the report.
3. The Report Tree displays the components of the selected report band in a tree structure, with the properties of the selected component displayed below.
4. The main editing canvas is where the various components of the report are arranged.
5. The Data Tree displays the data sources that have been set up from the report, and lets you create data-aware report components by dragging fields onto the editing canvas.

Setting the Primary Table

Whenever you go to the Design tab after making changes on the Data tab, the Specify Primary Detail Table window opens:



You must select which source on the Data tab represents the primary detail section of your report, i.e. the main table on which you will be building the report.

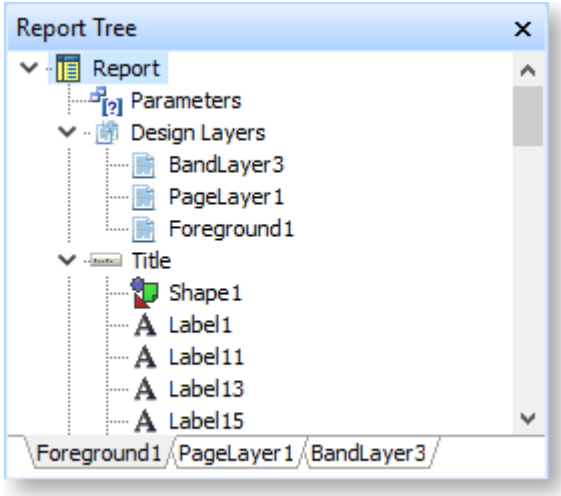
You can open this window at a later time by selecting **Data** from the Report menu.

Design Layers

A report can contain multiple layers, each with its own design. Layers are displayed on top of each other, allowing you to place elements like background images, watermarks or header/footer images into a background layer without affecting any parts of the main report in the foreground layer. Layers can also be saved and loaded independently of the main report, so that you can create reusable report elements or templates that are common to all of your reports.

> **Note:** In older versions, the Page Style report band was used to contain background elements like watermarks. Enabling the **Page Style** option in the Report menu now adds a new page layer containing only a single Page Style band.

All design layers in the report are displayed on the Report Tree. Initially, there is only a single Foreground layer; if more layers are added, tabs appear at the bottom of the tree, allowing you to switch between them:



Whatever layer is selected in the Report Tree is displayed at the "top" of the report canvas, with all other layers appearing greyed out behind it.

To add a new layer, right-click on **Design Layers** and choose one of the following options:

- **New Page Layer** - adds a new design layer that contains only a single Page Style report band.
- **New Band Layer** - adds a new design layer that contains all of the report bands that are in the Foreground layer.

You can right-click on a layer in the Report Tree to access management options:

Option	Description
Open	Loads the contents of the layer from a .LTM file.
Save	Saves the contents of the layer to a .LTM file, so that it can be reused in other reports.
Rename	Lets you enter a new descriptive name for the layer.
Delete	Deletes the layer and moves any items on it to the Foreground layer.
Delete Item With Components	Deletes the layer and all items on it.

Note: Only the **Rename** option is available for the Foreground layer.

Design Tab Menus

The Design tab includes the following functions in its dropdown menus. These menus are also available from the Preview and Detail tabs.

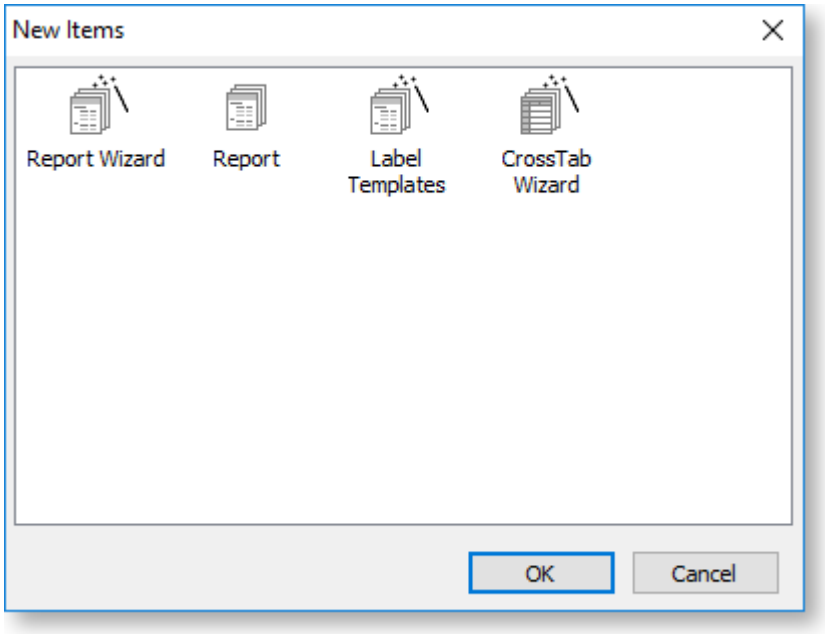
Note: Some of these functions require you to have set up a data source on the **Data tab** before you can use them.

File Menu

New

Offers several methods of creating a report:

- Use the Report Wizard to create a new report.
- Create a new blank report.
- Create a new report for printing labels.
- Create a new CrossTab report.



New Report

Creates a new blank report.

Open

Opens a previously saved report

Close

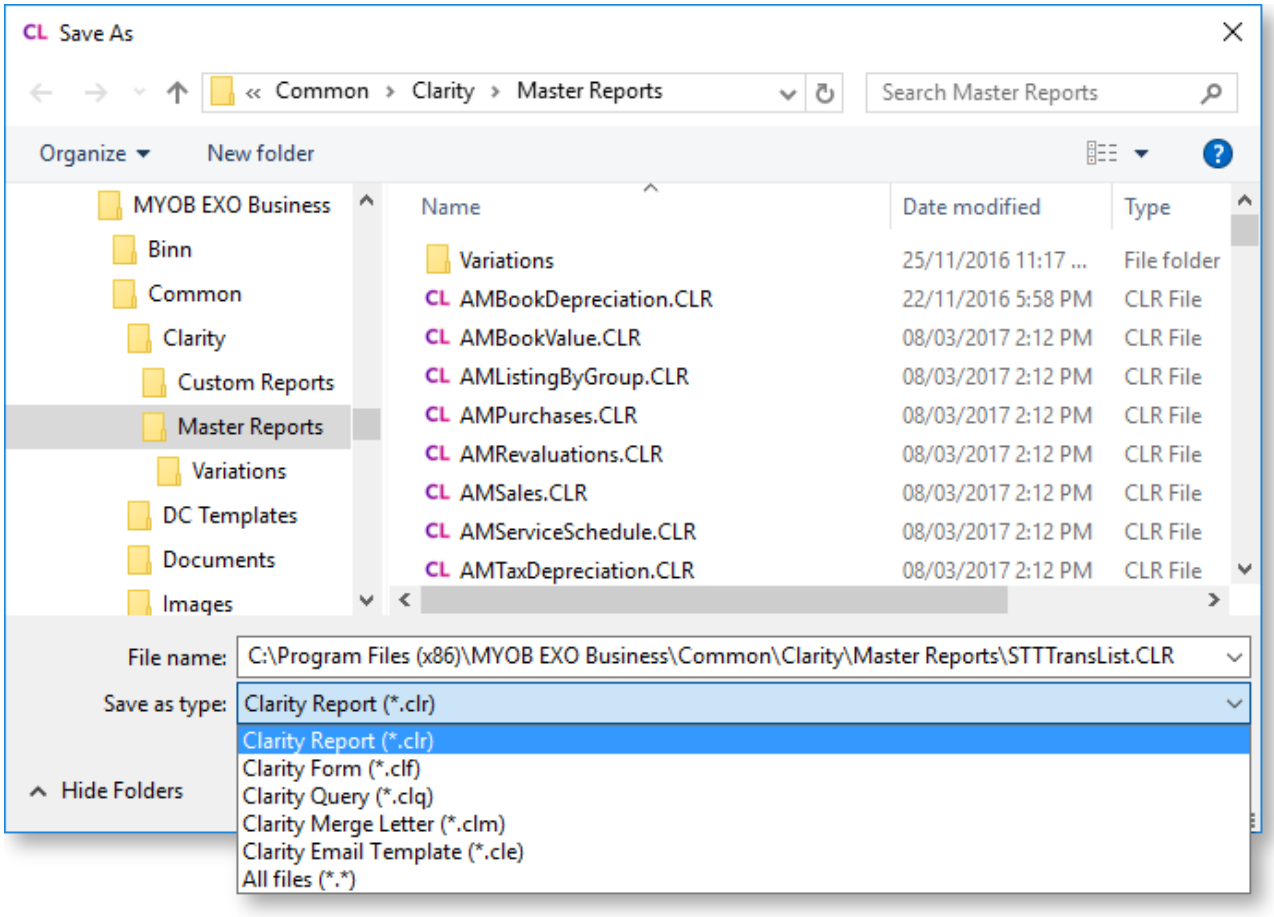
Closes the Exo Clarity application.

Save

Saves the current report.

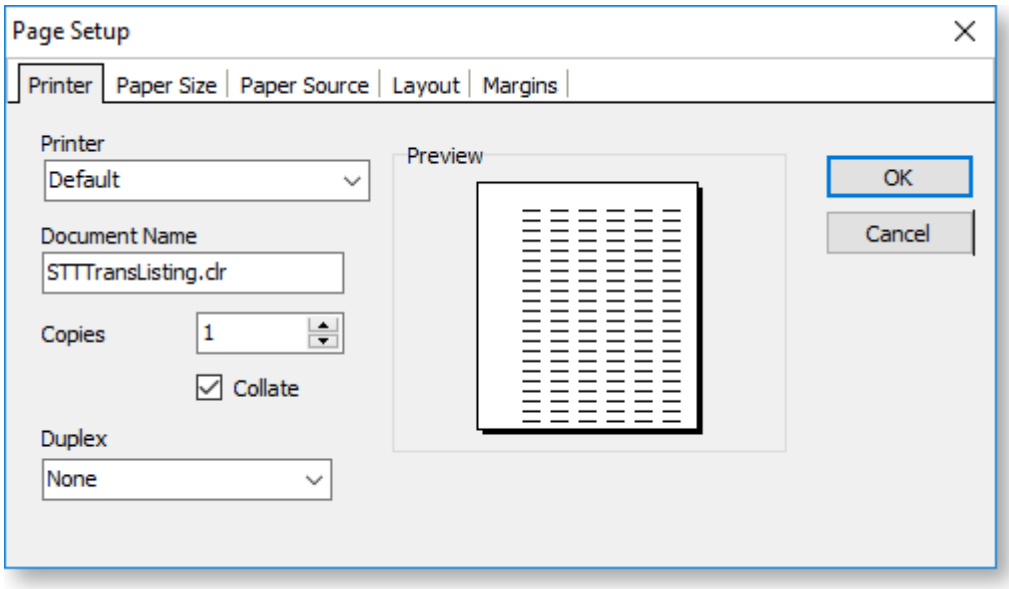
Save As

Saves the current report, giving you the option to enter a new name for the report and select its report type.



Page Setup

Opens the Page Setup window, where you can set up the page layout to use when printing the report.



Print

Prints a physical copy of the report.

Print to File Setup

Opens the Print to File Setup window, which is used to set up options for exporting the report to Excel.

Edit Menu

The Edit menu contains standard options for undoing edits as well as cutting, copying, pasting and deleting elements. It also contains options for ordering report elements when they are layered on top of each other:

- Bring to Front
- Bring Forward
- Send to Back
- Send Backward

Note: These ordering options are also available on the **Format toolbar**.

View Menu

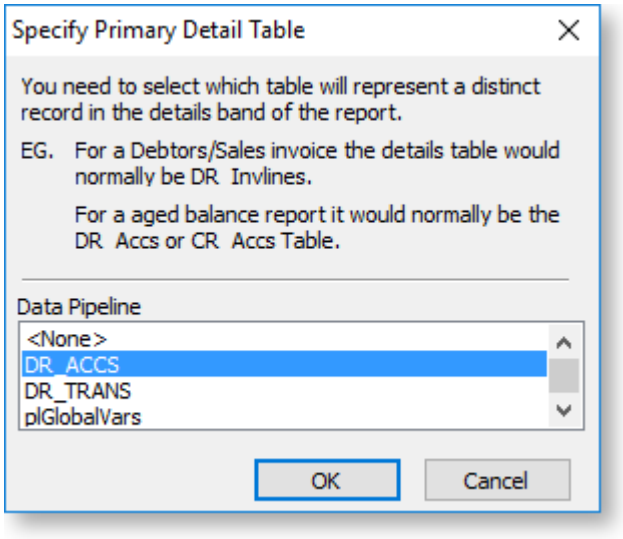
The View menu contains options that affect the appearance of the Exo Clarity application:

- Toolbars - this is a sub-menu that lets you turn all of the Design tab toolbars on or off.
- Rulers - lets you turn the rulers around the main report canvas on or off.
- Design Guides - turns on guide lines that appear when clicking on a component and dragging it, to help you align components with each other.
- Grid Options - opens a window that lets you set up the gridlines on the main report canvas.
- Show Data - for report components that will be replaced with real data when the report is run, this option lets you toggle between displaying the components' field names, or the data they will display when the report is run.
- Themes - select a visual theme for the Exo Clarity application.

Report Menu

Data

Exo Clarity needs to know which source on the Data tab represents the primary detail section of your report, i.e. the main table on which you will be building the report. Selecting this option opens the Specify Primary Detail Table window, where you can select the primary table:



When a report contains one or more sub reports, each sub report can have a different data source from the main report. You can use this window to select the data source for each sub report.

Title / Summary / Header / Footer / Page Summary / Page Style

These options let you turn individual report bands on or off.

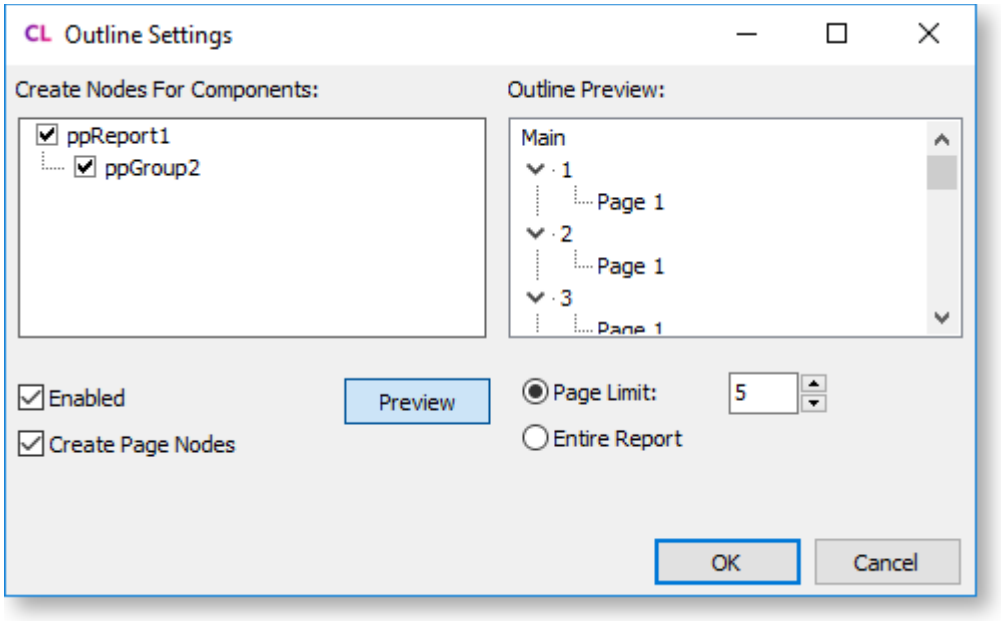
> **Note:** The function of the Page Style band is performed by **design layers**. Enabling the Page Style band from this menu adds a new page layer containing only a Page Style band.

Groups

Opens the Groups window, which lets you set up data grouping in the report.

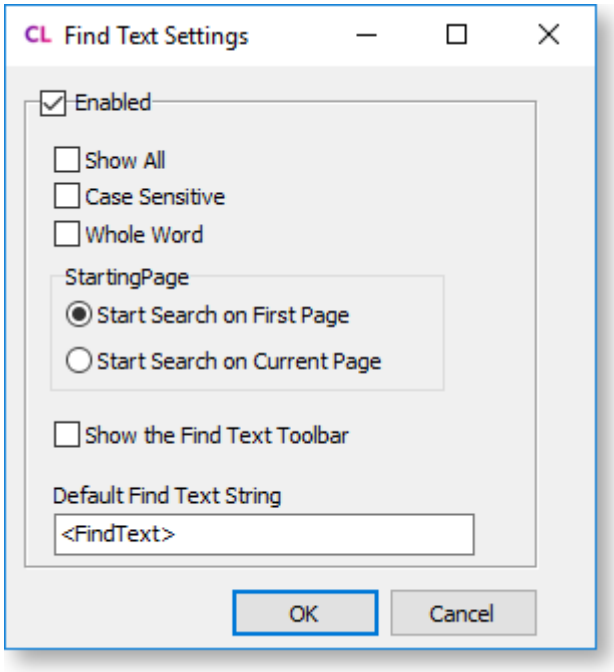
Outline Settings

Opens the Outline Settings window, where you can enable and set up a tree view display of the various groups in the report and optionally the pages they appear on to appear on the Preview tab.



Find Text Settings

Opens the Find Text Settings window, where you can enable and set up the Find Text toolbar on the Preview tab.



Portrait / Landscape

Select the page orientation for the report: portrait or landscape.

Cache Pages

If this option is enabled, all pages generated by the report engine will be saved an re-used. For example, if this setting is enabled and the report is sent to the Print Preview form, Exo Clarity will not regenerate the pages if the report is subsequently sent to the printer - it will just use the cached versions. This can save time when repeatedly generating large, data-intensive reports; however, it can take up a great deal of memory.

Pass Setting

This is an advanced option. If "Two Pass" is selected, Exo Clarity goes through the data twice, performing all calculations on the first pass and then formatting and printing the report on the second pass. If "One Pass" is selected, everything is done in one pass. In most cases, it shouldn't matter which option you choose; however, for more complex reports you may need to take this setting into account to ensure, for example, that variables are not calculated twice (when set to "Two Pass"), or that report totals are calculated correctly before generating (if set to "One Pass").

Units

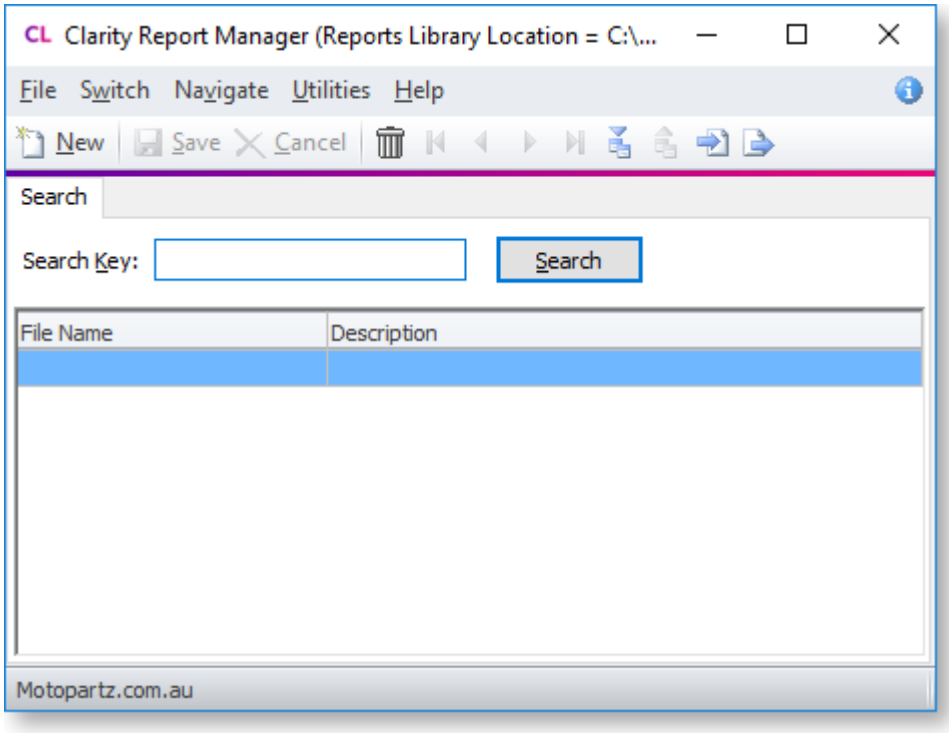
Select the units that report components' dimensions will be displayed in on the Report Tree.

Tools Menu

Report Manager

By default, MYOB Exo Business opens reports from the directories specified by the profile settings **Folder location for standard Clarity forms** and **Folder location for custom Clarity forms (.CLR,.CLF and .FMT)**. This may not be suitable in an environment where there is no access to the shared folders on other systems, however. If the **Store Clarity report files in database** profile setting is enabled, reports can be stored in and accessed from the Exo Business database.

The Report Manager utility lets you import Exo Clarity reports into the Exo Business database, individually or in batches.



Note: When reports are stored in the database using the Report Manager, there is a consistency in terms of the versions of the reports on all systems, which eliminates problems with report editing/version control, and makes management of reports easier. However, the downloading of reports becomes slower, because reports must be extracted from the database and then run.

Runtime Parameters

Opens the Parameters Editor window, where you can set up parameters for the report.

Convert

The Convert sub-menu provides access to a number of report conversion utilities:

- **FMT Conversion** - Converts legacy .FMT files to Exo Clarity .CLR files.
- **Custom Report Conversion** - Create a new custom report by selecting from a predefined set of reports.
- **Convert Report (StockCode)** - Converts clarity reports to 23 characters stock code.
- **Convert Report (IB <---> MSSQL)** - Converts a legacy Interbase report to an MSSQL one and vice versa. (Note that Interbase reports are only supported up to MYOB Exo Business version 6.2.)
- **GL Report Conversion** - Creates clarity forms for reports that were written in the GL Report Writer in Exo Business.

SQL to Report

Opens a utility that generates a report based on a SQL statement. Enter a SQL statement and follow the steps of the utility to generate a report based on it.

Validate Report

This performs a basic check on the validity of the report (fields, links, code, etc.) and gives a message of "No error found" if validation is completed successfully. It is a good idea to run this before you save your reports, although there are still errors and inconsistencies that this tool is not able to warn you about, so you should still check reports manually and preview them.

Summary to Bottom

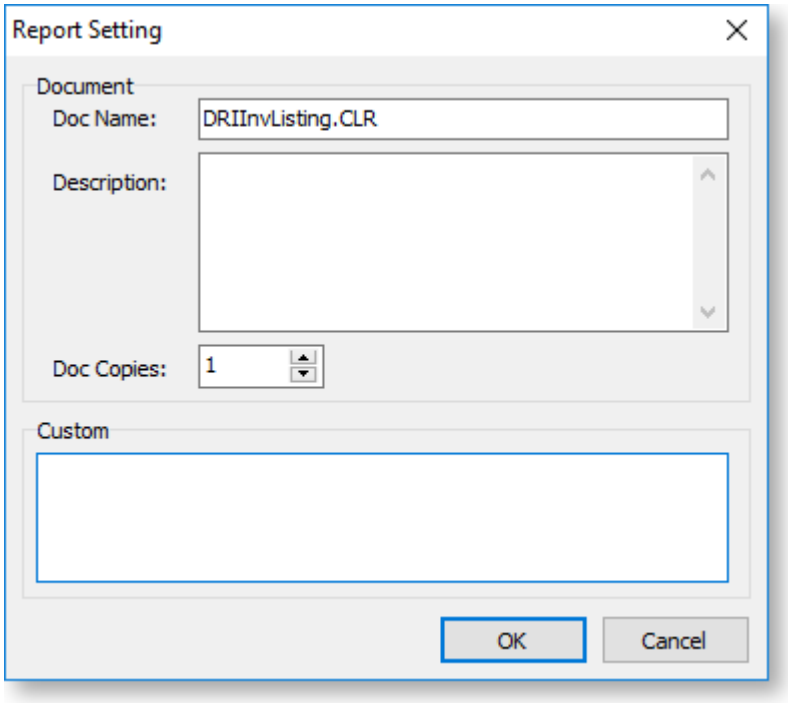
This option determines whether the Summary report band appears immediately after the last Detail item on your report (not selected), or at the bottom of the page, just above the Footer band (selected).

Switch

Select whether the report is a **Standard Report** or a **Custom Report**. This determines whether the report is stored in the Exo Business database as a Custom report or a Master/Standard report, and takes effect when Exo Business is set up to store report definitions in the database rather than in files.

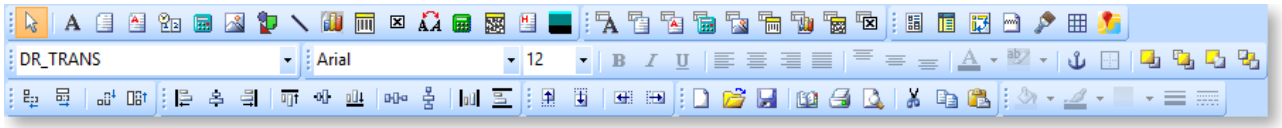
Setting

Opens the Report Setting window where you can enter the report document's name, some descriptive text for the report, the number of copies that should be printed when the report is output to a printer, and a **Custom** box to set advanced output options.



Design Tab Toolbars

The Design tab features a variety of toolbars that let you place and edit components on the report canvas. Toolbars can be turned on or off from the View menu.



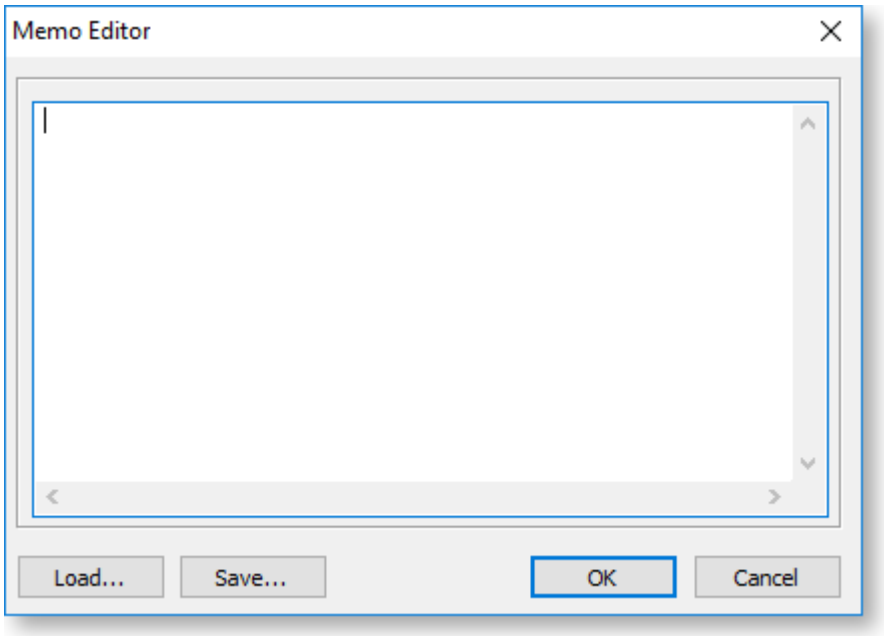
The following toolbars are available:

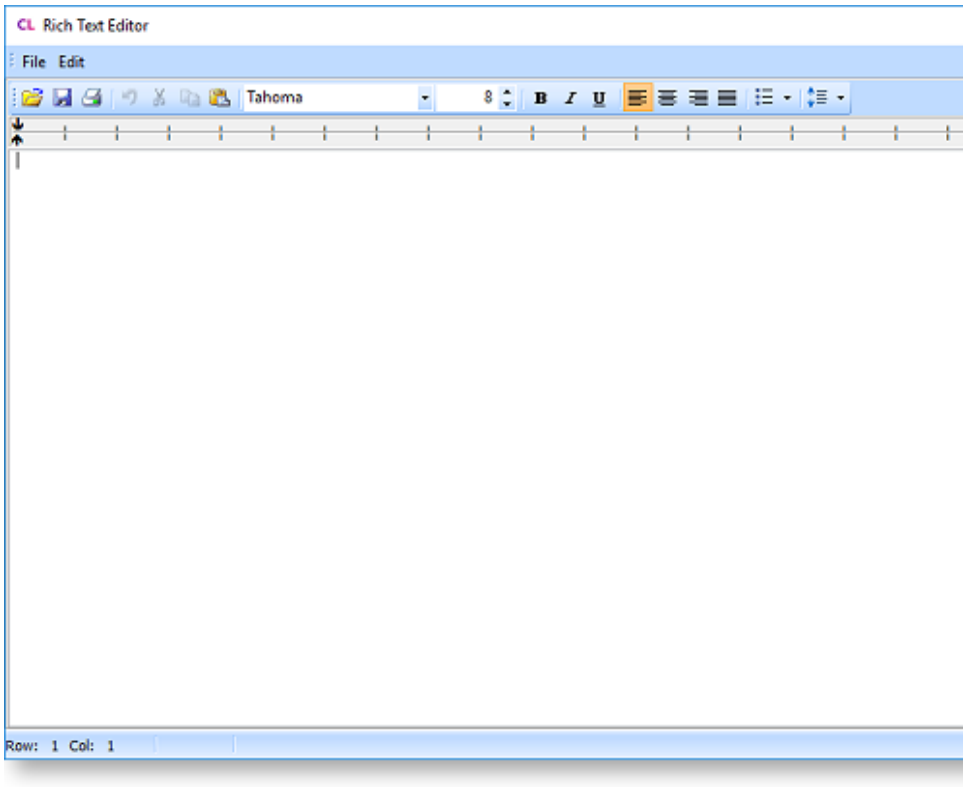
- Standard Components
- Data Components
- Advanced Components
- Edit
- Format
- Size
- Align or Space

- Nudge
- Standard
- Draw

Design Tab - Standard Components Toolbar

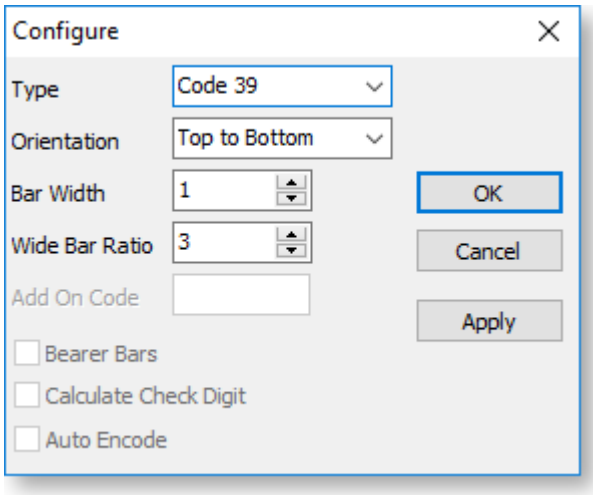
The tools on the Standard Components toolbar are mostly for adding display components to the report, e.g. headings, labels or company logos. To add a standard component to a report, click the button for the component you want to add, then click on the report canvas in the place where you want the component to be. Components can be moved and re-sized after you have added them, and right-click options are available to set advanced properties.

Button	Name	Description
	**Select**	When this tool is enabled, you can select components on the report canvas. This tool is enabled by default - whenever you use another tool to place a component, after the component is placed, the Select tool becomes enabled again.
	**Label**	Adds a text label to the report. You can double-click on a Label component to edit the text in it directly, or select it use the Edit toolbar, which becomes a text box where you can enter the label's caption. You can change the appearance of the label using the controls on the Format toolbar.
	**Memo**	Adds a block of text to the report. You can double-click on a Memo component to edit the text in it directly, or you can right-click on the component and select **Lines** to open the Memo Editor, where you can enter text, load text from a .TXT file, or save the text as a .TXT for re-use. 

	Rich Text	Adds a block of formatted text to the report. To edit the text, right-click on the component and select **Edit** to open the Rich Text Editor, where you can enter and format text. You can also load text from a .RTF file, or save it for re-use. This window also has a **Mail Merge** section that lets you set up a mail merge in a report. 
	System Variable	Adds a system variable like the page number, page count, print date and time, or current date and time to the report. When a system variable is selected, the Edit toolbar becomes dropdown where you can select the variable to display.
	Variable	Adds a variable for use in calculations on the report - these can be used in calculations on the Calc tab. You can right-click on a variable and select **Calculations** to open the Calculations Window and set up calculations for the variable directly.
	Image	Adds a graphic to the report. Right-click on the component and select **Picture** to choose the graphic file to display.
	Shape	Adds a simple shape, e.g. a rectangle or ellipse to the report. When a shape is selected, the Edit toolbar becomes a dropdown where you can select the kind of shape. You can change the shape's appearance using the controls on the Draw toolbar.
	Line	Adds a straight line to the report. When a shape is selected, the Edit toolbar becomes a dropdown where you can select the line's orientation. You can change the line's colour and width using the controls on the Draw toolbar.
	Chart	Adds a chart to the report.

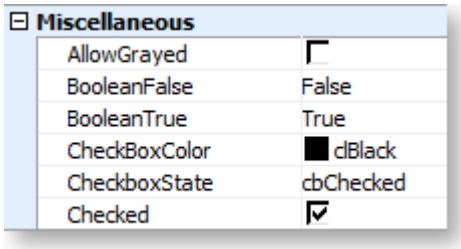
BarCode

Adds a barcode to the report. When a barcode is selected, the Edit toolbar becomes a text box where you can enter the text that should be encoded in the barcode. Right-click on the component and select **Configure** to opens a window where you can select the barcode standard to use and set display options.



Checkbox

Adds a checkbox graphic to the report. You can set the checkbox to appear checked, unchecked or greyed out by editing its by editing its **Checked** or **CheckboxState** properties on the Report Tree. (The greyed out state can only be selected if the **AllowedGrayed** property is ticked.)



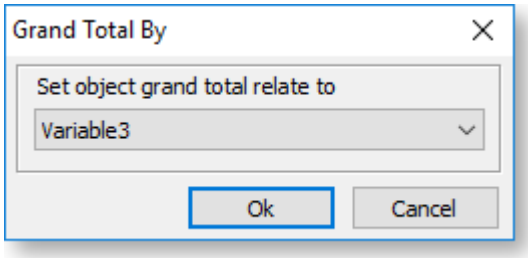
RotatedLabel

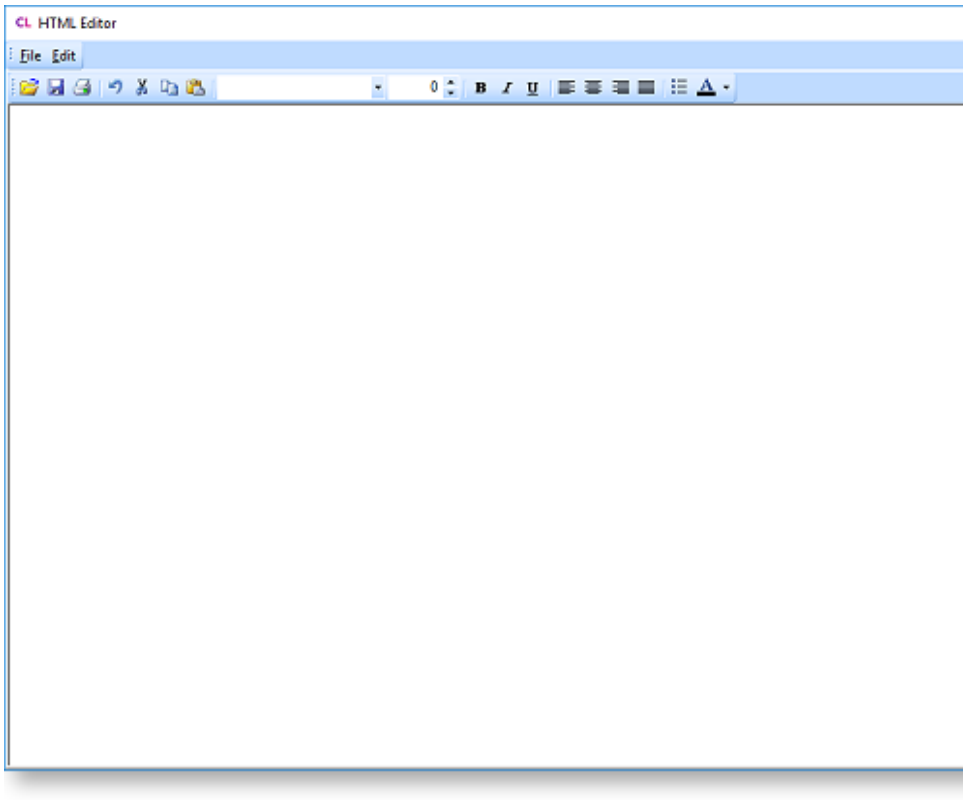
Adds a rotated text label to the report.

Note: This component has been superseded - you can edit the angle of a regular Label component by right-clicking on it and selecting **Position**.

TotalVariable

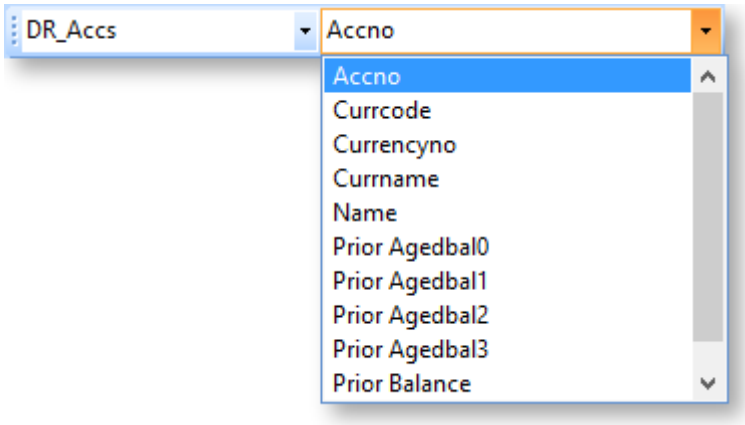
Adds a component that totals the values of a selected variable. When a total variable component is selected, the Edit toolbar becomes a dropdown where you can select the variable's data type. Right-click on the component and select **Grand Total** to open a window where you can select the variable that will be totalled:



	2DBarCode	Adds a two-dimensional barcode to the report. 2D barcodes, also known as QR Codes, can contain more data than a linear barcode. When a 2D barcode is selected, the Edit toolbar becomes a text box where you can enter the text that should be encoded in the barcode.
	HTML Text	Adds a block of HTML code to the report, which can be used as a template for HTML mail merges. To edit the text, right-click on the component and select **Edit HTML** to open the HTML Editor, where you can enter and format text. Report variables are available on the right of the window - you can add a variable to the text by double-clicking on it or by selecting it and clicking **Add Field**. You can also load text from a .HTML file, or save it for re-use.  Add a hyperlink by selecting text, right-clicking and selecting **Edit hyperlink**. Add an image by right-clicking and selecting **Insert image**. **Note:** Image source files are assumed to be in the Exo Business shared images folder, as specified by the **Folder location for images** profile setting.
	GradientPanel	Adds a rectangular panel containing a linear gradient to the report. **Note:** This component has been superseded - you can add a gradient to any Shape component using the controls on the **Draw toolbar**.

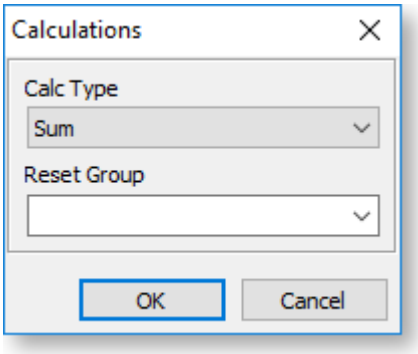
Design Tab - Data Components Toolbar

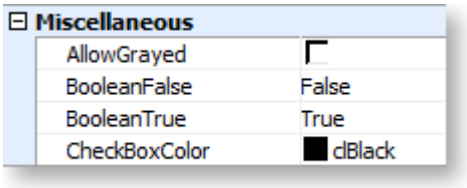
The tools on the Data Components toolbar are similar to those on the Standard Components toolbar; however the components that these tools are data aware, that is, they are bound to a field from the database, and display the value of this field at runtime. When any data component is selected, the Edit toolbar contains two dropdowns, which allow you to specify the table and field that the component is bound to:



The first dropdown contains all data sources that appear in the Data Tree - this includes any data sources set up on the Data Tab, as well as other data sources that are built into the report, such as global variables and user-created report parameters. The second dropdown contains the fields or parameters available for the selected data source.

To add a standard component to a report, click the button for the component you want to add, then click on the report canvas in the place where you want the component to be. Components can be moved and re-sized after you have added them, and right-click options are available to set advanced properties.

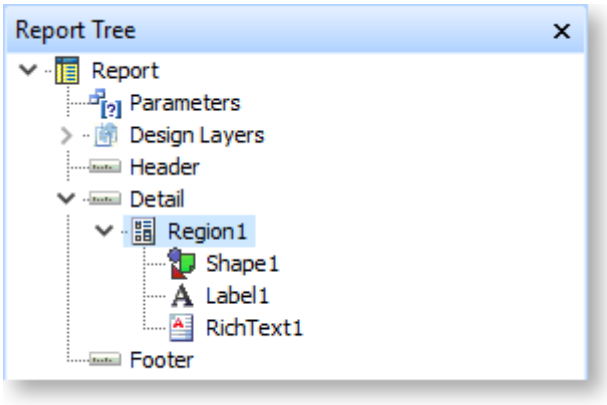
Button	Name	Description
	**DBText**	Adds a single-line plain text component to the report, which can handle most data types, but cannot display rich text or images.
	**DBMemo**	Adds a multi-line plain text component to the report, which can display plain text from the memo field of a database table.
	**DBRichText**	Adds a multi-line rich text component to the report, which can display formatted text from the memo field of a database table.
	**DBCalc**	Adds a field that performs simple database calculations (Count, Sum, Min, Max, Average). When a DBCalc components is first placed on a report it displays "Sum(?)". Use the Edit toolbar to choose the field that the calculation will be performed on. To perform a calculation other than "Sum", right-click on the component and select **Calculations** to open a window where you can select the type of calculation: 

	DBImage	Adds a graphic that is stored in a database field.
	DBBarCode	Adds a barcode to the report, which encodes data from a database field. Right-click on the component and select **Configure** to opens a window where you can select the barcode standard to use and set display options.
	DBChart	Displays a chart based on one or more data series from the defined data sources.
	DB2DBarCode	Adds a two-dimensional barcode to the report, which encodes data from a database field. 2D barcodes can contain more data than a linear barcode.
	DBCheckBox	Adds a checkbox graphic, whose status is taken from a Boolean value in the database. Select the field whose value determines the checkbox's status from the Edit toolbar, then enter the values that signify "checked" or "not checked" into the **BooleanTrue** and **BooleanFalse** properties under Miscellaneous on the Report Tree. By default, these are set to "True" and "False" but depending on the contents of database field they may need to be "T" and "F", "Y" and "N", "1" and "0", or some other combination. 

Design Tab - Advanced Components Toolbar

The tools on the Advanced Components toolbar let you add more components that help you deal with more complex reports.

Button	Name	Description
	Region	Adding a region to a report lets you group other components together. Once you've added a region component, you can place existing components on it by dragging them onto the region, or you can create new components in the region by clicking the button for the kind of component you want to add, then clicking directly on the region. The Report Tree shows which components are grouped under a region:



When the region is moved, all components on it move as well. If a region is made invisible, all components on it become invisible as well.

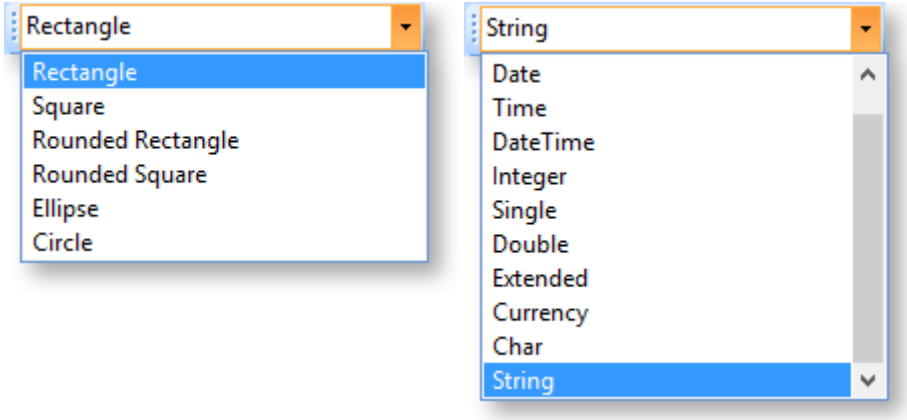
	SubReport	Adds a sub report to the report.
	CrossTab	Adds a crosstab to the report, which summarises grouped or aggregated data from database tables in a spreadsheet-like grid format.
	Page Break	Adds a page break in the report, allowing you to control the report's pagination more precisely.
	PaintBox	Adds a paintbox area to the report, which acts as a mini canvas that can be drawn on programmatically using the Calc tab. You can use the PaintBox component to create shapes and text that might otherwise be impossible using other report components.
	TableGrid	Adds a table layout to the report. Other components can be placed inside each cell of the table, in the same way that components can be placed in a region. Right-click on the table to add rows and columns.
	Google Map	Adds an image displaying a map from Google Maps. Use the component's properties on the Report Tree to control what appears on the map: • Enter an address into the **Address** property or latitude/longitude coordinates under **Coordinates** to control the location displayed on the map. • Select an option from the **MapType** dropdown to specify the kind of map to display: a road map, a satellite photograph, a hybrid map showing roads overlaid on a satellite photograph, or a terrain map. • Enter a **Zoom** value to control how far the map should zoom in on the address/coordinates. In general, 10 shows a whole city, 15 shows individual streets, and 20 shows individual buildings on a street. See the Google Maps documentation for more information on Google Map settings and parameters. **Note:** As with all components, you can change a Google Map component's properties using code entered on the **Calc tab**. This allows you to change a map's appearance dynamically, e.g. by loading the values of Debtor accounts' address fields into the component's Address property, you can display a map next to each Debtor record showing its address.

Design Tab - Edit Toolbar

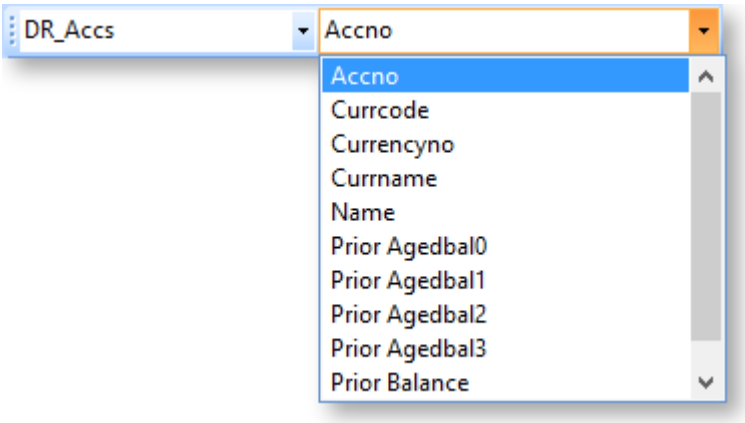
The Edit toolbar changes depending on the kind of component that is currently selected.

For components that display plain text, or components where the only thing that can be edited is the name, the Edit toolbar contains a simple text box where you can enter the name or text to display:

For components that have multiple options for how they should appear or what they should contain, the Edit toolbar contains a dropdown of available options. For example, when selecting a Shape component, the Edit toolbar dropdown contains the available shapes that the component can be; when selecting a Variable component, the Edit toolbar dropdown contains the available data types:



When a data-aware component is selected, the Edit toolbar contains two dropdowns, which allow you to specify the field that the component should be bound to:



The first dropdown contains all data sources that appear in the Data Tree - this includes any data sources set up on the Data Tab, as well as other data sources that are built into the report, such as global parameters and user-created report parameters. The second dropdown contains the fields or parameters available for the selected data source.

Design Tab - Format Toolbar

The Format toolbar lets you control the appearance of text-based components. It also contains options for ordering components that are layered on top of each other.

Note: These controls affect the entire component, e.g. clicking the Bold button makes all text in the component bold. If you want to have a mixture of text formatting in one component, you will need to use a **Rich Text or HTML Text component**.

Button	Name	Description
	**Font Name**	Select the font face for a text component.
	**Font Size**	Select the font size for a text component
	**Bold**	Sets all text in a component to **bold**.
	**Italic**	Sets all text in a component to *italic*.
	**Underline**	Sets all text in a component to underline.
	**Left Justify**	Aligns text to the left side of the component's bounding box.
	**Center**	Aligns text to the right side of the component's bounding box.
	**Right Justify**	Centers text relative to the component's bounding box.
	**Justify**	Fully justifies text relative to the component's bounding box, i.e. the text on each line is padded out so that it aligns with the left and right sides of the bounding box.
	**Align Top**	Aligns text to the top of the component's bounding box.
	**Align Middle**	Aligns text to the middle of the component's bounding box.
	**Align Bottom**	Aligns text to the bottom of the component's bounding box.
	**Font Colour**	Sets the colour of a component's text.
	**Highlight Colour**	Sets the background colour that will appear behind the component's text.
	**Anchors**	Defines how the component should be anchored - this affects how it will dynamically reposition itself if the report's layout changes. When the component's parent is resized, the component holds its position relative to the edges to which it is anchored. **Note:** Anchoring properties can also be set in the Layout - Dynamic section of the **Report Tree**.
	**Border**	Sets the component's border colour - select "None" for no border.

	Bring Forward	Where multiple components are layered on top of each other, this button brings the selected component forward one place.
	Bring to Front	Brings the selected component in front of all other components that it overlaps with.
	Send Backward	Where multiple components are layered on top of each other, this button sends the selected component backward one place.
	Send to Back	Sends the selected component behind all other components that it overlaps with.

Design Tab - Size Toolbar

The tools on the Size toolbar let you set multiple components to a uniform height and/or width. These tools have no effect on a single component - you need to select multiple components before you use them.

Button	Name	Description
	Shrink Width to Smallest	Resizes all selected components so that their width is set to the width of the narrowest selected component.
	Grow Width to Largest	Resizes all selected components so that their width is set to the width of the narrowest selected component.
	Shrink Height to Smallest	Resizes all selected components so that their width is set to the width of the narrowest selected component.
	Grow Height to Largest	Resizes all selected components so that their width is set to the width of the narrowest selected component.

Nudging Component Sizes

You can hold down SHIFT and use the arrow keys to "nudge" the sizing of one or more components by one pixel at a time:

- SHIFT+LEFT ARROW - decrease the width of all selected components by one pixel
- SHIFT+RIGHT ARROW - increase the width of all selected components by one pixel
- SHIFT+UP ARROW - decrease the height of all selected components by one pixel
- SHIFT+DOWN ARROW - increase the height of all selected components by one pixel

Design Tab - Align or Space Toolbar

The tools on the Align or Space toolbar let you position multiple components relative to one another. Most of these tools have no effect on a single component - you need to select multiple components before you use them. In some cases, the order in which you select the components matter - for example, when aligning multiple components, all components will be aligned with the component that you select first.

Button	Name	Description

	Align Left	Aligns the left edge of all selected components with the left edge of the component that was selected first.
	Align Middle	Horizontally centers all selected components relative to the horizontal center of the component that was selected first.
	Align Right	Aligns the right edge of all selected components with the right edge of the component that was selected first.
	Align Top	Aligns the top edge of all selected components with the top edge of the component that was selected first.
	Align Center	Vertically centers all selected components relative to the vertical center of the component that was selected first.
	Align Bottom	Aligns the bottom edge of all selected components with the bottom edge of the component that was selected first.
	Space Horizontally	Equally spaces all selected components based on the leftmost position of the first selected component and the rightmost position of the last selected component.
	Space Vertically	Equally spaces all selected components based on the topmost position of the first selected component and the bottom most position of the last selected component.
	Center Horizontally in Band	Horizontally center one or more selected components relative to the report band they are in. If multiple components are centered, the horizontal center of the entire selection is aligned with the center of the band.
	Center Vertically in Band	Vertically center one or more selected components relative to the report band they are in. If multiple components are centered, the vertical center of the entire selection is aligned with the center of the band.

Design Tab - Nudge Toolbar

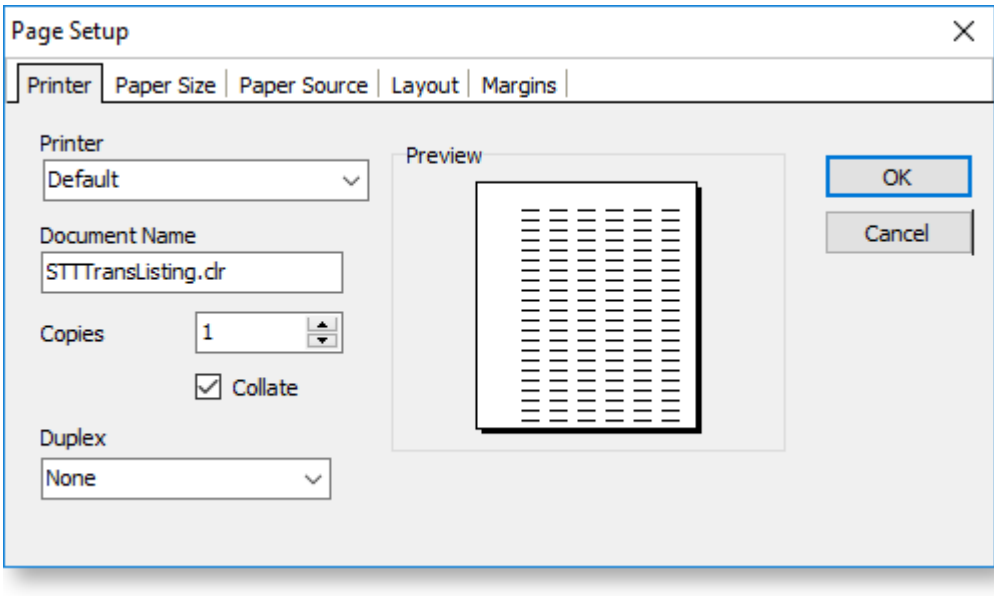
The Nudge toolbar lets you position one or more components with great precision by "nudging" them one pixel at a time. You can select multiple components and move all of them at the same time.

Note: You can also hold down CTRL and use the arrow keys to nudge the selected components.

Button	Name	Description
	Nudge Up	Moves all selected components up one pixel.
	Nudge Down	Moves all selected components down one pixel.
	Nudge Left	Moves all selected components left one pixel.
	Nudge Right	Moves all selected components right one pixel.

Design Tab - Standard Toolbar

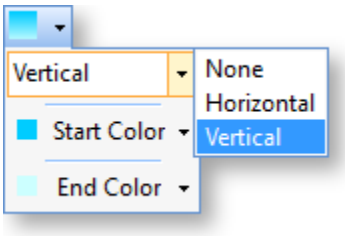
The Standard toolbar allows you to perform basic functions like opening, saving, and printing reports, as well as cutting and pasting selections.

Button	Name	Description
	**New**	Creates a new blank report.
	**Open**	Opens an existing report.
	**Save**	Saves the current report to a file.
	**Page Setup**	Opens the Page Setup window, where you can set up the page layout to use when printing the report. 
	**Print**	Prints a physical copy of the report.
	**Print Preview**	Displays the Preview tab.
	**Cut**	Cuts the selected component(s), moving them to the clipboard and removing them from the report canvas.
	**Copy**	Copies the selected component(s) to the clipboard.
	**Paste**	Pastes the contents of the clipboard onto the report canvas.

Note: You can paste components between reports, which can be useful when copying layouts from one report into another. However, be sure to check that any **data-aware components** pick up the correct data field, and reassign them where necessary. Variable expressions or Calcs code will not be transferred across, so will need to be entered manually.

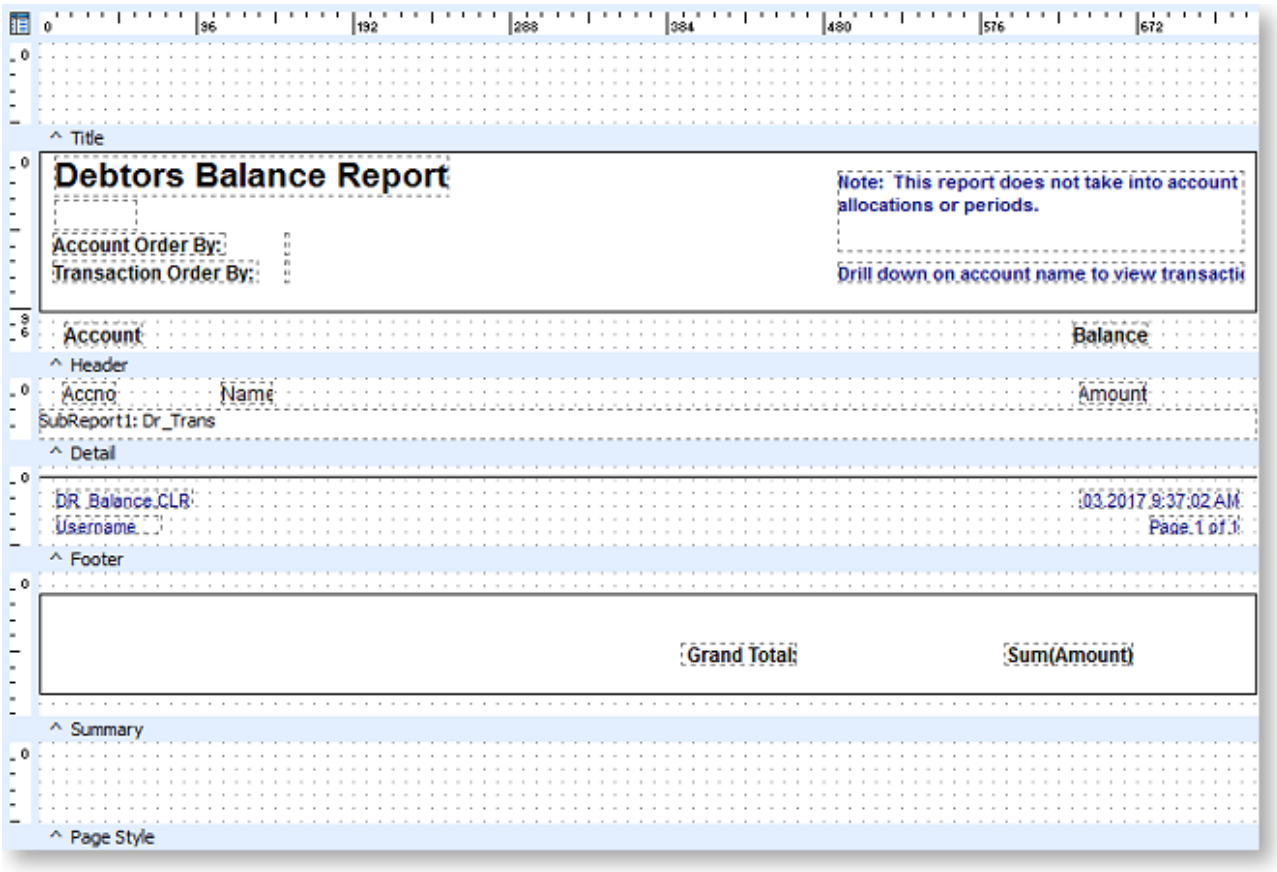
Design Tab - Draw Toolbar

The Draw toolbar contains tools for altering the appearance of graphical report components, e.g. lines, shapes or regions. You can select multiple components and edit them all at once.

Button	Name	Description
	**Fill Colour**	Specifies the fill colour for the selected components.
	**Line Colour**	Specifies the line or outline colour for the selected components.
	**Gradient**	Sets a gradient fill for the selected components. This button has a dropdown menu that lets you select the kind of gradient, the start colour and the end colour: 
	**Line Thickness**	Specifies the thickness of Line components (this does not affect the outline of other kinds of components).
	**Line Style**	Specifies the line or outline style for the selected components, e.g. solid, dotted or dashed.

The Report Canvas

The main area of the Design tab is the report canvas, which is where you place and arrange the components that make up the report. To add a component to a report, click on the toolbar button for the kind of component you want to add, then click on the canvas in the place where you want to add it. Once a component has been added to the canvas, you can position and re-size it, and edit its properties on the Report Tree.



Report Bands

The canvas is divided into bands (or sections), each of which control the contents and appearance of different parts of the report:

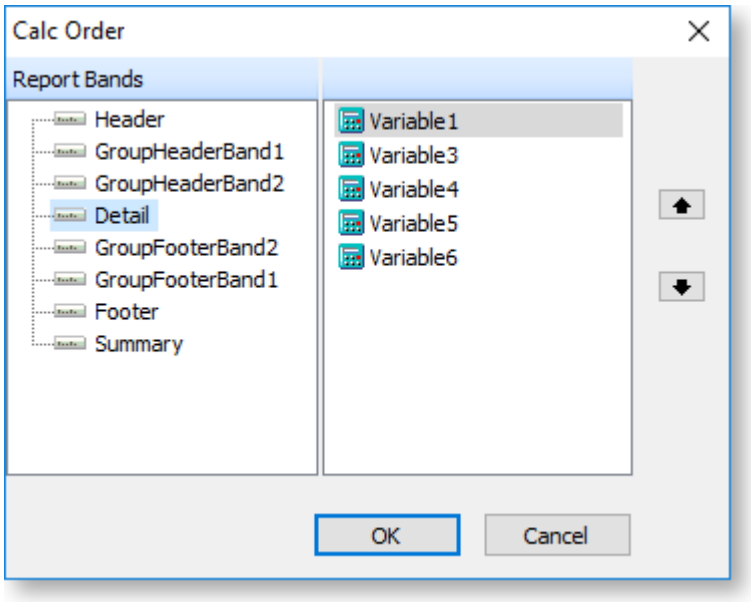
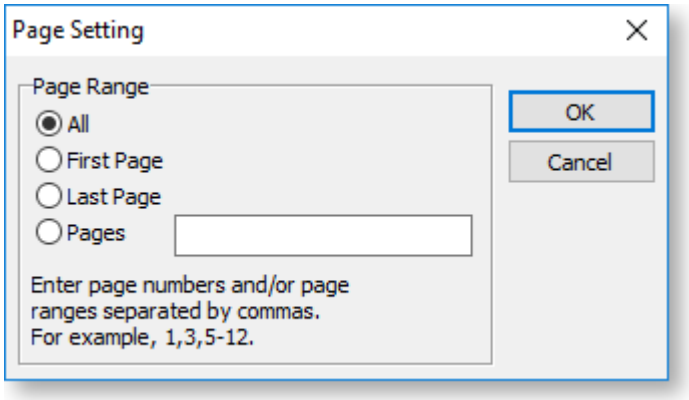
- **Title** - Generally contains the title of the report and prints on the first page only.
- **Header** - Appears at the top of each page.
- **Detail** - Appears once for every row of data.
- **Page Summary** - Appears on each page, after all of the data in the Detail band.
- **Footer** - Appears on the bottom of each page.
- **Summary** - Appears once at the end of the report.
- **Page Style** - This band lets you set components as a background for each page of your report, e.g. background images or watermarks. This band is only available on Page design layers.

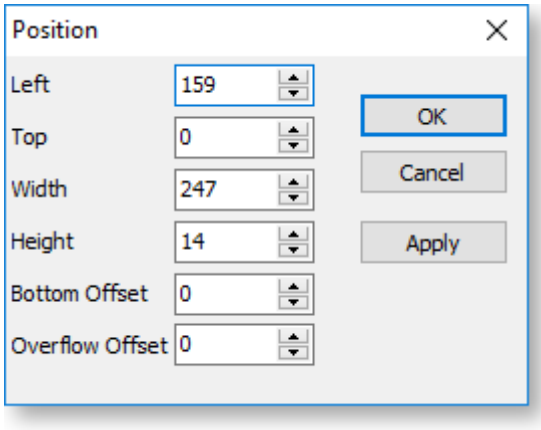
Note: For reports that **group data**, there is a separate Header and Footer band for each group, in addition to the main report Header and Footer.

Each band can be turned on or off using the options on the Report menu. You can make bands taller or shorter by clicking on the area between two bands and dragging up or down.

You can right-click on the area below a band to access controls that affect the band:

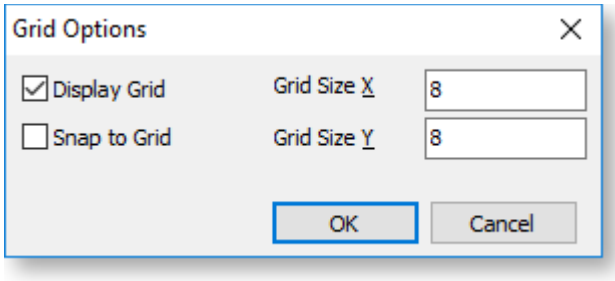
Option	Description
DynamicHeight StaticHeight	Bands that are set to DynamicHeight automatically resize to fit their contents. Bands that are set to StaticHeight maintain the same height regardless of their contents (this is the height that they appear as on the report canvas).

PrintOnFirstPage PrintOnLastPage	These options control whether or not the band should be printed on the first and/or last page of the report.
NewPage	This option applies to the Title or Summary band. If it is enabled, the band will appear on a separate page.
AlignToBottom	This option applies to the Summary band. This option determines whether the Summary report band appears immediately after the last Detail item on your report (not selected), or at the bottom of the page, just above the Footer band (selected).
Calc Order	This option opens a window where you can determine the order in which Variable component calculations are performed for each band. You can use this window when one variable's calculation depends on the value of another calculation, to ensure that they are calculated in the right order. 
Groups	This option applies to Group Header and Group Footer bands. It opens the Groups window, where you can set up grouping for the report.
HideWhenOneDetail	This option applies to Group Footer bands. If enabled, the Group Footer band will not appear if there is only one detail band for the group (or no detail bands). This can be helpful if a total is being calculated in the Group Footer band and duplicate or zero values are unnecessary.
Page Setting	This option applies to the Page Style band. It opens a window where you can specify which pages of the report the Page Style band should appear on: 

Position	This option opens a window where you can specify the exact position of the band. The units on this window are whatever unit was selected from the Report > Units sub-menu. 
Visible	If this option is enabled, the band will appear on the report; if not, the band will not appear on the report, although it will still be visible on the Design tab.

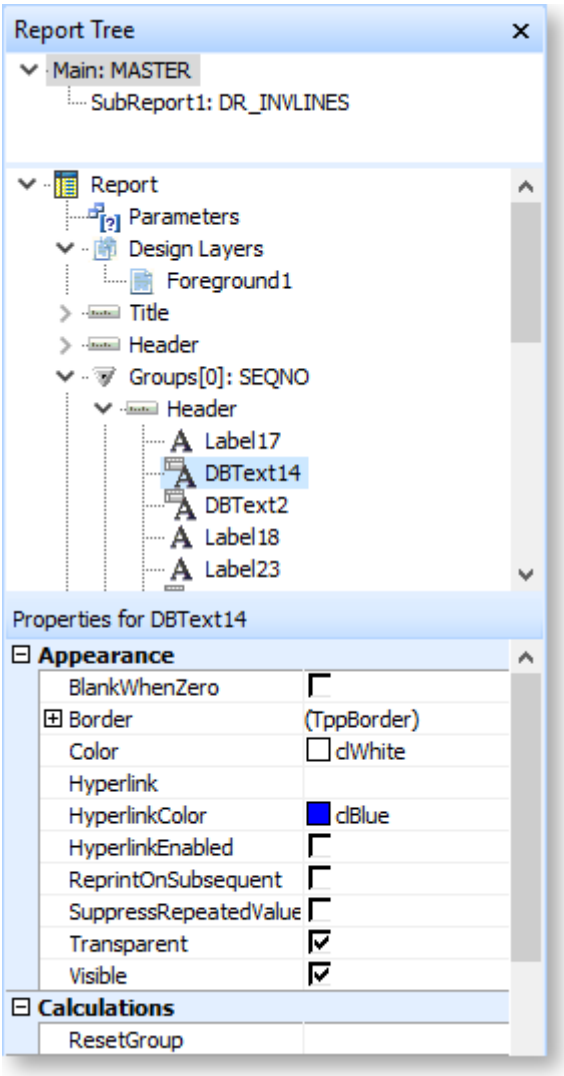
Grid Options

A grid is overlaid on the report canvas, to aid in sizing and aligning report components. You can turn the grid on or off, change the size of the grid, and choose whether or not to snap report components to the grid by selecting **Grid Options** from the View menu.



The Report Tree

The Report Tree displays the components of the report in a tree structure:



Clicking on a band or component on the main report canvas automatically selects that band or component on the Report Tree and vice versa. You can right-click on any component in the tree to access its right-click properties, or right-click on a report band to access its right-click properties.

The design layers set up for the report also appear on the Report Tree. Right-click on the **Design Layers** item to add new layers.

An outline of the report, showing any sub reports that have been set up, is available at the top of the Report Tree. This section can be turned on or off by right-clicking on the Report Tree and selecting **Report Outline**.

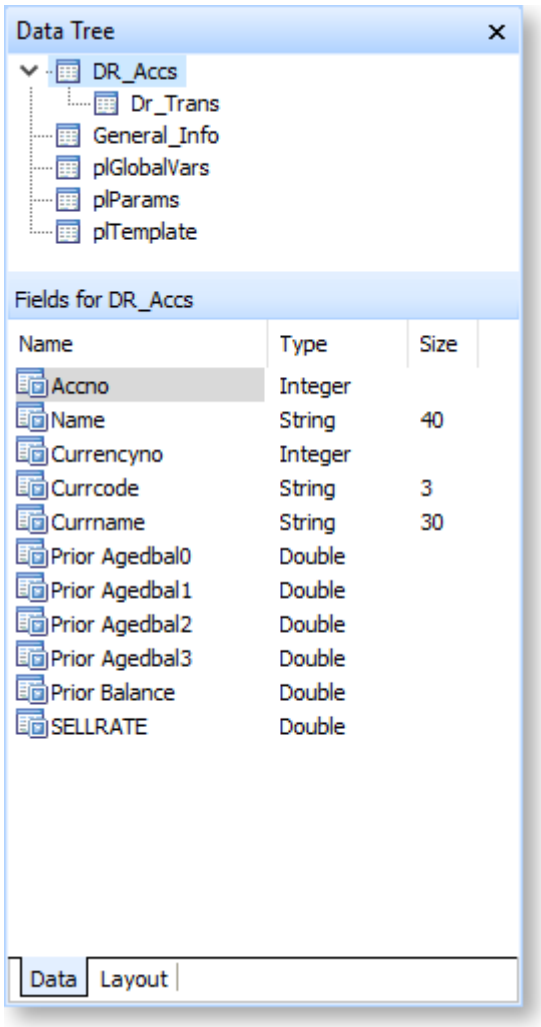
The properties of the selected band or component appear in a Properties section below the main tree. This section can be turned on or off by right-clicking on the Report Tree and selecting **Object Inspector**. All properties displayed in this section are editable - you can use the Properties section to quickly change a component's details or to set advanced options for a component that aren't available from the toolbars or menus.

The Data Tree

The Data Tree displays the data sources that have been set up for the report, and lets you create data-aware report components by dragging fields onto the editing canvas. The Data Tree has two tabs: Data and Layout.

Data Tab

The Data Tree section shows all available data sources for the report, including any data sources set up on the Data Tab, as well as other data sources that are built into the report, such as global variables and user-created report parameters. Selecting an item on this section displays all available fields, variables or parameters for that item in the Fields section below.

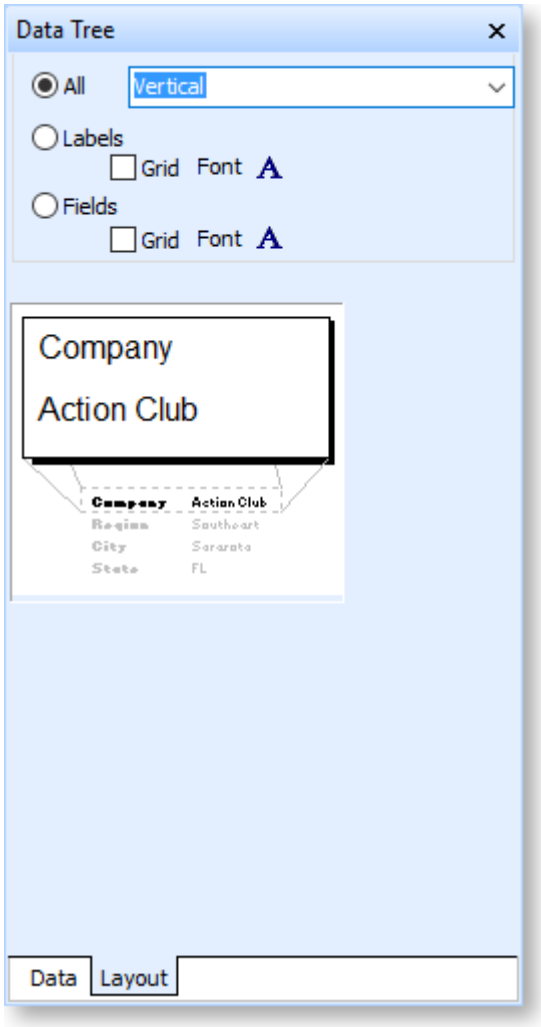


Drag one or more items from the Fields section onto the report canvas to add components to the report (you can select multiple items with the CTRL or SHIFT key). Depending on the options selected on the Layout tab, one or both of the following components will be created for each item:

- A label component, displaying caption text.
- A DBText component bound to the item that you dragged onto the canvas.

Layout Tab

The Layout tab defines the components that are created when data items are dragged onto the canvas and how they are laid out.



Choose from the following options:

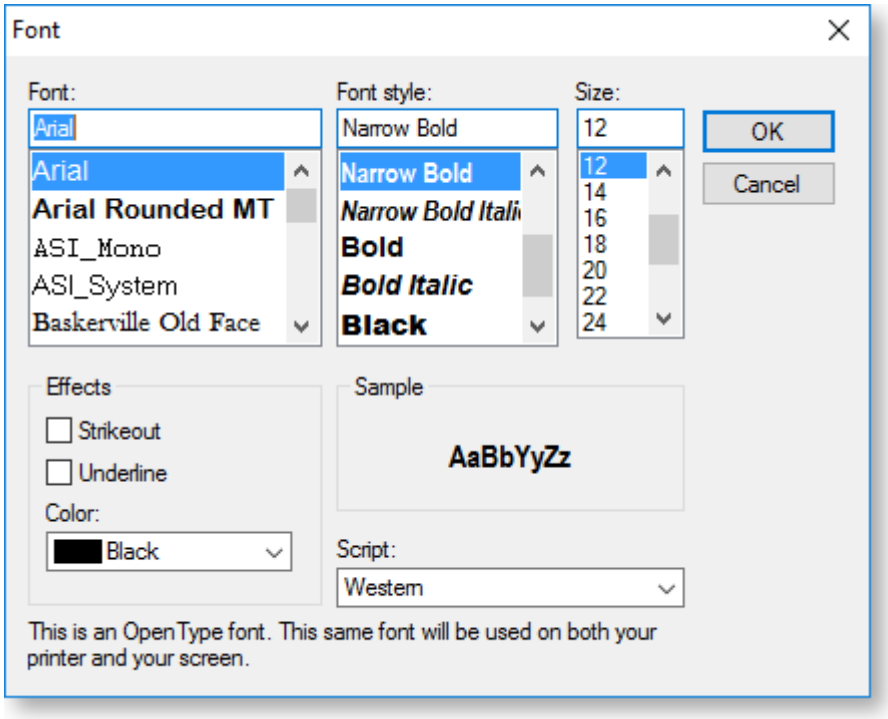
- **All** - create a label component and a data component for each item.
- **Labels** - create label components only.
- **Fields** - create field components only.

When "All" is selected, you can choose between a Vertical and Tabular layout.

In a vertical layout, label/data component pairs are arranged vertically, with each pair on one line:

In a tabular layout, component pairs are arranged horizontally with each pair in one column:

The Layout tab also contains **Grid** and **Font** controls for labels and fields. Tick the **Grid** boxes to display grid outlines around label and/or field components. Click the **Font** buttons to set up font options for label and field components:



Editing Report Components

Once a component has been added to a report using one of the Design tab toolbars, its position, size and properties can be edited in various ways.

Selecting Components

When the Select tool is enabled (by clicking the  toolbar button), report components can be selected individually or in groups, so that they can then be re-positioned or resized. You can also edit selected components' properties on the Report Tree.

You can select or deselect a single component by clicking on it; alternatively, you can select or deselect multiple components by clicking on an empty area of the canvas and dragging a rectangle around the components you want to select - any component that is touching the rectangle will be selected. Clicking on an empty area of the canvas clears all selections.

You can use the SHIFT and CTRL keys to be make more sophisticated selections:

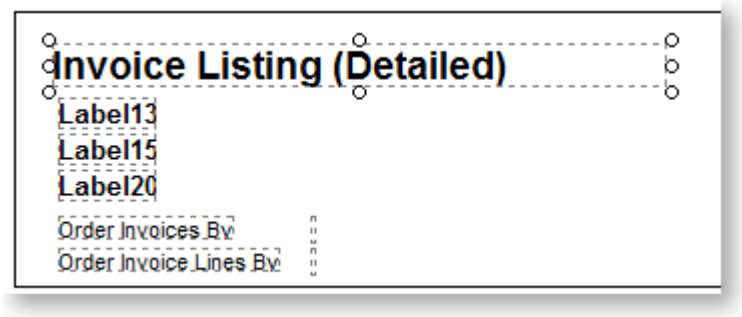
- Where multiple components are layered on top of each other, holding CTRL and clicking on a component selects that component and all components underneath it.
- Selecting multiple components that are positioned on top of a larger component can be difficult - if you try to drag a rectangle around them, you will instead select and drag the larger component. Holding CTRL and dragging a rectangle selects all components that touch the rectangle, without affecting whatever is under the pointer when you first click to start dragging.
- Holding down SHIFT while clicking on a component adds that component to the current selection, or deselects it if it's already selected. Any other components that are already selected remain selected.
- Holding down SHIFT and dragging a rectangle around components adds those components to the current selection, or deselects them if they're already selected. Unlike normal dragging, which selects all components in the rectangle, regardless of whether they were already selected or not, SHIFT+dragging reverses the selection of all components in the rectangle, i.e. unselected components become selected and selected components become deselected.

You can hold CTRL and SHIFT to combine both functions, e.g. CTRL+SHIFT+click selects a component and all components underneath it, adding them to the current selection.

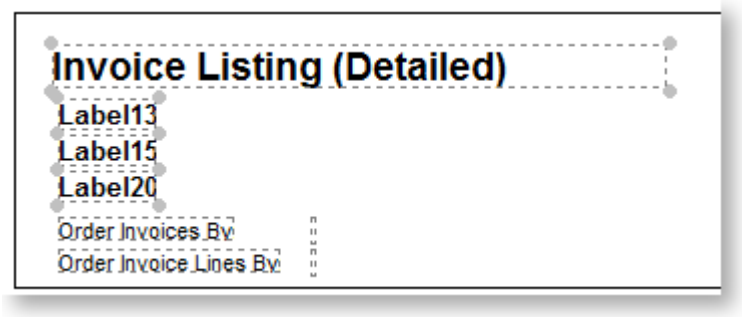
Moving and Resizing Components

You can drag selected components to move them around the canvas. You can also drag the corner and edge controls of a single component to resize it.

When one component is selected, the controls on its edges and corners are white circles - you can click on any of these controls and drag to resize the component:



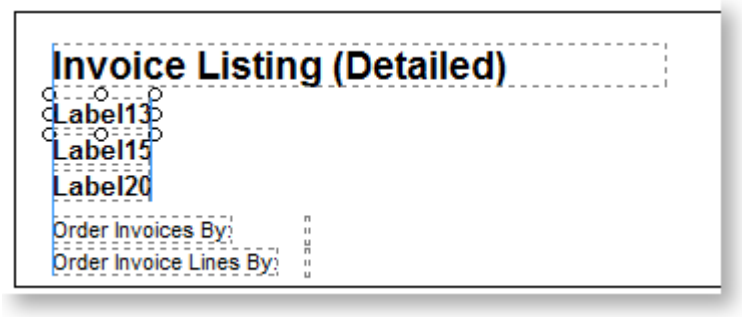
When multiple items are selected, the controls become grey circles - you can't resize in this case, but you can move all of the components together.



You can also move and resize components using the buttons on the Size, Nudge and Align toolbars.

> **Note:** Keyboard shortcuts are available for some of these functions: hold CTRL and use the arrow keys to nudge selected components; hold SHIFT and use the arrow keys to resize selected components.

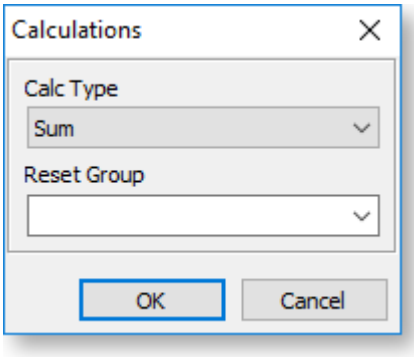
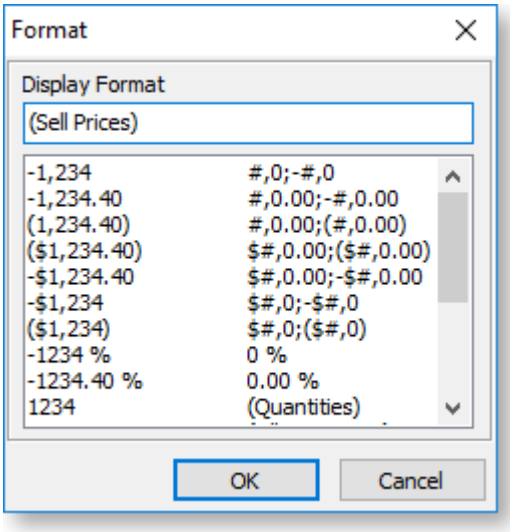
If design guides are turned on (by selecting **Design Guides** from the View menu), blue guide lines appear when dragging components, so that you can align them to other components:

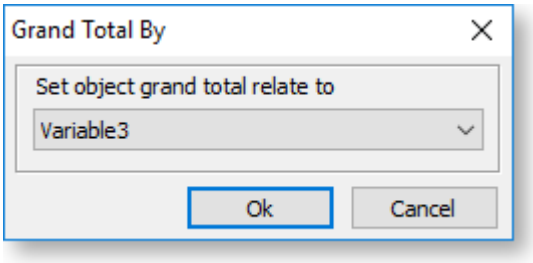


Right-Click Options

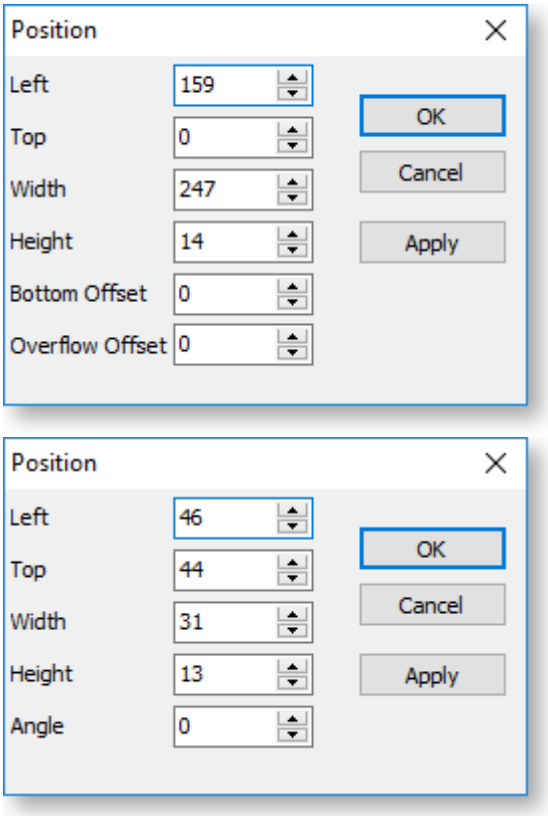
You can right-click on a report component to access more options around positioning and formatting them. Depending on the type of component, different options are available.

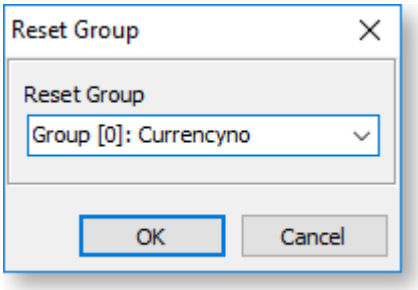
Option	Description
Bring to Forward Bring to Front Send Backward Send to Back	These options control the order of components that are layered on top of each other - they behave the same way as the options on the Edit menu.
Row Column Cell	These options apply to Table components - they let you set up the layout of the table.
CharWrap	This option is available for Memo and DBMemo components - if it is disabled, words will wrap to the next line if they do not fit. If it is enabled, words will be split across lines.
ForceJustify	This option is available for Memo and DBMemo components with justified text - if it is enabled, the last line of the memo will be justified.
Edit	This option is available for Rich Text and DBRichText components - it opens a Rich Text Editor window where you can edit the text of the component.
Edit HTML	This option is available for HTML components - it opens a window where you can edit the HTML text of the component.
Edit Chart	This option is available for chart components - it opens a window where you can set up the contents and appearance of the chart.
Export Chart	This option is available for chart components. For charts that are not bound to a data source, this window lets you export the chart as a bitmap image, a .WMF metafile or an .EMF enhanced metafile.
KeepTogether	This option is available for Memo and DBMemo components - it controls how the component should behave it the memo's contents do not fit on one page. If it is disabled, the contents are printed to the bottom of the available space, then continue on the next page. If it is enabled, the entire memo will be moved to the next page.
Lines	This option is available for Memo and DBMemo components - it opens a Memo Editor window where you can edit the text of the component.
Clear	This option is available for Image and DBImage components - it clears the component, removing the graphic that has been chosen for it.
DirectDraw	This option is available for Image and DBImage components - it specifies whether the image will be sent directly to the printer (enabled) or to an intermediate bitmap and then copied to the printer (disabled). If you are having difficulty getting an image to print correctly on the printer, try enabling this property.
Picture	This option is available for Image and DBImage components - it opens a window where you can select the graphic file to display.
MailMerge	This option is available for Rich Text components - if it is enabled, the component can be used for mail merges.

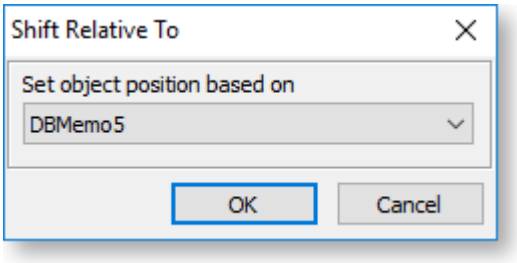
AutoSize	If this option is enabled, the component will resize to fit its contents. If it is disabled, it will remain at whatever size you make it - if it is too small to fit its contents, the contents will appear truncated.
AutoSizeFont	If this option is enabled, text on the component will resize to fit the size of the component.
BlankWhenZero	If this option is enabled, the component will not display zero values - if the value is zero, the component will simply appear blank.
Calculations	This option is available for Variable and DBCalc components. For Variables, it opens the Calculations window, where you can set up the calculation the component should perform. For DBCalc components, it opens a smaller window where you can specify the kind of calculation that should be performed on the data source that the component is bound to: 
DisplayFormat	This option is available for data components. It opens a window where you can select how the data should be displayed. The available options depend on the type of data, e.g. for currency data, there will be options for currency formats, for date/time data, there will be options for date formats, etc. 
GrandTotal	This option is available for Total Variable components - it opens a window where you can select the variable that will be totalled.

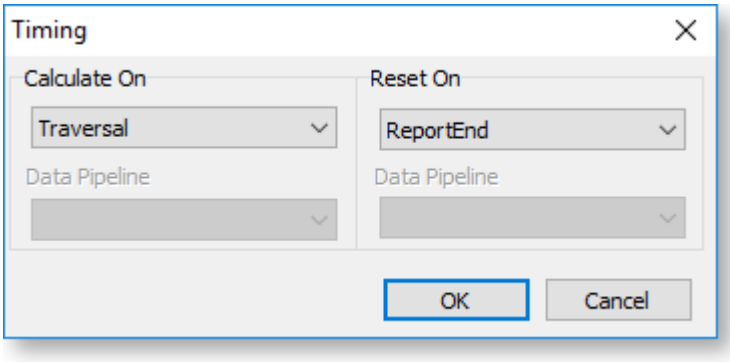


GraphicType	For DBImage components, this option lets you specify the image file type that the component will display.
LookAhead	Enabling this option allows a component to display data that is calculated further down the report, e.g. displaying summary calculations in the Title band, or page footer calculations in the Header band.
Configure	For Barcode components, this option opens a window where you can select a barcode standard and set display options. For CrossTab components, this option opens the Configure CrossTab window where you can set up and format the CrossTab.
MaintainAspectRatio	This option is available for Image and DBImage components - if enabled, the image will retain its width-to-height ratio when it is resized.
Pagination	This option specifies how CrossTab should display when split across multiple pages.
ParentHeight	This option is available for Shape components - it resizes the shape so that it fills the entire height of the band it appears in.
ParentWidth	This option is available for Shape components - it resizes the shape so that it fills the entire width of the band it appears in.
Position	This option opens a window where you can specify the exact position of the component. The units on this window are whatever unit was selected from the Report > Units sub-menu. The available options depend on the kind of component being edited - for some components, you can specify an **Angle**, which allows you to rotate the component.



ReprintOnOverFlow	This option applies to components that are set to stretch with the parent component. If this option is enabled, then when the parent component overflows onto a new page, this component will be reprinted on the new page as well.
SuppressRepeatedValues	This option is available for data-aware components. If it option is enabled, the component will not display repeated values in the dataset.
ReprintOnSubsequent	If repeated values are supressed for a component, then the component will only display a repeated value once in the entire report. Enabling this option means that the value will appear once on each page.
ResetGroup	For components in a group, this option lets you set the control group for the component. If the group breaks across a page, the component will be reprinted with it. 
ShiftRelativeTo	If this option is enabled, then when the component's parent band or region resizes, the component will move to stay in the same position relative to another component. When the option is first enabled, a small window opens where you can select the component that this component will shift relative to.



Stretch	If this option is enabled, the component will resize to fit its contents.
StretchWithParent	If this option is enabled, the component will resize as its parent band or region resizes.
Style	This option is available for CrossTab components - it determines how the CrossTab's captions appear when it is split across multiple pages.
Timing	This option is available for Variable components - it determines at what point in the report generation process the variable should be calculated.  **Note:** To perform calculations based on the results of other calculations, use the Calc Order window (right-click on the report band that the variable appears in and select **Calc Order**).
Transparent	If this option is enabled, the component will not display any foreground colour, allowing any components behind it to show through.
Visible	If this option is enabled, the component will appear on the report; if not, the component will not appear on the report, although it will still be visible on the Design tab.
WordWrap	If this option is enabled, the component's text will wrap to a new line if it does not fit horizontally. If it is disabled, text will be truncated once it reaches the edge of the component.

Calculations and Programming

Programming Basics

Exo Clarity uses a small subset of the Delphi programming language. While limited in one sense, this subset still allows you to perform complex calculations, string manipulation, database interaction as well as many built-in Exo Business-specific functions.

In order to use the Calc tab effectively, it is important for you to become familiar with a few concepts.

Data Types

There are several fundamental data types used in databases and programming, but we will only describe five main ones here very briefly.

Integers are whole numbers with no decimal portion, e.g. an account number.

Floating Point numbers (or “Floats”) are like integers, but they do allow numbers after the decimal point, e.g. an amount field. You may see floats being called “Extended” in Exo Clarity.

Boolean values have two possible values, True or False (or on/off, 1/0, Y/N - they are all just representations of the same thing).

Date / DateTime these data types store years, months, days, and for DateTime hours, minutes, seconds and fractions of a second. Sometimes you only need to work with a date (you can use “Date”); other times you only need to work with a time (You can use “DateTime”, and the date part will be irrelevant).

Strings are just text. Even if you have a string with some numbers in it, you can't do any mathematical operations on the numbers without forcing the computer to interpret it as such. Look at some of the conversion functions to convert string values if you need to.

Objects

In the simplest sense, Objects are just abstract entities that represent "things". Your report is an object. A label on your report is an object. A picture on your report is an object. Lines, regions and anything else that make up your report are also objects.

Properties

Objects have properties that either store information about the object, or determine how they look or behave. For example, your report has a property called **PageNo** that can be used by your code at any time during the generation of the report to tell you which page it is currently on. Be aware that properties also have a data type associated with them.

Events

Objects also have events tied to them. Events are granular steps in the generation of your report, and each time an event happens, an "event handler" is run if it exists. Event handlers are pieces of code that tell Exo Clarity what to do when the event happens. An example of an event is a label's "OnGetText" event, which allows you to manipulate the label text at runtime. You might add an event handler to change the label text based on some other parameter, or to update a counter.

Variables

In Exo Clarity the term "variable" applies to two different things:

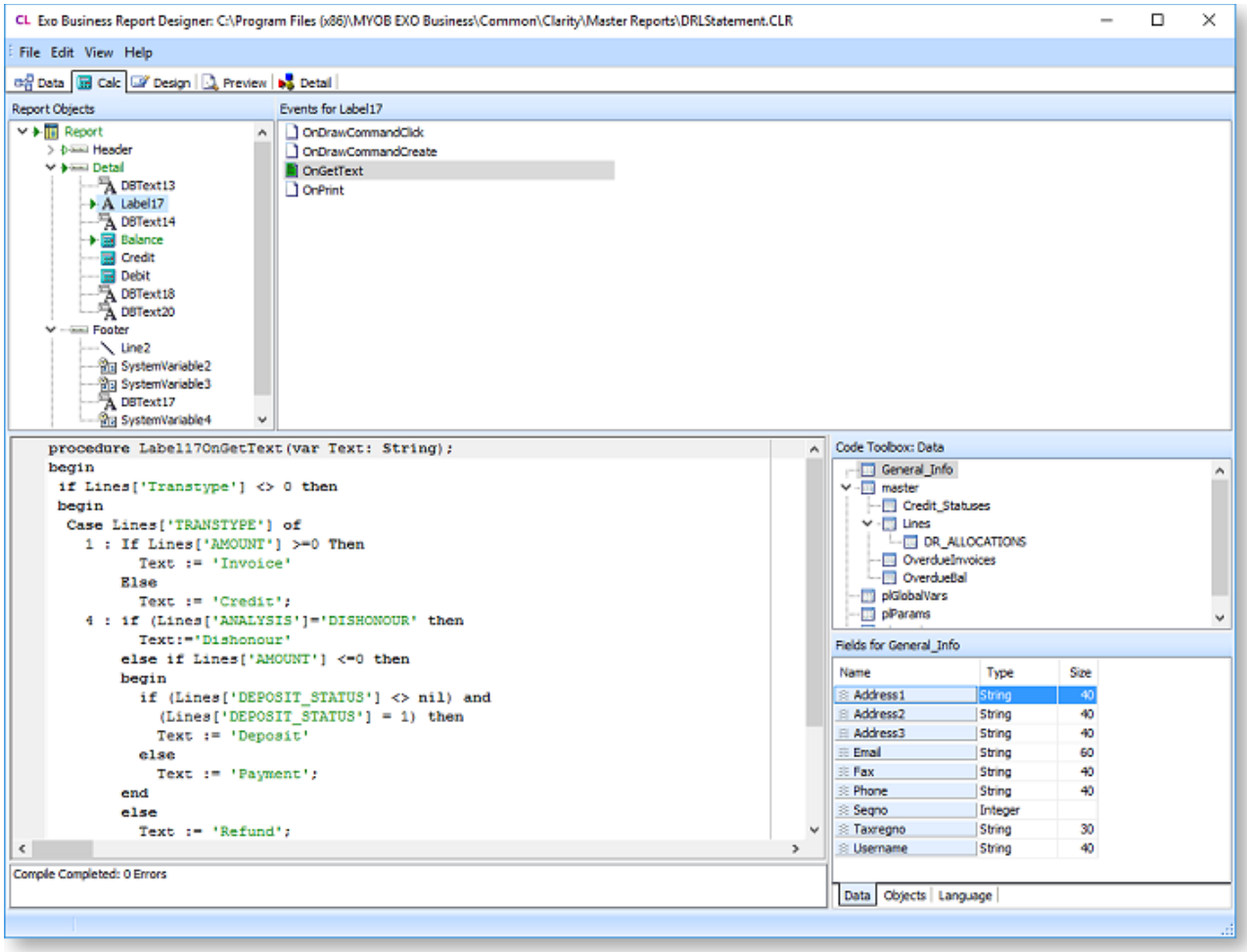
- A *component* that you can add to your canvas from the Standard Components toolbar.
- A *value* that is defined and manipulated by code in the report. This value can be a counter, a text string, or a list of strings. If you want to display the value of the variable in the report,

you need to manually assign it to an existing component on the report, such as a label. Like Object properties, variables also have a data type associated with them.

Note: You can set up *global variables* on the **Calc Tab** - these are available everywhere from anywhere in the report, which means they can be used in any functions or procedures. It also means they can be passed between the main report and any **sub reports**.

The Calc Tab

The Calc tab lets you enter code to control how the report works in very fine detail. It gives access to all parts of the report and all of the elements that it contains, and lets you enter code that affects them - this can mean calculating a value or controlling the report's behaviour, e.g. hiding or showing elements of the report or performing certain actions when parts of the report are clicked on. For reports that contain sub reports, tabs are available at the bottom of the Calc tab, so that you can specify code for each sub report individually.



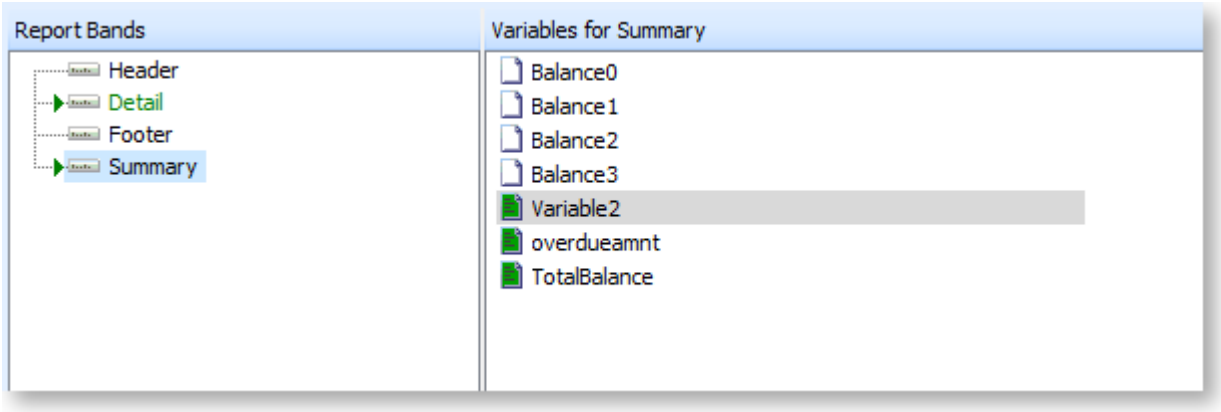
Note: Writing code for a report requires a good degree of familiarity with Exo Business database structures and knowledge of the Delphi programming language.

Tree View

The top section of the Calc tab displays a tree view the controls the kind of data you can . The tree has three modes: Variables, Events and Module. You can swap between modes by right-clicking on the tree or by selecting an option from the View menu.

Variables

In Variables mode, the tree displays each report band that is visible in the report. When a band is selected, all Variable components on that band appear in the pane on the right, allowing you to set the calculation code for multiple variables quickly and easily.

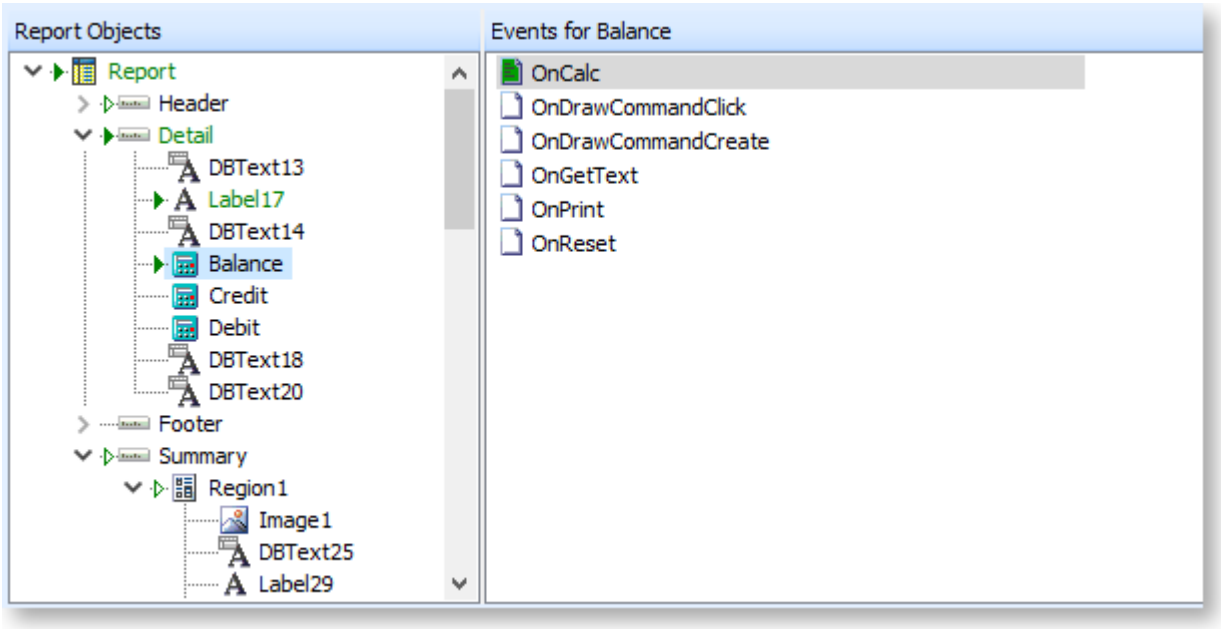


Variables that already have code entered for them are highlighted in green; similarly, any report bands that include variables that have code entered for them are also highlighted green.

> **Note:** You can **edit the code for an individual variable** from the Design tab by right-clicking on the variable component and selecting **Calculations**.

Events

In Events mode, the tree displays all report bands and the components they contain, similar to the Report Tree on the Design tab. When a report component is select, all events that can be triggered for the component appear in the pane on the right - these can include events such as when the component is drawn, when it is clicked on or when its value is calculated. This lets you control how the report should behave whenever one of these events occur.

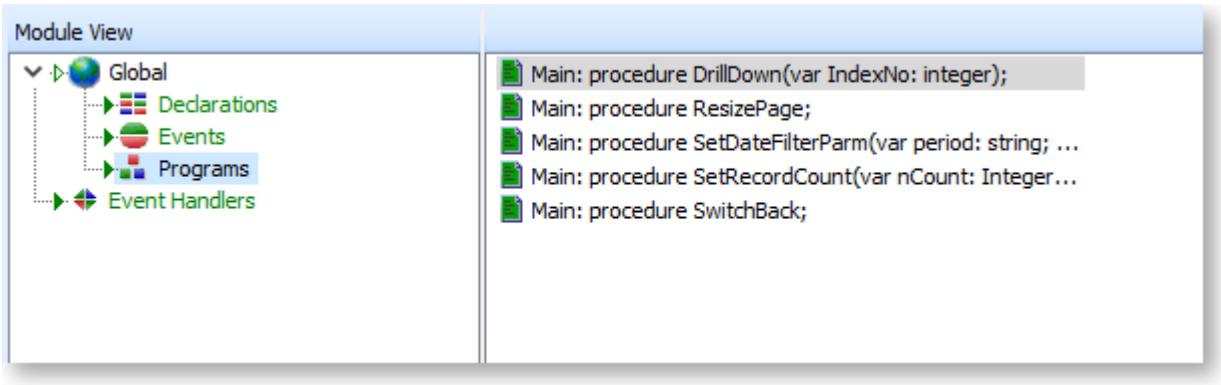


Events that already have code entered for them are highlighted in green; similarly, any report components that include variables that have code entered for them are also highlighted green.

Module

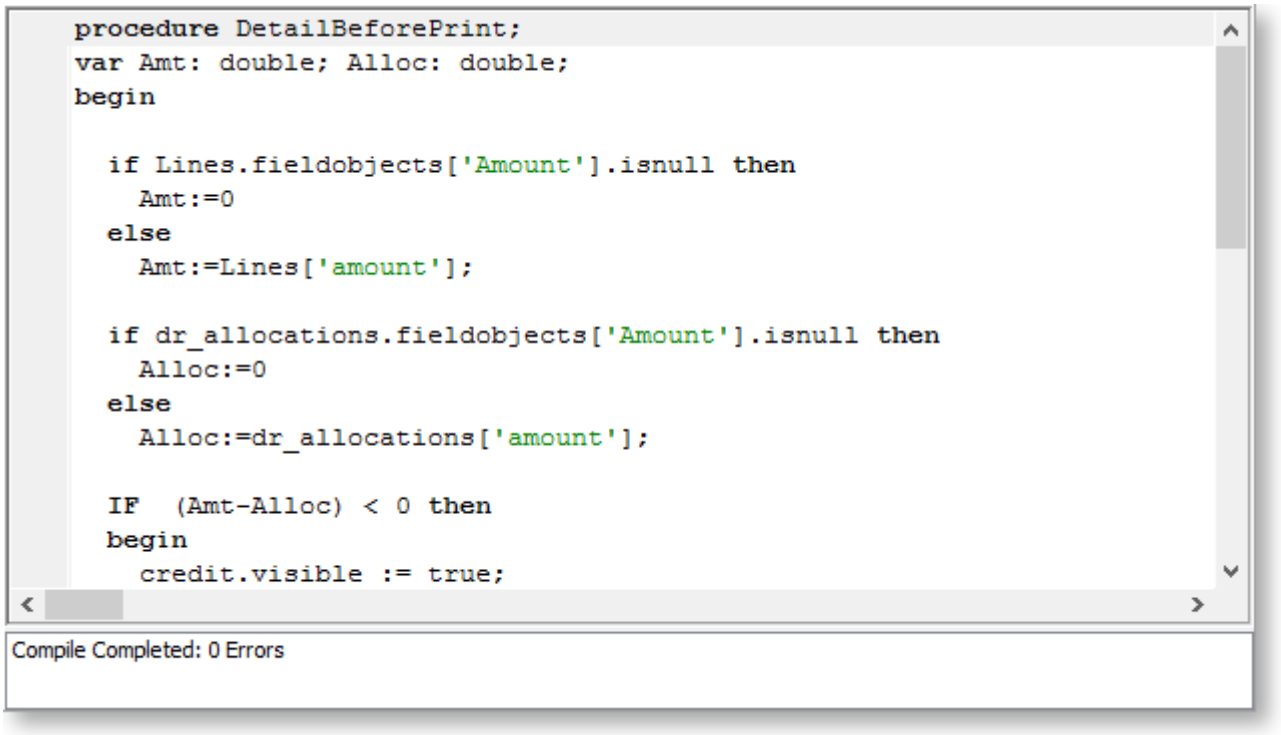
In Module mode, the tree shows all code that has been entered for the report so far, and lets you edit any existing code. The tree has four sections:

- Global > Declarations - Lets you enter global constants and variables for the entire report. Declaring a variable or constant here will ensure that you will be able to access it from any functions or procedures. It also means it can be passed between the main report and any sub reports.
- Global > Events - Lists the two global events for the report: OnCreate, which is triggered as soon as the report is run, and OnDestroy, which is triggered when the report is closed (or when it finishes printing if there's no user input).
- Global > Programs - Lets you enter global functions and procedures that can be reused elsewhere in the report. (A function is a piece of code that returns a value, while a procedure performs actions but does not return anything.) To create a new global function or procedure, right click on the list of functions and procedures and select **New Function** or **New Procedure**.
- Event Handlers - Lists all events that currently have code written for them. You can edit the existing code in this mode - to enter code for a new event, you must use Events mode.



Code Pane

The lower left area of the Calc tab is where you enter your code. When you first click in the code pane, the outline of a function, procedure or variable declaration will be filled in for you as appropriate.



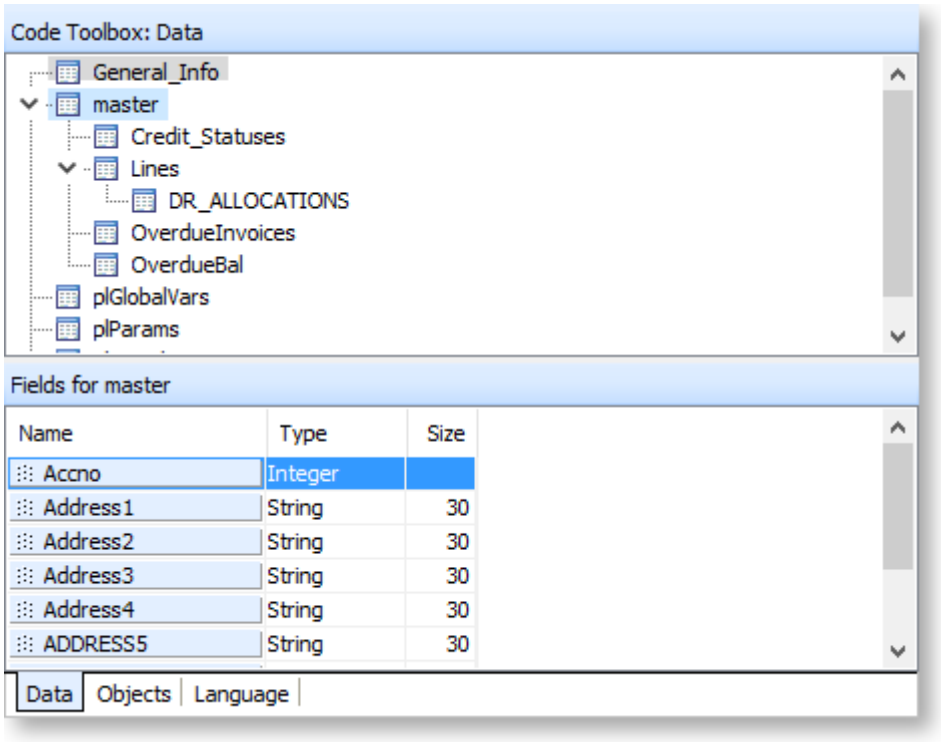
If the messages pane is enabled (select **Messages** from the View menu), any compilation errors with your code will appear at the bottom of the code pane.

Code Toolbox

The lower right section of the Calc tab contains a library of elements and code templates that you can drag and drop into the code pane to save typing time. The toolbox is divided into three tabs: Data, Objects and Language.

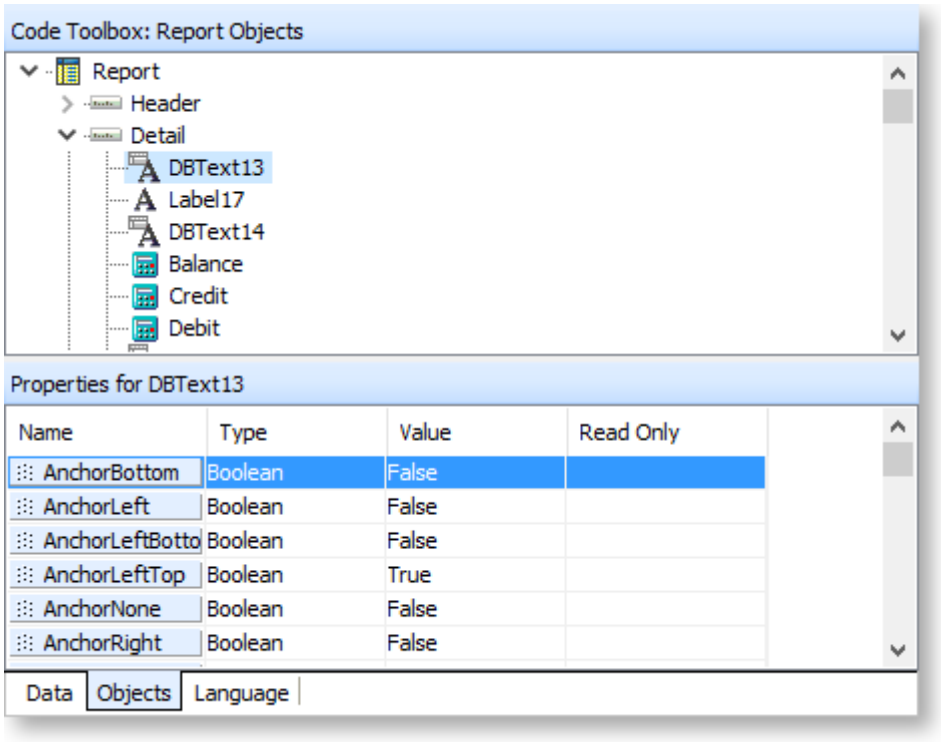
Data Tab

This tab lists all data sources that have been set up for the report. Clicking on a data source displays all fields in that source - you can drag a field into the code pane to insert code for that field.



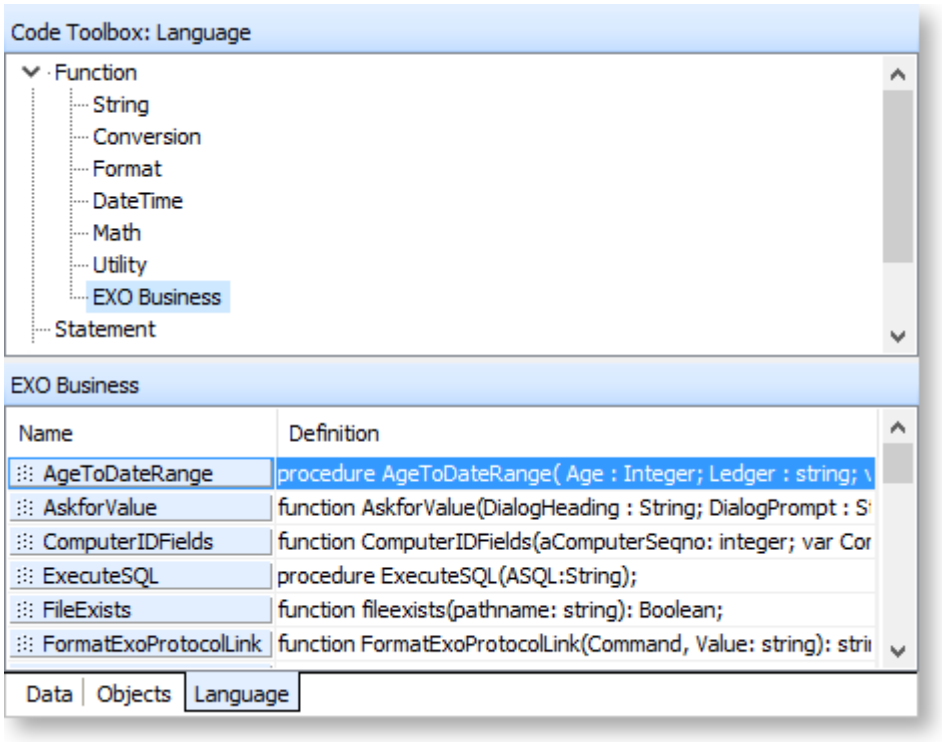
Objects Tab

This tab displays all report bands and the components they contain, similar to the Report Tree on the Design tab. Clicking on a component displays all of its properties - you can drag a property into the code pane to insert code that defines that property.



Language Tab

This tab contains a library of pre-defined functions and values that you can drag and drop into the code pane. Functions and values are organised into groups a tree structure - click on any group to display the functions or values in that group. Each function or value has a definition which describes its properties or inputs and outputs.



Calc Tab Menus

File

This menu offers standard file operations:

- **Save** - Saves the current report.
- **Save As** - Saves the current report, giving you the option to enter a new name for the report and select its report type.
- **Close** - Closes the Exo Clarity application.
- **Import** - Imports saved code from a .CTM file into the Calc tab, replacing the current code.
- **Export** - Exports the current setup of the Calc tab to a .CTM file. This file can be imported or merged into another report, to save you having to set up the same code multiple times.

Edit

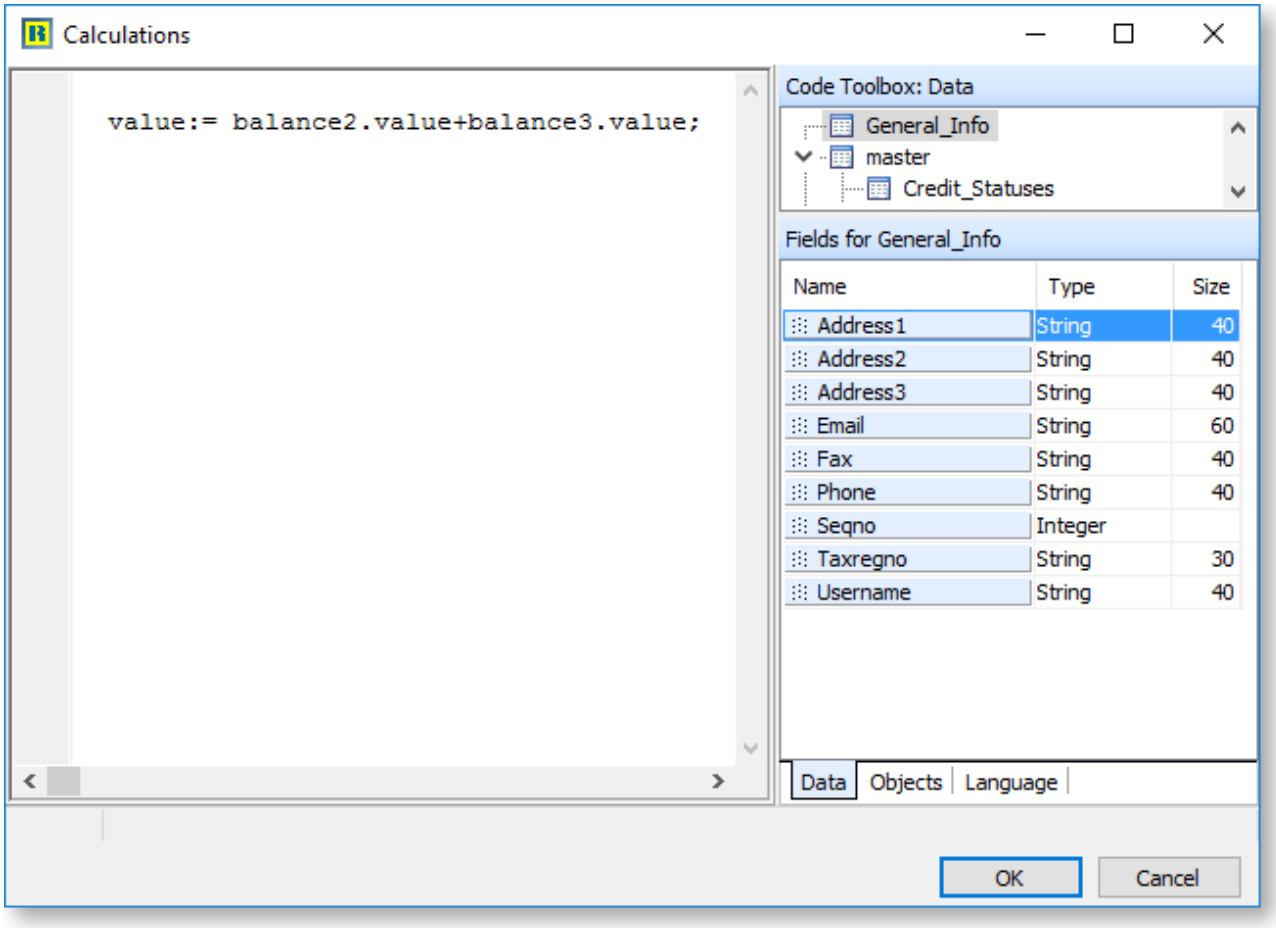
This menu contains standard options for undoing edits as well as cutting, copying, pasting and deleting text.

View

This menu lets you control the layout of the Calc tab by swapping the tree view between Variables, Events and Modules views and by turning the ode Toolbox and Messages pain on and off.

The Calculations Window

Right-clicking on a Variable component and selecting **Calculations** opens the Calculations window, where you can enter code to calculate the value of the variable:



This window contains a subset of the controls on the Calc tab - since it is specific to the variable that was clicked on, there is no need for the tree view. The Code Toolbox is fully available, however, so you can use any of the data sources, objects and functions to set up the calculation for the variable.

The calculations that appear here are also available on the Calc tab when the variable is selected from the report tree, and any calculations entered on the Calc tab for the variable will appear on the Calculations window.

Language Reference

Basic Delphi Syntax

Comments

As a starting point you should know how to make comments. A comment is text that is ignored by Exo Clarity and is simply for making a note about the code. This can be to explain how a complex piece of code works or perhaps give a reason for doing something unusual in the code. This helps both you and others after you who may come back to the code long after it was written to fix or modify it.

Don't make comments too long-winded - you don't have to explain every single step, just potentially confusing ones. It's also a good idea to add the date of the change and name or initials to the comment if you are making modifications to an existing report.

You can enclose a comment in curly braces - { }. It is best to put comments at the end of a line of code, or on a line on their own. For example:

```
Value := 1;  {This is a comment here, after a line of code}
             {This is a comment on its own line}
```

Variables

Variables need to be declared before they can be used. Use the var keyword to declare variables. The format is:

```
var variableName : type;
```

where "type" is one of the defined data types.

Semicolon and Layout

The semicolon is a statement separator. Every clause must end in a semicolon, with a few exceptions which are noted below. While you could put all your code on one line (the semicolons help Exo Clarity know where one statement stops and the next one starts), it is not very practical to do so. Don't be afraid to give your code plenty of space, lay it out neatly, and indent properly. This helps with readability.

Delphi uses begin and end to create *code blocks*. A code block is a logical sequence of commands that occur one after the other. You need to use the begin and end with all major structures in Delphi when you need to do more than one command. A code block always ends in a semicolon, and each clause within the block (including embedded blocks) must end in a semicolon.

A procedure block (same for functions):

```
procedure MyProcedure(var inputvar: string);
var
      a,b,c : integer;
begin
      {Do something with a,b,c and inputvar here}
end;
```

Notice that the variable declaration occurs before the procedure code block. In the code examples in this document, the procedure header, begin and end keywords are mostly omitted.

Operators

Operators are the symbols in your code that enable you to manipulate all types of data. The various types of operators that are most commonly used are explained below. Note the use of comments in this document is to explain the concepts, and commenting as verbosely is not usually necessary.

Assignment Operators

To assign a value to a variable use the **:=** operator.

Example:

```
Variable1.Value := 5;              {assigning value of 5 to a variable
component}
Dbtext2.text := Master['Name'];   {assigning a value of a database field
to a dbtext component}
```

Comparison Operators

Comparison operators consist of **=** (equals), **<** (less than), **>** (greater than), **<=** (less than or equal to), **>=** (greater than or equal to), **<>** (not equal to). These are most often used in "if" statements.

Example:

```
if dr_trans['transtype'] = 1 then        {Comparison used}
         Variable2.value := 'Invoice';   {Assignment used}
```

Note: The **:=** operator is used to assign a value to a variable. The **=** operator compares the values of two operands.

Logical Operators

AND and OR logical operators are most commonly used as a part of an "if" statement or loop as demonstrated in the following examples:

```
if (a = 2) AND (b < 3) then
          { ... Do something ... }
if (a = 2) OR (b = 2) then
          { ... Do something ... }
```

Note: If you have an "if" statement that makes multiple comparisons, make sure that you enclose each set of comparisons in parentheses, as shown above, or it may not work correctly.

Arithmetic Operators

These are standard mathematical operators used in the same way you would on a calculator.

	Operator
Addition	+
Subtraction	-
Multiplication	*
Division	/
Modulus	Mod

Because of Delphi's *operator precedence,* the order in which arithmetic operations are calculated is not necessarily the same as the order you type them, just like on a scientific calculator. You can use parentheses to group operations, and the parenthesised operations will happen first. While operator precedence is outside the scope of this document, two examples are used below to illustrate:

```
A := 1 + 2 * 3;     {A is 7 after this statement}
A := (1 + 2) * 3;   {A is 9 after this statement}
```

String Concatenation Operator

Strings also have an operator, the concatenation operator, which is the same as arithmetic addition:

```
Text := 'Dear ' + Dr_Contacts['Firstname'] + ' ' +
Dr_Contacts['Lastname'];
```

If you need extra text or more than one data field in a line, this is the best way to do it. It makes the report look a lot tidier than it would if you just dumped the fields next to each other.

Testing Conditions

Testing conditions allow you to give your code some "flow control":

If Statements

An IF statement enables you to determine whether certain conditions are met before executing some code. If statements don't need semicolons, but the statements after them (or blocks of statements) do.

Example:

```
if x = 4 then    {no semicolon here}
       y := x;  {there's a semicolon here because it ends the if
statement}
```

Use the **begin** and **end** keywords if you want to execute multiple lines of text when a given condition is true. This is called a "code block".

Example:

```
if x = 6 then
   begin
       DoSomething;     {semicolons after each of these lines in the
block}
       DoSomethingElse;
       DoAnotherThing;
   end;                 {semicolon after the end keyword}
```

You can combine multiple conditions using the IF … ELSE construct:

```
if  x = 100 then
     Somefunction             {no semicolon here for a single line of
code}
else if x = 200 then begin
          Someotherfunction1;
          Someotherfunction2;
     end else begin
          Somethingelse1;
          Somethingelse2;
     end;                     {End of the if statements, so there's a
semicolon here}
```

Case Statements

A case statement provides a means for choosing one condition among many possibilities without needing lots of embedded “if…else” constructs. Case statements only work with numeric values though, not strings of text.

Example:

```
case Cr_Trans['Transtype'] of   {Depending on this value, one of these 4
lines are run}
     1:  Text := 'Invoice';
     4:  Text := 'Payment';
     5:  Text := 'Adjustment';
     else Text := '';
end;
```

The TStringList (1-dimensional String Array)

This is the only type of "array", as it is known to programmers, that is available for use in the Clarity Calc tab. It is a data type you can use anywhere in your code to hold a list of string values that can be referenced by index in the list. You add a new item to the list as follows:

```
var MyStringList : TStringList;
begin
    MyStringList.Add('This is the first item in my list');
    MyStringList.Add('This is the second item in my list');
    ...
```

You can then access any item in the list like this:

```
    ...
    ShowMessage(MyStringList[0]); { This displays the first item }
    ShowMessage(MyStringList[1]); { This displays the second item }
end;
```

Notice the square brackets after MyStringList – you must use square brackets for this. If the entry you request from the string list does not exist you will get an error (e.g. if you were to have another line referencing MyStringList[2] you would get an error). TStringLists use a "zero-based" index, because the first item is always zero.

For example, four TStringLists are used in the **DashTopCustbyMargin.CLR** standard dashboard report. These are global variables; they are defined in the Definitions section of the Calc tab's tree view, empty records are created in the list (before they are needed) in the GlobalOnCreate event handler, then they are populated in the DetailAfterGenerate event handler as the data is retrieved. A description of each is below:

- **gslAccNos** – this is used to store the account numbers (ACCNO field) of each account that is shown in the list, so that when one of the bars on the chart is clicked the ValueIndex of the clicked bar corresponds to an entry in the list that holds the clicked account's AccNo which is then applied to the filter for the drill-down data pipeline, then the pipeline can be refreshed to show the data for that account.
- **gslHeadAccNos** – this is used for the special case where an account in the list has a HEADACCNO field set – i.e. it is a sub-account of another account. It will not be discussed here.
- **gslAccNames** – this is used to store the account name so that it can be used to set the chart heading of the "drill-down" chart showing the customer sales history. It is easier to store it with the other values than pick it up when the database is re-queried.
- **gslAlphaCode** – similar to the account name, this is used to store the alpha code of the account for use in the history chart heading.

Events

You can control the functioning of a report in great detail by adding to code the events that fire at various points in the report generation process. Different parts of the report have different events - the available events for each part are described below.

> **Note:** You can use the **ShowMessage() function** to see the order in which events fire. Enter `ShowMessage('[Write the event name here]');` into the code pane of each event. A message box will then pop up when the event is fired.

Global Events

Events for the report as a whole can be edited at Global > Events when the Calc tab's tree view is set to "Module".

Event	Description
OnCreate	This event is triggered when the report is first created.
OnDestroy	This event is triggered when the report is closed.

Report Events

Events for the main report and each sub report can be edited on the Calc tab when the tree view is set to "Events".

> **Note:** Events highlighted in blue are available for the main report and sub reports; all other events are available for the main report only.

Event	Description

AfterAutoSearchDialogCreate	This event fires after the Search window on the Preview tab has been created, but before it has been populated.
AfterEmail	This event fires after an email has been sent either programmatically from within the report or by selecting to send an email from the Print Preview window.
AfterOpenDataPipelines	This event fires after the report engine opens all of the data sources associated with the report, sub reports, and data-aware controls.
AfterPrint	This event fires after the print process has been completed. When printing to the printer or to the file, AfterPrint fires after the print job has been closed or after the file has been closed. When printing to the screen, AfterPrint fires after the Print Preview form has been closed.
BeforeAutoSearchDialogCreate	This event fires before the Search window on the Preview tab is created.
BeforeEmail	This event fires before an email has been sent either programmatically from within the report or by selecting to send an email from the Print Preview window.
BeforeOpenDataPipelines	This event fires before the report engine opens all of the data sources associated with the report, sub reports, and data-aware controls. You can use this event to apply custom parameter values to SQL or generate custom SQL.
BeforePrint	This event fires before the print process begins.
OnAssignPreviewFormSettings	This event fires before the Preview form settings are assigned to the Preview form. Provides the proper timing to assign them new values before they are transferred to the preview form. You can use the OnPreviewFormCreate to control the Preview form's properties directly, as the OnPreviewFormCreate event fires after the settings have been transferred.
OnAutoSearchDialogClose	This event fires after the Search window on the Preview tab has been closed.
OnCancel	This event fires when the user cancels the print process.
OnCancelDialogClose	This event fires when the Cancel window is closed. This window is closed either when the user clicks the **Cancel** button, or when the printing process is complete. The Cancel window is only displayed when printing to the printer or to a file.
OnCancelDialogCreate	This event fires after the Cancel window has been created.
OnEndColumn	This event fires after a column has completed printing.
OnEndFirstPass	This event fires when the report has completed the first pass. You can save any grand totals or other calculated values at this point.

OnEndPage	This event fires after a page has completed printing. If you set **DonePrinting** to True in this event handler, no further pages will print.
OnEndSecondPass	This event fires after the report has completed the second pass. If you are taking manual control of the print process, and have variables which are no longer needed when the report is complete, you can clear them here.
OnFileDeviceCreate	This event fires after a FileDevice has been created - this is the object that is created when printing a report to a file.
OnGetAutoSearchParameters	This event fires when the report is generated, if AutoSearch values exist for the report.
OnInitializeParameters	This event fires whenever a report is generated. It occurs prior to the AutoSearch events. You can use this event to set parameters and AutoSearch values.
OnNoData	This event fires when no data is found by the data source connected to the report. This is used in some Exo Business reports and dashboard widgets to display a warning message if the user has selected parameters that return no data.
OnOutlineNodeCreate	When enabled, an outline tree structure is dynamically generated by the report engine and rendered by the report previewer. This event fires whenever an outline node is created during the report generation process. It can be used to customize the outline as it is being generated.
OnPageRequest	This event fires after the Print method has been called and all of the currently connected devices have been polled for the pages of the report that are needed. The result of this polling process is summarized into one PageRequest object, which is then passed to the report engine.
OnPreviewFormClose	This event fires after the Print Preview window has closed.
OnPreviewFormCreate	This event fires after the Print Preview window is created.
OnPrintFormClose	This event fires after the Print window has closed.
OnPrintFormCreate	This event fires after the Print window is created.
OnPrinterDeviceCreate	This event fires after a PrinterDevice has been created - this is the object that is created when printing a report to the printer.
OnPrinterDeviceStateChange	The PrinterDevice fires this event while sending pages to the printer.
OnSaveText	This event only fires when printing to a text file.
OnStartColumn	This event fires before a new column begins printing on the report.
OnStartFirstPass	This event fires when the report begins the first pass, when the report performs all calculations.
OnStartPage	This event fires before a new page begins printing on the report.

OnStartSecondPass	This event fires when the report begins the second pass, which formats and prints the report (if "Two Pass" is selected for the **Pass Setting** option in the Design tab's Report menu).

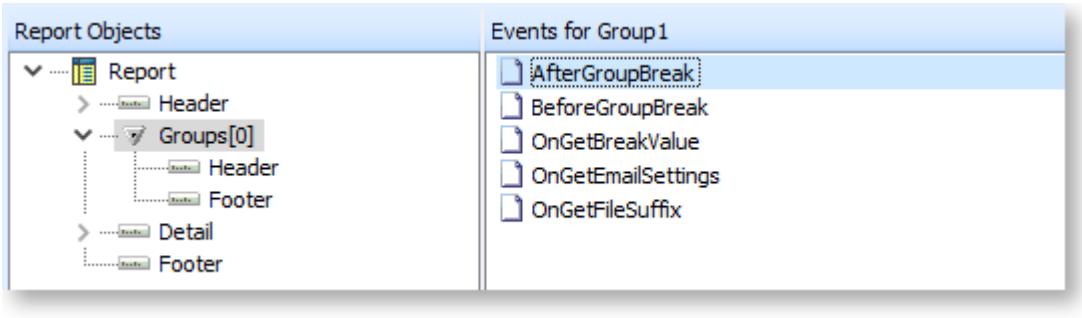
Report Band Events

These events are available for each report band that is in use. They are available on the Calc tab when the tree view is set to "Events".

Event	Description
AfterGenerate	This event is triggered after the band is printed. AfterGenerate differs from AfterPrint in that AfterPrint print fires every time a band has the opportunity to print. Sometimes the band has an opportunity to print, but for various reasons does not (usually printing on the next page instead.) In these cases, AfterPrint will fire, but AfterGenerate will not.
AfterPrint	This event fires after the band has printed.
BeforeGenerate	The BeforeGenerate event fires before the band prints. BeforeGenerate differs from BeforePrint in that BeforePrint fires every time a band has the opportunity to print. Sometimes the band has an opportunity to print, but for various reasons does not (usually printing on the next page instead.) In these cases, BeforePrint will fire, but BeforeGenerate will not.
BeforePrint	This event fires before a band begins printing. Use this event to perform actions before a band prints. One action is to control the visibility of objects which should or should not print in the band. This is the best event to use for setting the visibility of controls in a band.
OnCreateDrawCommand	This event fires each time the band prints on a page. You can use this event to add additional draw commands to the page.

Group Events

When a report uses grouping, you can edit events for each group on the Calc tab when the tree view is set to "Events":



Event	Description
AfterGroupBreak	This event fires before the report breaks for a group.
BeforeGroupBreak	This event fires after the report breaks for a group.
OnGetBreakValue	This event fires each time a group checks the break value (the value of the field that the report is grouped by).

OnGetEmailSettings	If the group's **Email new fIle** property is ticked on the Groups window, a new file will be exported and emailed each time a group break occurs (allowing you to email each group to a different recipient). This event fires each time a new file is created and emailed, and allows you to change the email settings for each group, e.g. specify different email addresses for each group.
OnGetFileSuffix	If the group's **Create new fIle** property is ticked on the Groups window, a new file will be exported each time a group break occurs. This event fires each time a new file is created.

Note: The Header and Footer bands for each group have the same events as the main Header and Footer bands.

Component Events

Each component that is placed on the report has events available for it. The events that are available different by the type of component - the most common events are detailed below.

Event	Description
OnCalc	This event applies to components like the Variable component, where a value is calculated. It fires after the calculation for the component has been performed.
OnDrawCommandClick	This event fires when the component is clicked on in the Print Preview window or the Preview tab.
OnDrawCommandCreate	This events fires when the component is drawn on the report.
OnGetMemo	This event applies to components like the Memo component, where a body of formatted text is displayed. It fires when the component is populated, but before it is drawn. You can use it to manipulate the contents of the component before it appears on the report.
OnGetText	This event applies to components that display plain text, such as labels and variables. It fires every time the component's text is referenced. You can use it to override the component's contents.
OnPrint	The OnPrint event fires before the component appears on the report ("print" can mean printing to a printer, exporting to a file, or appearing on the Print Preview window). Use the OnPrint event when you want to dynamically control the appearance of the report component, e.g. if you want to conditionally display a component by setting its **Visible** property at run-time based on some data field value.

Chart Events

Chart components have several events specific to them - these are discussed in Chart Events.

Drawing on the PaintBox

The PaintBox component acts as a mini canvas that can be drawn on programmatically using the Calc tab. You can place a PaintBox component using the Advanced Components toolbar on the Design tab, then add code to its OnPrint event to create shapes and text that might otherwise be impossible using other report components.

The PaintBox has a single property: Canvas, which supports a variety of methods for adding lines, shapes, text and images to the PaintBox, for example:

```
procedure PaintBox1OnPrint;
begin
  PaintBox1.Canvas.Brush.Color := clRed;
  PaintBox1.Canvas.Pen.Color := clLime;
  PaintBox1.Canvas.FillRect(0, 10, 50, 90);
  PaintBox1.Canvas.Ellipse(0, 10, 50, 90);
end;
```

Note: All coordinates are relative to the bounding box of the PaintBox component. Shapes that are drawn outside the bounding box will not appear on the report.

Setting Drawing Properties

The following methods can be used to set the properties of the things you draw on the PaintBox:

- **Pen** determines the colour and/or width of the outlines of any shapes you draw.
- **Brush** determines the fill colour and/or pattern of any shapes you draw.
- **Font** determines the properties of any text you draw.

Colors can be specified as standard colours, e.g. clBlue or clLime (look at an object's **Color** property on the Design tab to see a dropdown list of available standard colours). You can also specify a colour in RGB values using the RGB() conversion function:

```
PaintBox1.Canvas.Brush.Color := RGB(97,0,165);
```

Adding Shapes and Lines to a PaintBox

The following methods can be used to add lines and geometric shapes to a PaintBox.

LineTo

Draws a line with the current Pen from the current pen position to the specified coordinates. The pen position is the last point that a line was drawn to. It defaults to 0,0 if nothing has been drawn yet, and can be set with MoveTo (see below).

```
PaintBox1.Canvas.Pen.Color := clRed;
PaintBox1.Canvas.MoveTo(10, 10);
PaintBox1.Canvas.LineTo(100, 100);
```

MoveTo

Moves the pen position to the specified coordinates.

Rectangle

Draws a rectangle with the current Pen, filled with the current Brush.

```
PaintBox1.Canvas.Rectangle(0, 10, 50, 90);
```

FrameRect

Draws a rectangle outline with the current Pen.

FillRect

Draws a solid rectangle, filled with the current Brush.

Ellipse

Call Ellipse to draw a circle or ellipse on the canvas. Specify the bounding rectangle by giving the top left point at pixel coordinates (X1, Y1) and the bottom right point at (X2, Y2).

```
PaintBox1.Canvas.Ellipse(0, 10, 50, 90);
```

Arc

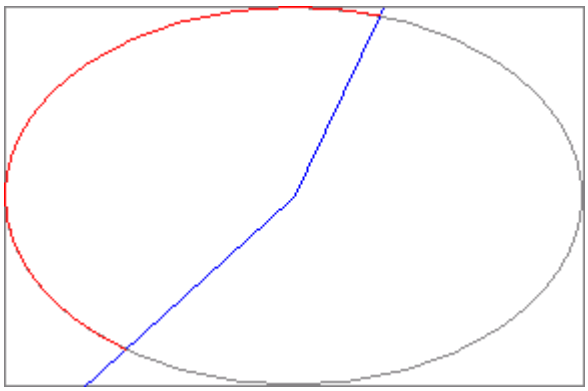
Use Arc to draw an elliptically curved line with the current Pen. The arc traverses the perimeter of an ellipse that is bounded by the points (X1,Y1) and (X2,Y2). The arc is drawn following the perimeter of the ellipse counterclockwise from the starting point to the ending point. The starting point is defined by the intersection of the ellipse and a line defined by the center of the ellipse and (X3,Y3). The ending point is defined by the intersection of the ellipse and a line defined by the center of the ellipse and (X4, Y4).

For example, the following code produces the following arc:

```
procedure PaintBox1OnPrint;
begin
 PaintBox1.Canvas.Pen.Color := clRed;
 PaintBox1.Canvas.Arc(10,10,300,200,200,10,50,200);
end;
```

The following code shows the bounding boxes and lines that define the same arc:

```
procedure PaintBox1OnPrint;
begin
 PaintBox1.Canvas.Pen.Color := clGray;
 PaintBox1.Canvas.Rectangle(10,10,300,200);
 PaintBox1.Canvas.Ellipse(10,10,300,200);
 PaintBox1.Canvas.Pen.Color := clBlue;
 PaintBox1.Canvas.MoveTo(155,105);
 PaintBox1.Canvas.LineTo(200,10);
 PaintBox1.Canvas.MoveTo(155,105);
 PaintBox1.Canvas.LineTo(50,200);
 PaintBox1.Canvas.Pen.Color := clRed;
 PaintBox1.Canvas.Arc(10,10,300,200,200,10,50,200);
end;
```



Adding Text to a PaintBox

The following methods can be used to add text to a PaintBox.

TextWidth/TextHeight

Use to calculate how much space is required to print the specified string in the current font.

TextOut

Outputs the specified string at the specified coordinates in the current Font:

```
PaintBox1.Canvas.Font.Name := 'Arial';
PaintBox1.Canvas.Font.Height := 15;
PaintBox1.Canvas.TextOut(10, 10, 'Hello, World!');
```

TextRect

Outputs the specified string at the specified coordinates in the current font, clipped to a rectangle:

```
PaintBox1.Canvas.Font.Name := 'Arial';
PaintBox1.Canvas.Font.Height := 15;
PaintBox1.Canvas.TextRect(10, 10, 100, 10, 'Hello, World!');
```

Adding Images to a PaintBox

The following methods can be used to add existing image files to a PaintBox.

Draw

Use to draw an existing image on the canvas at the specified coordinates:

```
PaintBox1.Canvas.Draw(0, 0, FormLogo.Picture.Graphic);
```

StretchDraw

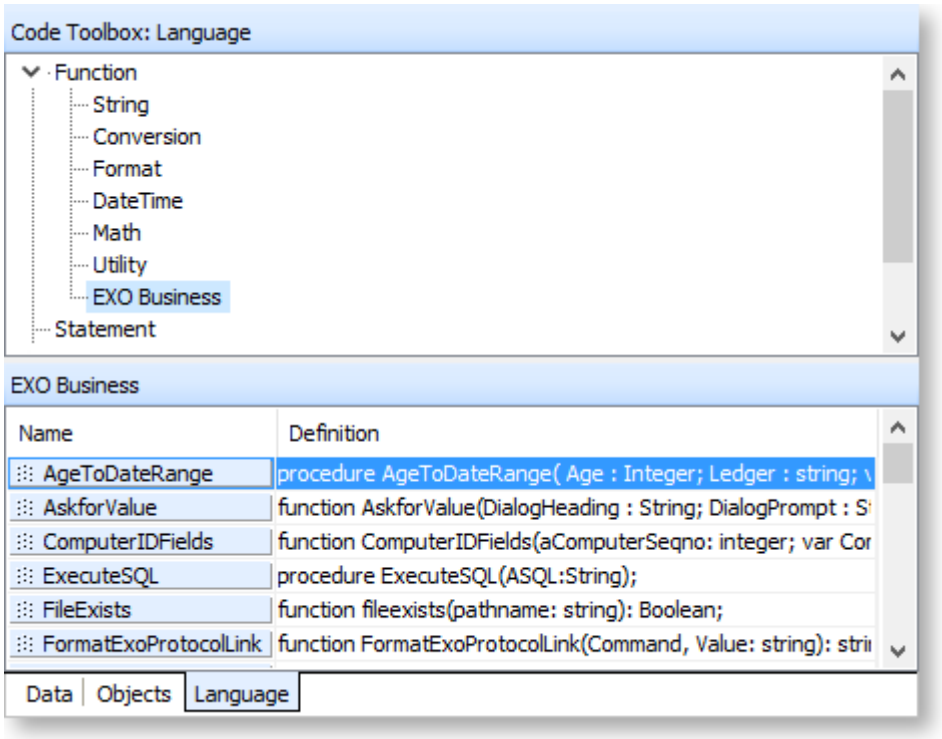
Use to draw an existing image on the canvas at the specified coordinates, scaled to a fixed size:

```
PaintBox1.Canvas.StretchDraw(0, 0, 100, 200, FormLogo.Picture.Graphic);
```

Functions

Available Functions

The Language tab of the code toolbox in the Calc tab provides a list of functions that can be used to manipulate and format the data before being displayed in the report.



The available functions are grouped into sections:

- String Functions
- Conversion Functions
- Format Functions
- DateTime Functions
- Utility Functions
- Exo Business Functions

String Functions

AmountStr

Converts a dollar amount to a string, e.g. the amount 300.50 would convert to "Three Hundred Dollars Fifty Cents".

Declaration: function AmountStr(const aAMount: Double): String;

Capitalize

Capitalizes the first character following a space for every word in a string.

Declaration: Function Capitalize(const S: String) : String;

CompareText

Compares two strings by ordinal value with case sensitivity.

Declaration: function CompareText(const S1, S2: String): Integer;

Copy

Returns a substring of a string or a segment of a dynamic array

Declaration: function Copy(S:String; Index, Count: Integer) : String;

Delete

Removes a substring from a string.

Declaration: procedure Delete(var S: String; Index, Count: Integer);

Insert

Inserts a substring into a string beginning at a specified point.

Declaration: procedure Insert(Soure: String; var S: String; Index: Integer);

Length

Returns the number of characters in a string.

Declaration: function Length(S: String): Integer;

LowerCase

Converts an ASCII string to lowercase.

Declaration: function LowerCase(const S: String): String;

NumberStr

Converts a number to an English string (e.g. for cheques).

Declaration: function NumberStr(const aNumber: Integer): String;

PadLeft

Pads a string with characters up to a fixed length.

Declaration: function PadLeft(const S: string; tolength: Integer; withchar: Char): string;

PadRight

Pads a string with characters up to a fixed length.

Declaration: function PadRight(const S: string; tolength: Integer; withchar: Char): string;

Pos

Returns the index value of the first character in a specified substring that occurs in a given string.

Declaration: function Pos(Substr: String; S: String): Integer;

Trim

Returns a copy of the string S with leading spaces and control characters removed.

Declaration: function Trim(const S: String): String;

TrimLeft

Returns a copy of the string S with leading spaces and control characters removed.

Declaration: function TrimLeft(const S: String): String;

TrimRight

Returns a copy of the string S with trailing spaces and control characters removed.

Declaration: function TrimRight(const S: String): String;

UpperCase

Returns a copy of the string in uppercase.

Declaration: function UpperCase(const S: String): String;

Conversion Functions

BooleanToChar

Converts a boolean value to a string.

Declaration: function BooleanToChar(S: Boolean): String;

CharToBoolean

Converts a string to a boolean value.

Declaration: function BooleanToChar(S: String): Boolean;

Chr

Returns the character with the ordinal value (ASCII value) of the byte-type expression X.

Declaration: function Chr(X: Byte): Char;

CurrToStr

Formats a currency value as string.

Declaration: function CurrToStr(Value: Currency): String;

DateTimeToStr

Converts a Datetime value to string.

Declaration: function DateTimeToStr(aDateTime: DateTime) : String;

DateToStr

Converts a Date value to string.

Declaration: function DateToStr(aDate: Date): String;

FloatToStr

Converts a floating point value to string.

Declaration: FloatToStr(Value: Extended): String;

IntToStr

Converts an integer to a string.

Declaration: function IntToStr(Value: Integer): String;

Ord

Returns the ASCII value of a character.

Declaration: function Ord(X: Char): Integer;

RGB

Returns a red, green, blue (RGB) color based on the arguments supplied. The intensity for each argument is in the range 0 through 255. If all three intensities are zero, the result is black. If all three intensities are 255, the result is white.

Declaration: function RGB(bRed, bGreen, bBlue: Integer): Integer;

StrToCurr

Converts a string to a currency value.

Declaration: function StrToCurr(const S: String): Currency;

StrToDate

Converts a string to a date value.

Declaration: function StrToDate(const S: String): Date;

StrToDateTime

Converts a string to Datetime value.

Declaration: function StrToDateTime(const S: String): DateTime;

StrToFloat

Converts a given string to a floating point value.

Declaration: StrToFloa (const S: String): Extended;

StrToInt

Converts a string that represents an integer (decimal or hex notation) to a number.

Declaration: function StrToInt(const S: String): Integer;

StrToIntDef

Converts a string that represents an integer (decimal or hex notation) to a number. If S does not represent a valid number, StrToIntDef returns the number passed in Default.

Declaration: function StrToIntDef(const S: String; Default: Integer) : Integer;

StrToTime

Converts a string to time value.

Declaration: function StrToTime(const S: String): Time;

TimeToStr

Returns a string that represents a DateTime value.

Declaration: TimeToStr(aTime: Time): String;

Format Functions

FormatCurr

Formats a currency object. See the FormatFloat function for more information on supported formats.

Declaration: function FormatCurr(const Format: String; Value: Currency): String;

FormatDateTime

Formats a datetime value.

Declaration: function FormatDateTime(const Format: string; aDateTime: datetime): String;

The following format specifiers are supported:

Specifier	Description
C	Displays the date using the format given by the Windows short date format, followed by the time using the format given by the Windows long time format. The time is not displayed if the fractional part of the DateTime value is zero
d	Displays the day as a number without leading zero (1-31)
dd	Displays the day as a number with a leading zero (01-31)

ddd	Displays the day as an abbreviation (Sun-Sat)
dddd	Displays the day as a full name (Sunday-Saturday)
ddddd	Displays the date using the format given by the Windows short date format.
dddddd	Displays the date using the format given by the Windows long date format.
m	Displays the month as a number without leading zero (1-12). If the m specifier immediatel follows an h or hh specifier, the minute rather than month is displayed.
mm	Displays the month as a number with a leading zero (01-12). If the mm specifier immediatel follows an h or hh specifier, the minute rather than month is displayed.
mmm	Displays the month as an abbreviation (Jan-Dec)
mmmm	Displays the month as a full name (January-December)
yy	Displays the year as a two digit number (00-99)
yyyy	Displays the year as a four digit number (0000-9999)
h	Displays the hour without leading zero (0-23)
hh	Displays the hour with a leading zero (00-23)
n	Displays the minute without leading zero (0-59)
nn	Displays the minute with a leading zero (00-59)
s	Displays the second without leading zero (0-59)
ss	Displays the second with a leading zero (00-59)
z	Displays the millisecond without leading zero (0-999)
zzz	Displays the millisecond with a leading zero (000-999)
t	Displays the time using the format given by the Windows short time format
tt	Displays the time using the format given by the Windows long time format
am/pm	Uses the 12 hour clock for preceding h or hh specifier, and displays 'am' for any hours before noon and 'pm' for any hour after noon. The am/pm specifier can use lower, upper, or mixed case and the result is displayed accordingly.
a/p	Uses the 12 hour clock for preceding h or hh specifier, and displays 'a' for any hours before noon and 'p' for any hour after noon. The a/p specifier can use lower, upper, or mixed case and the result is displayed accordingly.
ampm	Uses the 12 hour clock for preceding h or hh specifier, and displays the windows AM symbol for any hour before noon and contents of Windows PM Symbol for any hour after noon.
/	Displays the date separator character given by the Windows date separator.
:	Displays the time separator character given by the Windows time separator.

The following example assigns 'The meeting is on Wednesday, February 15, 1995 at 10:30 AM' to the string variable S.

```
S:= FormatDateTime('"The meeting is on " dddd, mmmm d , yyyy, " at "
hh:mmAM/PM', StrToDateTime('2/15/95 10:30am'));
```

FormatFloat

Formats a floating point value.

Declaration: function FormatFloat (const Format: String; Value: Extended): String;

The following format specifiers are supported:

Specifier	Description
0	Digit placeholder. If the value being formatted has a digit in the position where the '0' appears in the format string, then that digit is copied to the output string. Otherwise, a '0' is stored in that position in the output string.
#	Digit placeholder. If the value being formatted has a digit in the position where the '#' appears in the format string, then that digit is copied to the output string. Otherwise, nothing is stored in that position in the output string.
.	Decimal point. The first '.' Character in the format string determines the location of the decimal separator in the formatted value; any additional '.' Characters are ignored. The actual character used as a decimal separator in the output string is determined by the Windows decimal separator specified in the Number format of the International section in the Windows control panel.
,	Thousand separator. If the format string contains one or more ',' characters, the output will that thousand separators inserted betweeneach group of three degits to the left of the decimal point. The placement and number of ',' characters in the format string does not affect the output, except to indicate that thousand separators are wanted. The actual character used as thousand separator in the output is determined by the Windows thousand separator specified in the Number format of the International section in the Windows control panel.
E+	Scientific notation. If any of the strings 'E+', 'E-', 'e+', 'e-' are contained in the format string, the number is formatted using scientific notation. A group of upto four '0' characters can immediately follow the 'E+','E-','e+', or 'e-' to determine the minimum number of digits in the exponent. The 'E+' and 'e+' formats cause a plus sign to be output for positive exponents and a minus sign to be output for negative exponents. The 'E-' and 'e-' formats output a sign character only for negative exponents.
;	Separates sections for positive, negative and zero numbers in the format string.

The locations of the leftmost '0' before the decimal point in the format string and the rightmost '0' after the decimal point in the format string determine the range of digits that are always present in the output string.

The number being formatted is always rounded to as many decimal places as there are digit placeholders ('0' or '#') to the right of the decimal point. If the format string contains no decimal point, the value being formatted is rounded to the nearest whole number.

To allow different formats for positive, negative and zero values, the format string can contain between one and three sections separated by semicolon:

- One section: The format string applies to all values.
- Two sections: The first section applies to positive values and zeros and the second section applies to negative values.
- Three sections: The first section applies to positive values, the second applies to negative values and the third applies to zeros.

The following table shows some sample formats and results produced when formats are applied to different values:

Format String	Value			
	1234	**-1234**	**0.5**	**0**
0	1234	-1234	1	0
0.00	1234.00	-1234.00	0.50	0.00
#.##	1234	-1234	.5	
#,##0.00	1,234.00	-1,234.00	0.50	0.00
#,##0.00;(#,##0.00)	1,234.00	(1,234.00)	0.50	0.00
#,##0.00;;Zero	1,234.00	-1,234.00	0.50	Zero
0.000E+00	1.234E+03	-1.234E+03	5.000E-01	0.000E+00
#,###E-0	1.234E3	-1.234E3	5E-1	0E0

DateTime Functions

CurrentDate

Returns the current date.

Declaration: function CurrentDate: Date;

CurrentDateTime

Returns the current date and time.

Declaration: function CurrentDateTime: DateTime;

CurrentTime

Returns the current time.

Declaration: fnction CurrentTime: DateTime;

DayOfWeek

Returns the day of week of the specified date as an integer between 1 and 7, where Sunday is the first day of the week and Saturday is the seventh.

Declaration: function DayOfWeek (aDate: DateTime): Integer;

DaysBetween

Returns the number of days between two dates.

Declaration: function DaysBetween(ANow, AThen:TDateTime): Integer;

DecodeDate

Breaks the value specified as the Date parameter into year, month and day values. If the given time value is less than or equal to zero, the year, month and day return parameters are all set to zero.

Declaration: procedure DecodeDate(aDate: DateTime; var Year, Month, Day: Integer);

DecodeTime

Breaks a DateTime value into hours, minutes, seconds and milliseconds.

Declaration: procedure DecodeTime(aTime: DateTime; var Hour, Min, Sec, Msec: Integer);

EncodeDate

Returns a datetime value from the values specified as the Year, month and day parameters. The year must be between 1 and 9999. Valid month values are 1 throught 12. Valid day values are 1 through 28,29,30 or 31, depending on month value. For example, the possible day values for month 2(February) are 1 through 28 or 1 through 29, depending on whether or not the year is a leap year.

Declaration: function EncodeDate(Year, Month, Day: Integer): DateTime;

EncodeTime

Encodes the given hour, minute, second and millisecond into a DateTime value. Valid hour values are 0 through 23. Valid Min and Sec values are 0 through 59. Valid MSec values are 0 through 999.

Declaration: function EncodeDate(Hour, Min, Sec, Msec: Integer): DateTime;

IncMonth

Returns a TDateTime value that is a certain number of months greater than the supplied start date.

Declaration: function IncMonth(aDate: TDateTime; aNumberOfMonths: Integer): TDateTime;

MonthsBetween

Returns the number of months between two dates.

Declaration: function MonthsBetween(ANow, AThen:TDateTime): Integer;

WeeksBetween

Returns the number of weeks between two dates.

Declaration: function WeeksBetween(ANow, AThen:TDateTime): Integer;

YearsBetween

Returns the number of years between two dates.

Declaration: function YearsBetween(ANow, AThen:TDateTime): Integer;

Utility Functions

MessageBeep

Plays the PC's default notification sound.

Declaration: procedure MessageBeep

OutputDebugString

Writes a string to the debug output.

Declaration: procedure OutputDebugString(const Message: string);

RecordCount

Returns the number of records in the specified data source.

Declaration: function RecordCount(const PipeLineName: string): Integer;

Note: This function must be called after data sources have been opened; otherwise it will return 0.

ShowMessage

Displays a message box with an **OK** button. The Msg parameter is the message string that appears in the message box.

Declaration: procedure ShowMessage(const Msg: string);

ShowMessage() can be useful for debugging your code. You can put messages in to find out what value variables have, when code is being run and when it's not, and many other things. Just remember that ShowMessage needs a string, so you can't pass it a number without converting it first:

```
ShowMessage('The value of x is: ' + IntToStr(x));
```

Exo Business Functions

AgeToDateRange

Returns the start and end dates of the specified period age. The age is specified as it appears in Exo Business, e.g. 0 is the current period, 1 is the previous period, 2 is two periods ago, etc. The ledger is specified by a single letter, which must be one of:

- D = Debtor
- C = Creditor
- G = General Ledger
- S = Stock

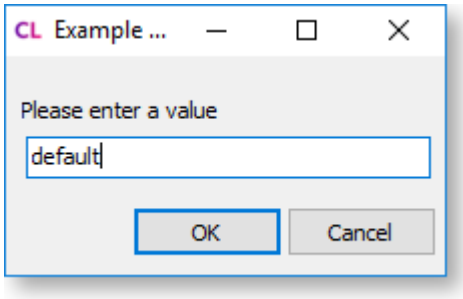
Declaration: procedure AgeToDateRange(Age: Integer; Ledger: string; var StartDate, EndDate: TDateTime);

AskforValue

Opens a small window prompting a user to enter a value, then returns the value that the user entered. You can specify the heading that should appear in the window's title bar, the prompt that appears on the window, a default value (which is returned if the user clicks **Cancel**), and whether or not the user must enter an integer value. For example, the following code:

```
ShowMessage('The value is ' + AskforValue('Example of AskForValue',
'Please enter a value', 'default', false));
```

will result in this window appearing to the user:



The value that the user enters is then displayed using the standard ShowMessage function.

Declaration: function AskforValue(DialogHeading: String; DialogPrompt: String; DefaultValue: String; RequireAnIntegerValue: Boolean): String;

Note: The function always returns a string, even if the user was required to enter an integer.

ComputerIDFields

Returns ID information for the computer with the specified SEQNO. The output string of the function is CLIENTNAME\COMPUTERNAME.

Declaration: function ComputerIDFields(aComputerSeqno: Integer; var ComputerName: String; var ComputerID: String; var ClientName: String; var EFTCAID: String): String;

Execute SQL

Executes any SQL Server command from within Clarity, e.g.

```
ExecuteSQL('update stock_items set status = "L" where stockcode = ' +
'"LABOUR"');
```

This function only available in the Report Designer if the User-level profile setting **Allow Clarity ExecSQL Function** is set to "Design and Runtime".

Declaration: procedure ExecuteSQL(ASQL: String);

FileExists

Returns true if the specified file exists. This can be used to determine if a company logo image exists and if so display it, as in the following code:

```
if FileExists(GlobalProfileReadString('IMAGESDIR') + '\' +
GlobalProfileReadString('FORM_LOGO')) then
 begin
   FormLogo.Picture.LoadFromFile(GlobalProfileReadString('IMAGESDIR') +
'\' + GlobalProfileReadString('FORM_LOGO'));
 end;
```

Declaration: function FileExists(pathname: String): Boolean;

FormatExoProtocolLink

Returns an Exo protocol link for a specified Command and Value, e.g.

```
Link := FormatExoProtocolLink('draccount', 10);
```

This will return `exo://<current alias>/draccount(10)` into the link.

Declaration: function FormatExoProtocolLink(Command, Value: String): String;

> **Note:** This function uses the Exo protocol handler to enforce security constraints, i.e. as with all Exo protocol links, users cannot use them to access part of the Exo Business system that they do not have access rights for.

GetChartSeries

Returns the series object of a chart that corresponds with the supplied series index number. This is used in some dashboard reports to open Exo Business records when a chart is clicked on, using Exo Business procedures like ShowDebtor, ShowCreditor or ShowStockItem (see below) to open the record.

See the **DashTopNCustomersbyYTDMargin.CLR** or **DashTopNStockbyYTDSalesValue.CLR** reports for examples.

Declaration: function GetChartSeries(TeeChart: TppTeeChart; const SeriesIndex: Integer): TComponent;

GetCheckDigit

Returns a check digit (used with bank accounts) for the specified invoice.

Declaration: function GetCheckDigit(InvoiceRef: String): String;

GetTaxValue

Returns the tax value for a line, using the following inputs:

- ATotal - the line total amount
- Ledger - the ledger, which should be 'Dr' or 'Cr'
- StockCode – the stock code of the stock item on the line
- ATaxStatus – the ID of the tax rate to use, or enter -1 to let Exo Business determine the tax code to use

Declaration: function GetTaxValue(ATotal: Double; ALedger, AStockCode: String, ATaxStatus: Integer): Double;

GolbalProfileReadBoolean

Returns the value of the specified profile setting as a boolean value. Profile settings are identified by their profile name, not their descriptive name, e.g. 'ALLOWAGEDBALEDITING', not 'Allow editing of debtor/creditor aged balances'.

Declaration: function GlobalProfileReadBoolean(aProfileField: String): Boolean;

GlobalProfileReadString

Returns the value of the specified profile setting as a string. Profile settings are identified by their profile name, not their descriptive name, e.g. 'FORM_LOGO', not 'Company logo filename'.

Declaration: function GlobalProfileReadString(aProfileField: String): String;

IsZeroValue

Returns true if the supplied value is zero.

Declaration: function IsZeroValue(ANumber: Double): Boolean;

LoadStockVal

Loads a pipeline used by stock valuation reports.

Declaration: procedure LoadStockVal(const aLocation: Integer);

LoadVariance

Loads a pipeline used by stock variance reports.

Declaration: procedure LoadVariance(const aLocation: Integer);

LoggedInStaffNo

Returns the STAFFNO of the Exo Business staff member who is currently logged in.

Declaration: function LoggedInStaffNo: Integer;

MarketingClassName

Returns the descriptive name of the marketing class associated with the supplied CLASSNO in the format "ID. Name", e.g. "1. Accounts".

Declaration: function MarketingClassName(aClassNo: Integer): String;

RefreshPage

Refreshes the data sources by re-querying the Exo Business database, redraws the resized page and refreshes any components that have not updated themselves.

Declaration: procedure RefreshPage(aReport: TppReport);

> **Note:** This function should not be used carelessly - it does a complete refresh of the entire report including all data source querying, as if you had closed the report and reopened it. If it is called too many times, your report will slow down significantly. Refreshing does not happen immediately, so it is not advisable to call this function to get the record count to calculate - use the **LookAhead option** instead.

SellPriceName

Returns the descriptive name of the sell price associated with the supplied PRICENO in the format "ID. Name", e.g. "1. Internet".

Declaration: function SellPriceName(aSellPriceNo: Integer): String;

SetChequeNo

This procedure is used to increment the cheque number when print cheque remittance reports. You can see it in use in **FinPPCheqWr.CLF**.

Declaration: procedure SetChequeNo(iChequeNo, iSeqNo: Integer);

ShellExecute

Executes an external DOS command.

Declaration: function ShellExecute(AFileName: String; AParams: String; AFolder: String; AWaitFor: Boolean): Integer;

ShowCreditor

Opens the Creditor Account Details window, showing the account with the specified ACCNO.

Declaration: procedure ShowCreditor(AccNo: Integer);

ShowDebtor

Opens the Debtor Account Details window, showing the account with the specified ACCNO.

Declaration: procedure ShowDebtor(AccNo: Integer);

ShowGLCode

Opens the General Ledger Account Details window, showing the specified GL account. The GLCode can be a single account number, e.g. 1000, or a formatted string, e.g. 1000-10 or 2-1000-10 with the branch.

Declaration: procedure ShowGLCode(GLCode: String, HasBranch: Boolean);

ShowJob

Opens the Job Details window, showing the account with the specified JOBNO.

Declaration: procedure ShowJob(JobNo: Integer);

ShowNonAccount

Opens the Non Account Details window, showing the account with the specified SEQNO.

Declaration: procedure ShowNonAccount(SeqNo: Integer);

ShowServiceUnit

Opens the Exo Serviceable Units window, showing the account with the specified SEQNO.

Declaration: procedure ShowServiceUnit(SUSeqNo: Integer);

ShowStockItem

Opens the Stock Item Details window, showing the stock item with the specified STOCKCODE.

Declaration: procedure ShowStockItem(StockItem: String);

TaxRateName

Returns the name of the tax rate with the supplied SEQNO. If ReturnLongName is true, the tax rate's NAME is returned; if it's false, the tax rate's SHORTNAME is returned.

Declaration: function TaxRateName(aTaxRateNo: Integer; ReturnLongName: Boolean): String;

TaxRateRate

Returns the rate of the tax rate with the supplied SEQNO.

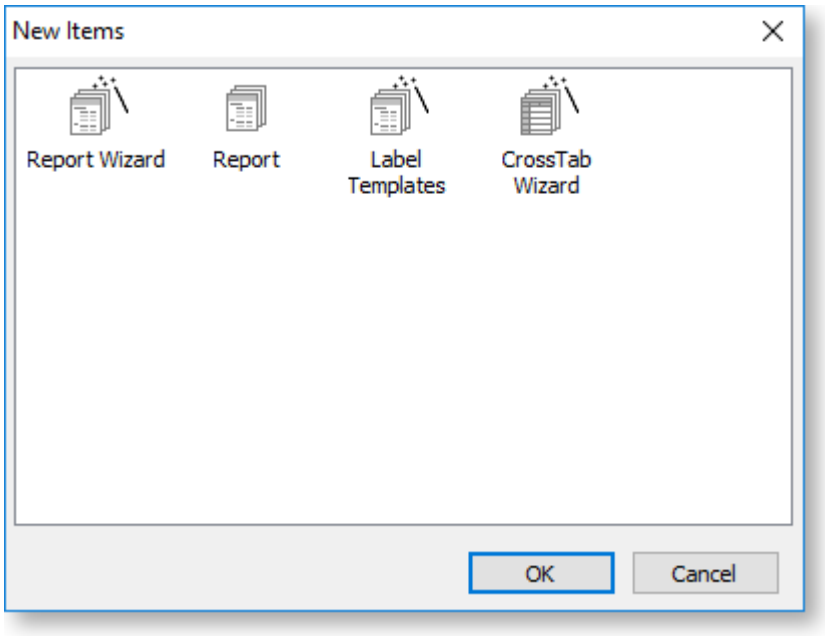
Declaration: function TaxRateName(aTaxRateNo: Integer): Double;

Creating Reports

The Report Wizard

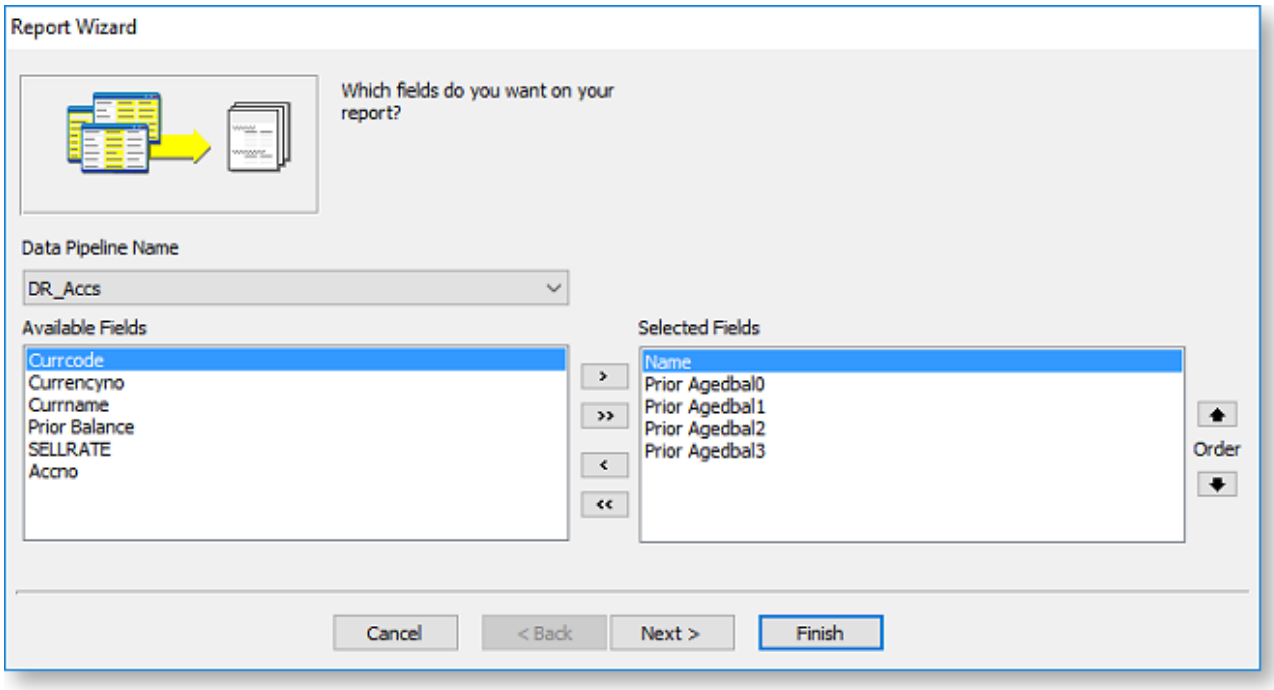
The Report Wizard takes you through the basic steps of setting up a simple report. Before running the wizard, you must have set up one or more data sources/pipelines on the Data Tab.

Start the Report Wizard by selecting **New** from the Design tab's File menu, then selecting **Report Wizard** from the New Items window.



Select Fields

The first screen of the wizard lets you choose which fields should appear on the report.



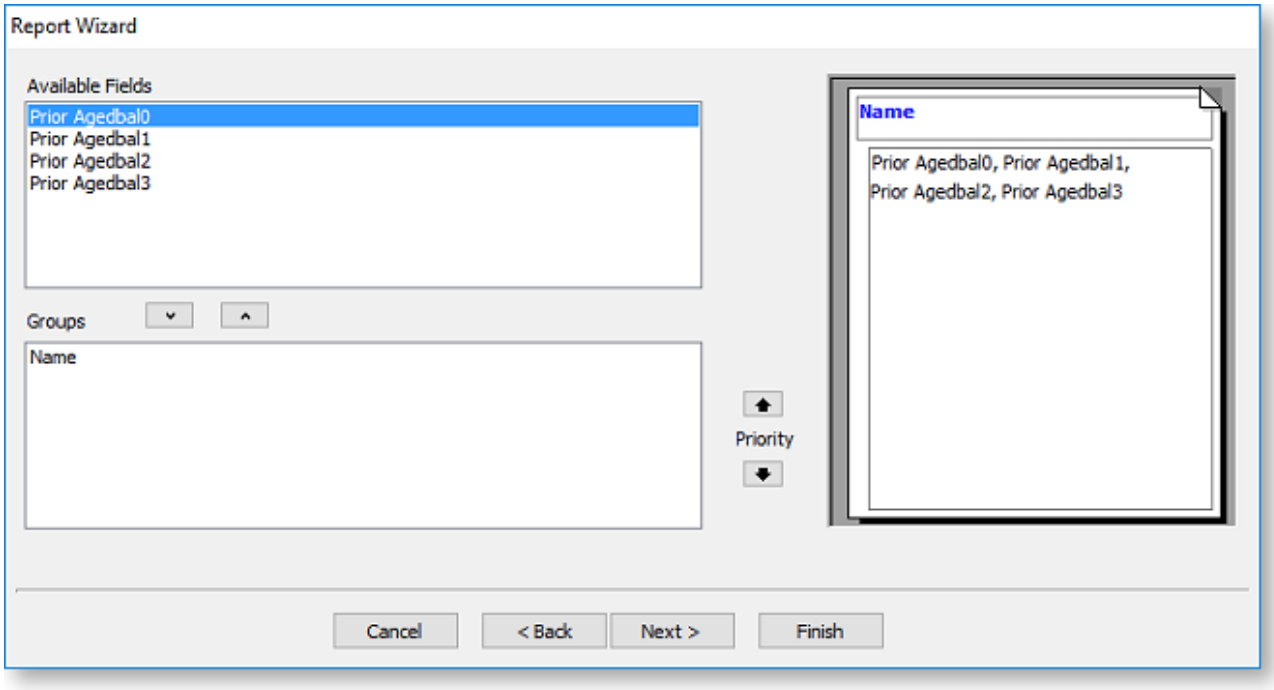
The **Data Pipeline Name** dropdown contains all data sources set up on the Data tab - select the data source you want to use for this report. All fields in the selected data source appear in the Available Fields section. Move fields between the Available Fields and Selected Fields sections using the following methods:

- Double-click on a field in the Available Fields section to move it to the Selected Fields section (and vice versa).
- Select one or more fields, then click the left and right arrow buttons to move them from one section to the other. You can select multiple fields using the CTRL and SHIFT keys.
- Click the double arrow buttons to move all fields from one section to the other.

Once you've added all of the fields you want to include in the report, you can order them using the **Order** arrow buttons next to the Selected Fields section. Click **Next** to go to the next wizard screen.

Group Fields

The next wizard screen lets you group the report by one or more of the fields you selected on the previous screen.



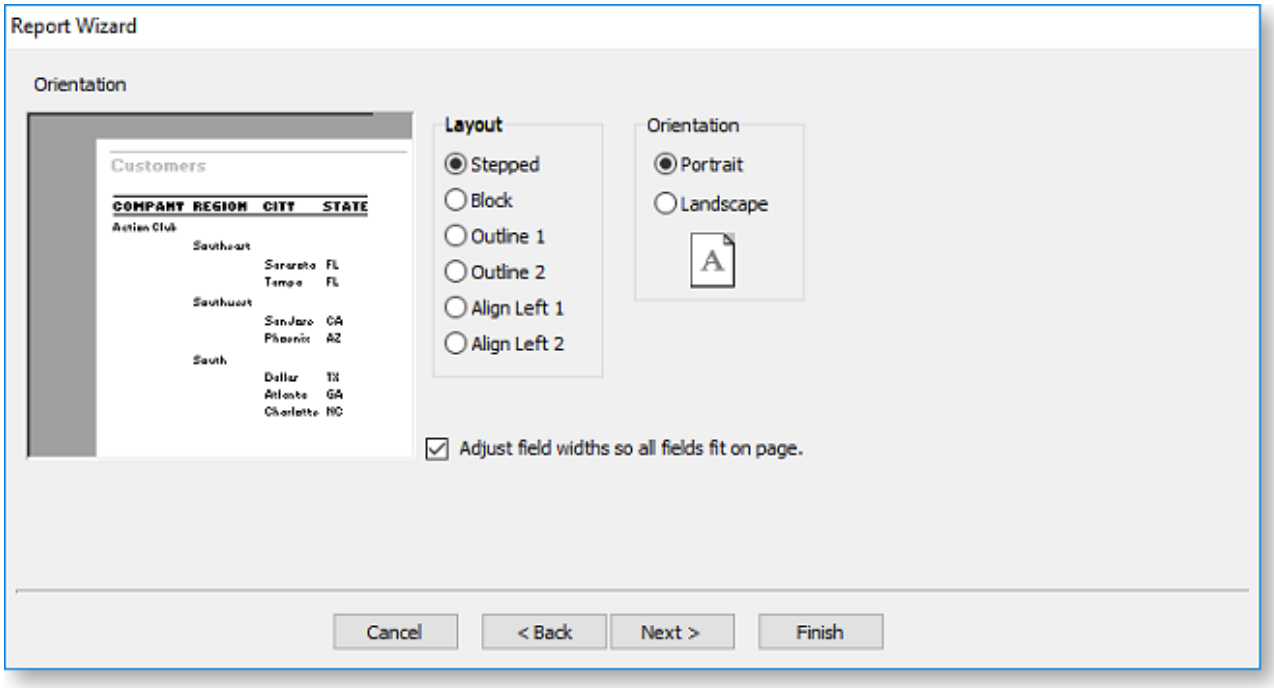
If you want to group the report by a field, double-click on it in the Available Fields pane, or select it and click the down arrow button. The report preview on the right of the window updates to show how the grouping will appear.

If you are grouping by more than one field, you can use the **Priority** arrow buttons to change the order in which the grouping will be applied.

Once you have set up grouping, click **Next** to go to the next wizard screen.

Set the Report's Layout

The next wizard screen contains options for how the report should be laid out.



Specify whether the report should be in Portrait or Landscape orientation, then select a layout format. The layout options differ depending on whether or not you set up grouping on the previous screen.

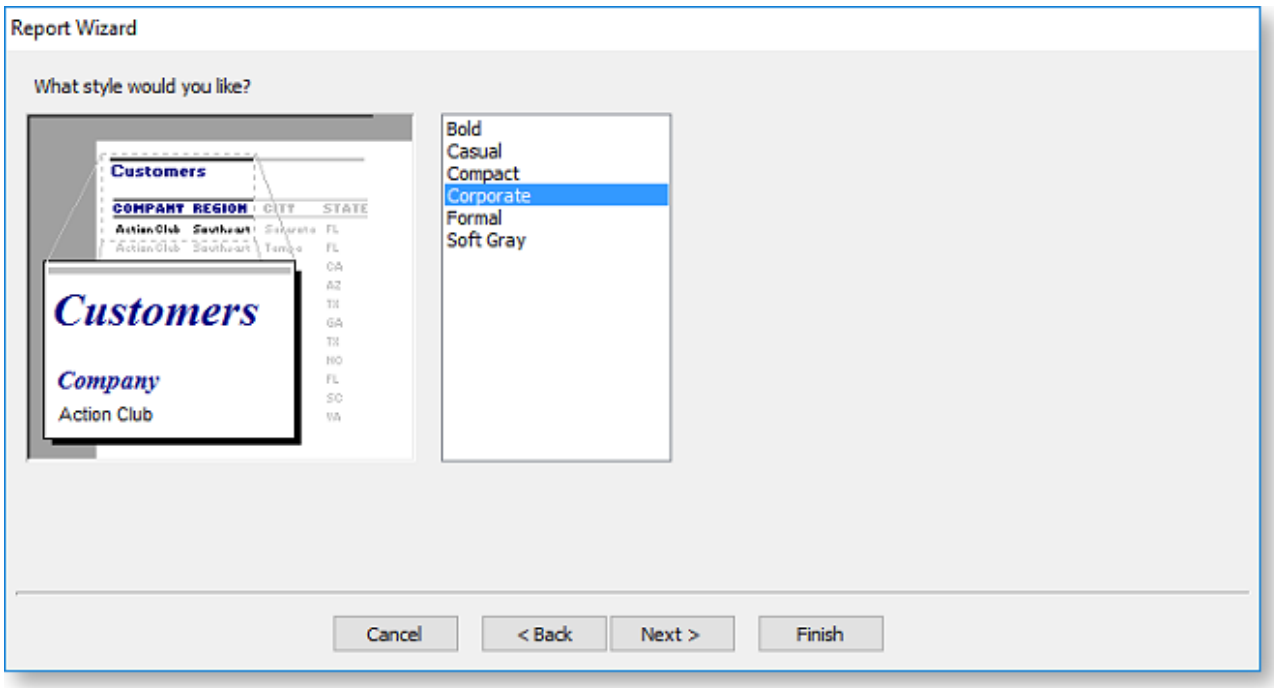
If the report is not grouped, you can choose between a **Vertical** and **Tabular** layout. In a vertical layout, report records are arranged vertically, with each record in its own section. In a tabular layout, report records are arranged horizontally in lines with a header line at the top of the report.

If the report is grouped by one ore more fields, a variety of options are available for how the groups should be laid out - the preview at the left of the screen updates as you select each option.

Once you have set up the report's layout, click **Next** to go to the next wizard screen.

Set the Report's Appearance

The next wizard screen lets you select a visual theme for the report.

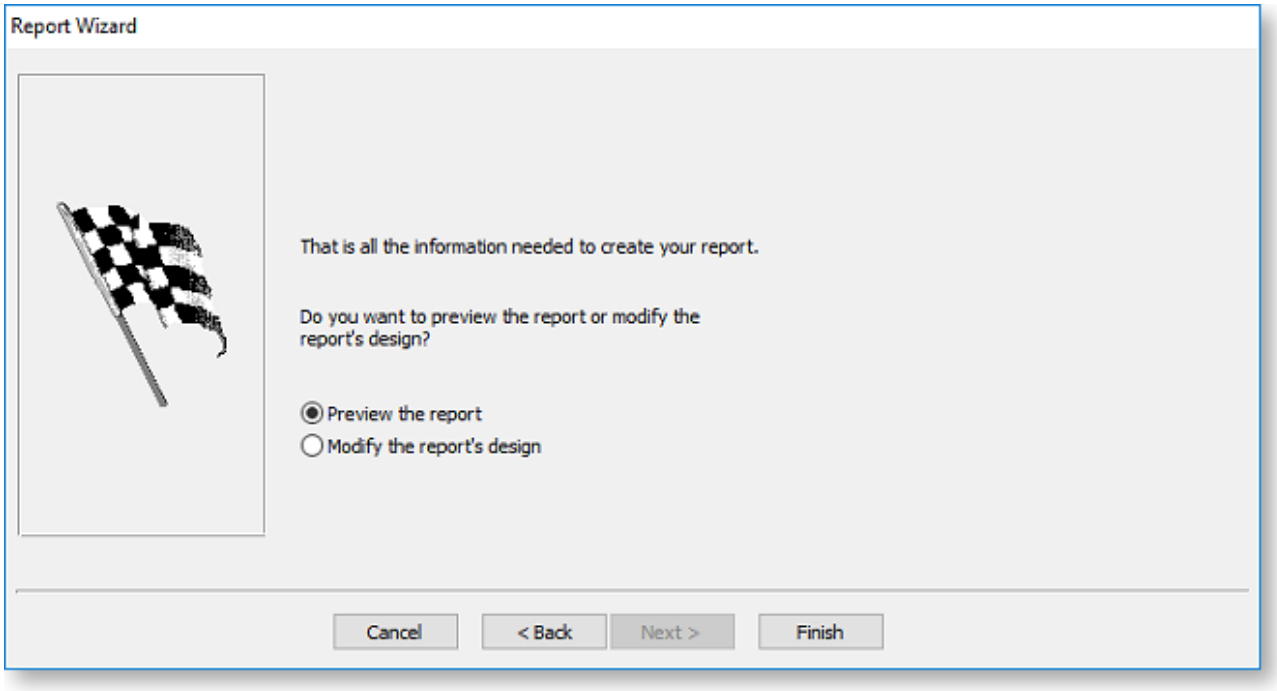


Select a theme from the available options. Each theme has a different combination of fonts and colours.

Click **Next** to go to the next wizard screen.

Finish the Wizard

Exo Clarity has all the details it needs to create the report.



Click **Finish** to create the new report and perform one of the following actions:

- Display a preview of the report on the Preview tab.
- Open the report on the Design tab where you can continue to edit it.

Grouping Data

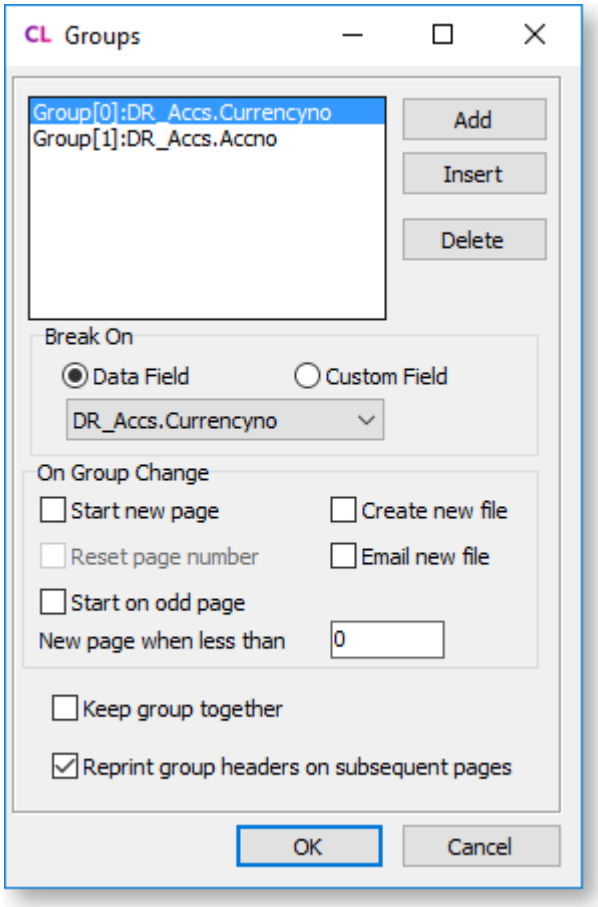
Grouping is added to a report so that summary information can be extracted from specific subsets of data. Grouping works for fields where a significant number of records have the same value, e.g. grouping transactions on ACCNO lets you report on all transactions for each account. A field like TRANSDATE would not work well for grouping, since each transaction is likely to have a unique date and time.

> **Note:** You should also sort the report by any field that you are grouping by, e.g. if you are grouping by ACCNO, sort by ACCNO in addition to any other fields you want to sort on. Sorting is set up on **the Data tab**.

Grouping can be useful for arranging data so that the report is easier to read, e.g. grouping transactions by Debtor instead of showing all transactions in one long list. It can also save you time when you need to send out common documents to many recipients, e.g. Debtor statements - instead of printing the same report many times, changing the recipient each time, you can create one report, group it by recipient and ensure that each group appears on a separate page, which can then be sent out individually. Settings are available to start each group on a new page and restart page numbering for each group - you can even direct the report to create an entirely separate output file for each group (see below).

> **Note:** This is different from the grouping that can be performed in the **Query Designer**, which is done when using calculated fields (set up on the Calcs tab) to aggregate data in the data source.

You can set up groups on the Design tab by selecting **Groups** from the Report menu. This opens the Groups window:



To add a new grouping, click **Add**. A new group record appears in the area at the top of the window. To set the field that the data will be grouped by, select a field from the dropdown in the **Break On** section. You can choose to group by a data field from the report's primary data source, or a custom field on the report.

The rest of the settings on the Groups window determine how the report should behave with respect to the selected group - you can have different settings for each group.

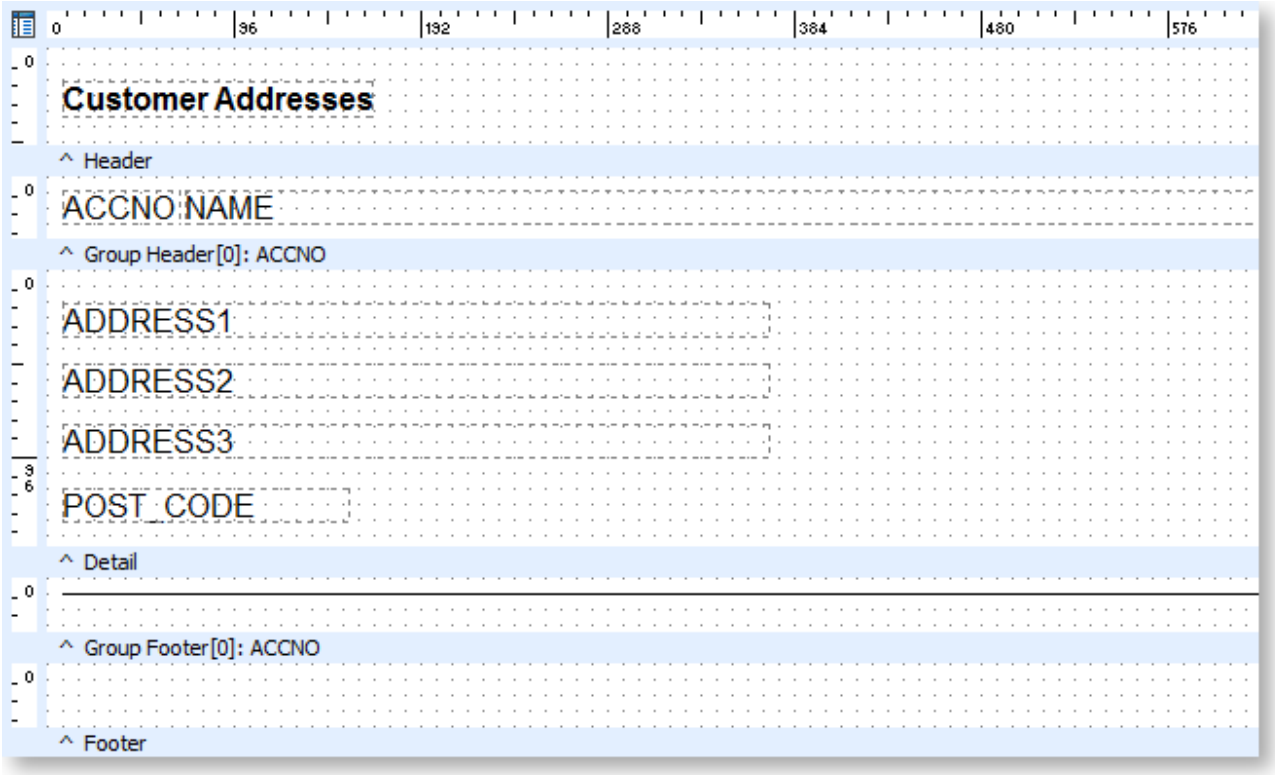
Name	Description
Start new page	If this option is ticked, each group will start on a new page.
Reset page number	If **Start new page** is ticked, this option determines whether or not to reset the page number to 1 at the start of each group.
Start on odd page	If this option is ticked, each group will start on a new page. This is useful for generating duplexed reports.
New page when less than	This option is available when **Start new page** is unticked. Set it to a value greater than zero to prevent "orphaned" headers, i.e. headers at the bottom of a page with all their detail lines on the next page. If there is not enough room for the number of detail lines you specify here, the group will be moved to a new page.
Create new file	If this option is ticked, a separate output file will be created for each group.
Email new file	If this option is ticked, a separate file will be created and emailed for each group. This allows you to send each group to a separate email recipient.

Note: To send emails to different addresses for each group, you will need to add code to the group's **OnGetEmailSettings event** to set up the Report.EmailSettings object each time a group is emailed.

Keep group together	This option controls how the component should behave it the group does not fit on one page. If it is disabled, the group is printed to the bottom of the available space, then continues on the next page. If it is enabled, the entire group will be moved to the next page.
Reprint group header on subsequent pages	If this option is disabled, the group's header will appear once in the report; if it is enabled, the group header will appear at the top of every page.

Groups on the Design Tab

When a report contains groups, extra report bands appear on the Design tab. A Group Header band and Group Footer band is added for each group:



Sub Reports

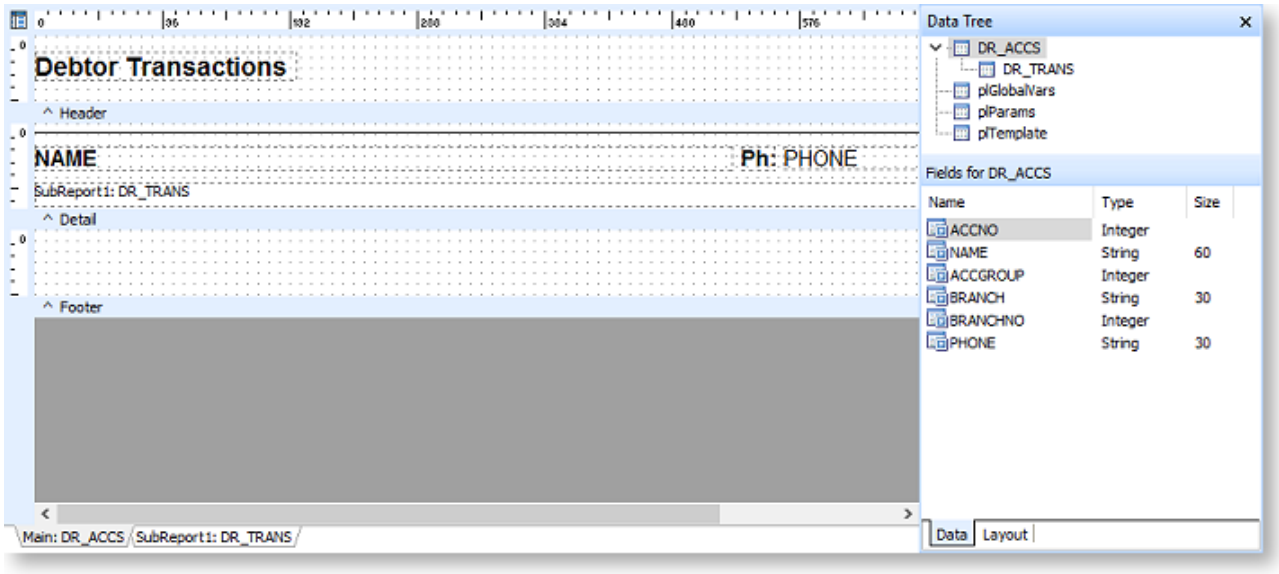
An Exo Clarity report can include one or more sub reports - these are separate reports that can use a different data source from the main report. This can be useful if:

- There is a master/detail relationship between the data sources, e.g. transactions for a Debtor account or serial numbers on a document line.
- The report requires two or more different data sources that do not link in any way.
- The report contains sets of data that require different search criteria.
- The report requires line-level and summary data.
- You want to be able to click on the report to drill down to detailed data.

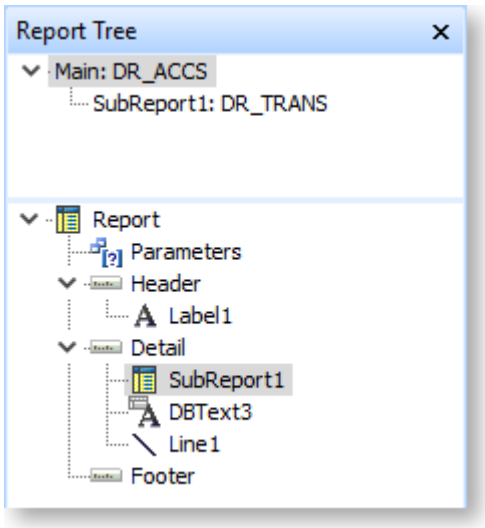
Note: In some cases, the effects of a sub report could also be achieved by **grouping data**.

Adding a Sub Report

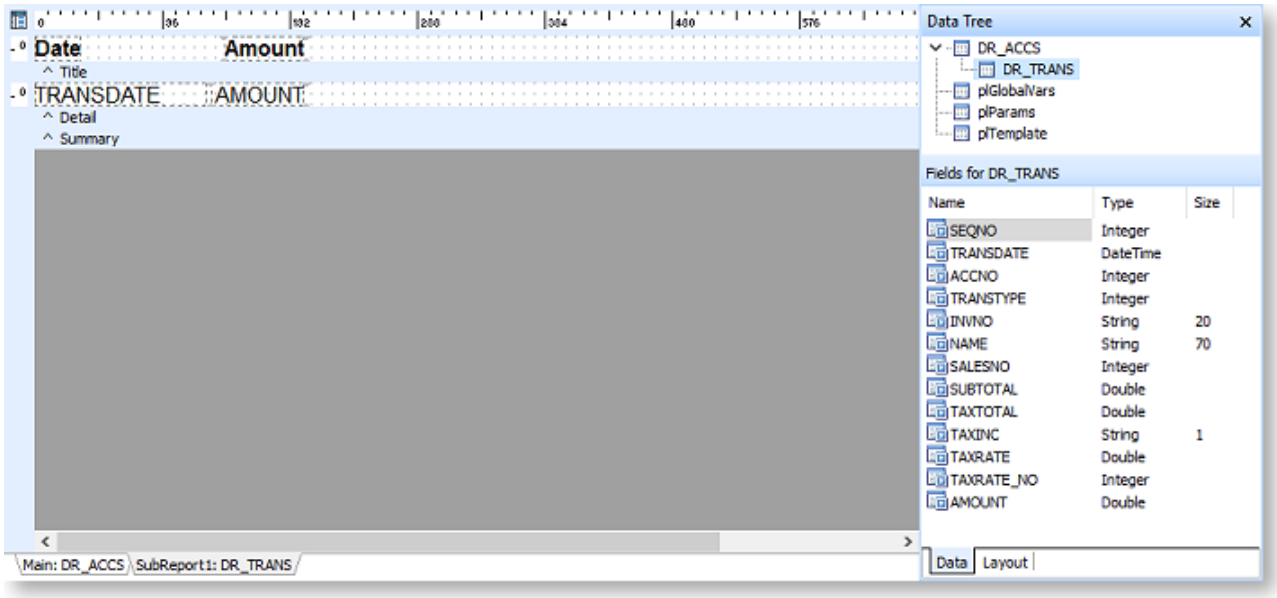
Sub reports are added on the Design tab. To add a sub report to a report, click the SubReport button ( ) on the Advanced Components toolbar then click on the report canvas. The SubReport component appears as a blank rectangle, which can be positioned and resized as necessary:



When one or more sub reports exist in a report, tabs become available at the bottom of the canvas and the bottom of the Calc tab, and the sub report(s) appear on the Report Tree:

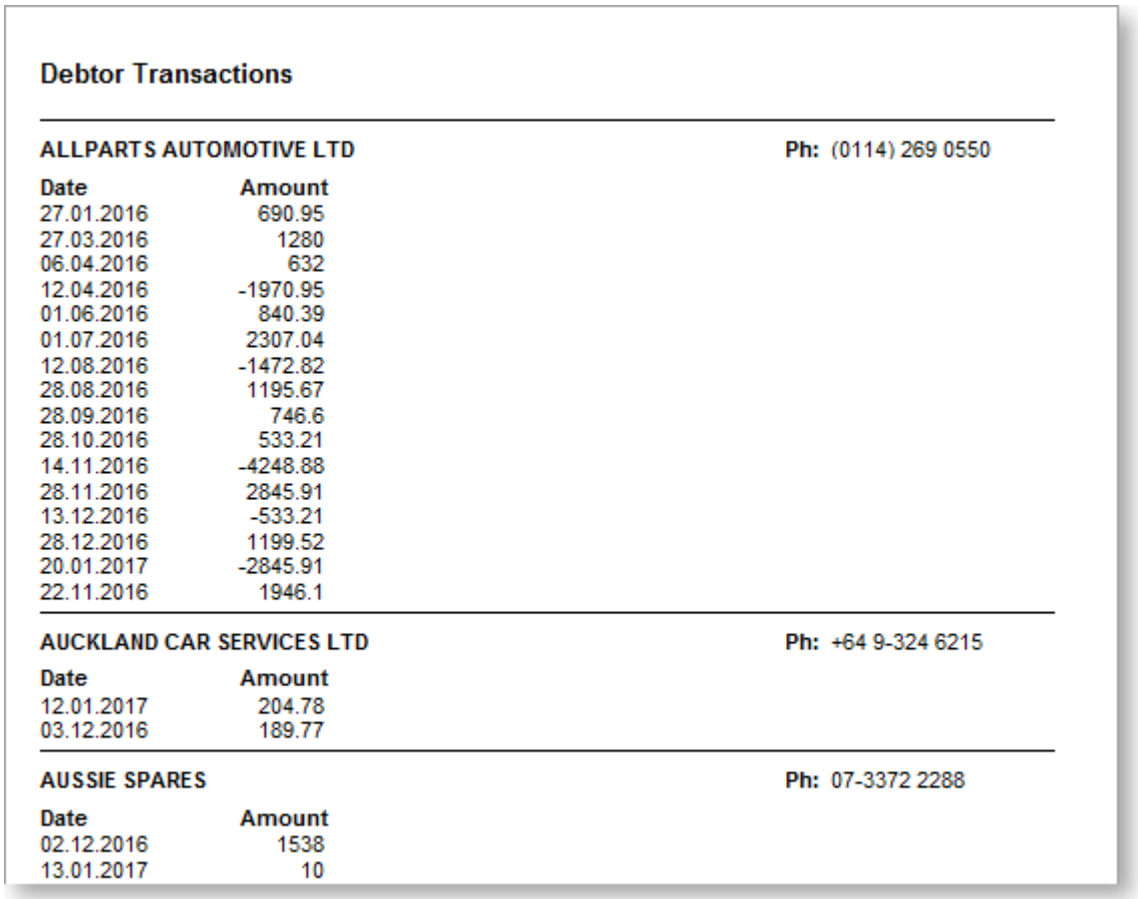


Clicking the sub report's tab opens a separate canvas for the sub report:



The sub report can have a different primary data source from the main report - select **Data** from the Report menu to choose a data source for the sub report.

The screenshots above show the setup for a simple report listing transactions for Debtor accounts - the main report uses DR_ACCS as its data source to show details about Debtors, while the sub report uses the DR_TRANS data source to show details of the transactions:



Debtor Transactions

ALLPARTS AUTOMOTIVE LTD **Ph:** (0114) 269 0550

Date	Amount
27.01.2016	690.95
27.03.2016	1280
06.04.2016	632
12.04.2016	-1970.95
01.06.2016	840.39
01.07.2016	2307.04
12.08.2016	-1472.82
28.08.2016	1195.67
28.09.2016	746.6
28.10.2016	533.21
14.11.2016	-4248.88
28.11.2016	2845.91
13.12.2016	-533.21
28.12.2016	1199.52
20.01.2017	-2845.91
22.11.2016	1946.1

AUCKLAND CAR SERVICES LTD **Ph:** +64 9-324 6215

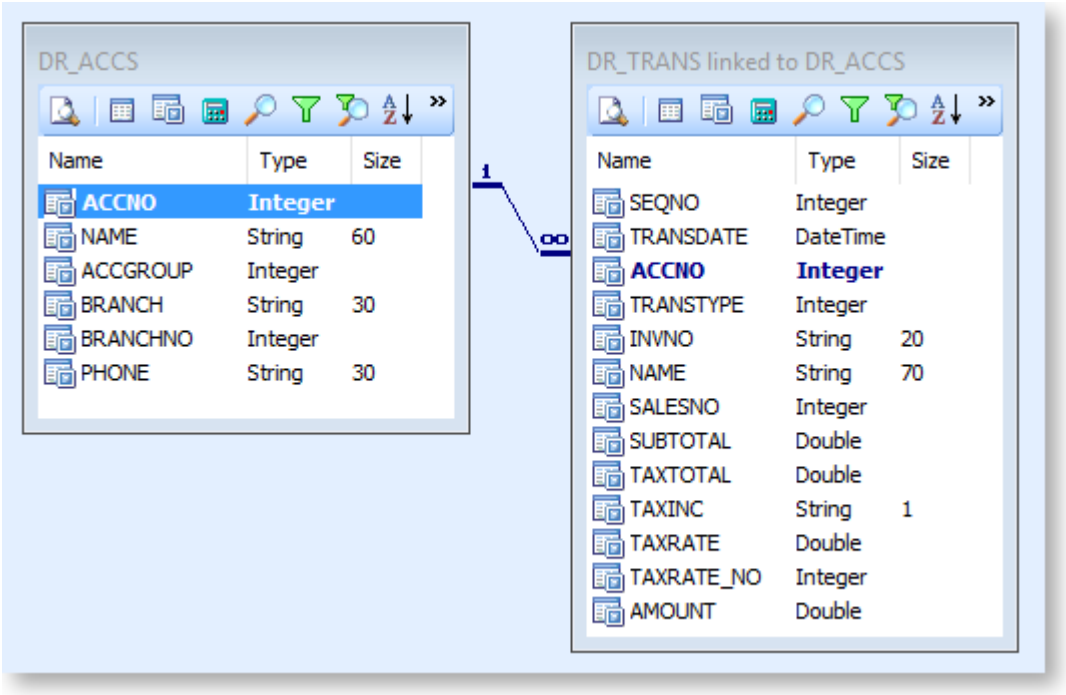
Date	Amount
12.01.2017	204.78
03.12.2016	189.77

AUSSIE SPARES **Ph:** 07-3372 2288

Date	Amount
02.12.2016	1538
13.01.2017	10

Linked and Non-linked Data Sources

In the example above, the sub report shows details of Debtor transactions. To ensure that the sub report shows only the transactions that relate to the Debtor that they appear under, there needs to be a link between the reports' data sources. In this example, the ACCNO field of the DR_TRANS data source is linked to the ACCNO field of the DR_ACCS data source on the Data tab:



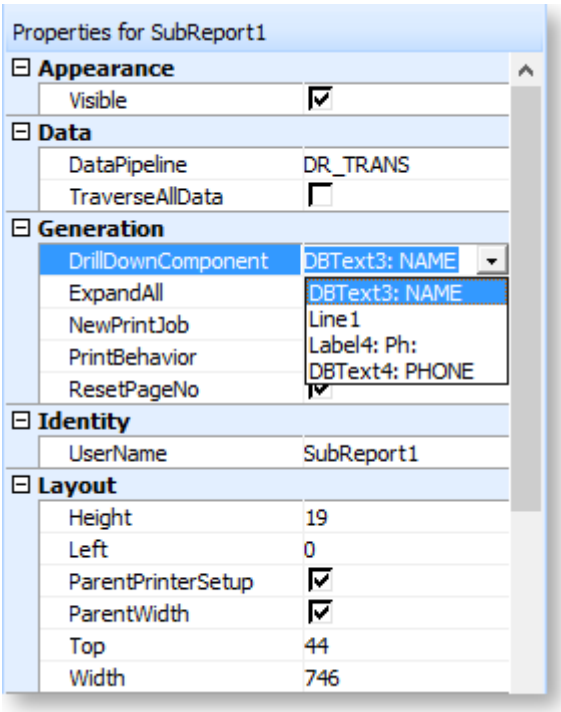
If this link was not set up, the sub report would not be able to relate the transactions to the accounts, and would show all Debtor transactions under every Debtor.

Not all sub reports require linked data, however. For example, some tax reports need to draw on information from both the Debtors and the Creditors tables. There is no way to join these tables together, so you could instead construct two separate sub reports and pass the results of their calculations to a third report which summarises the information.

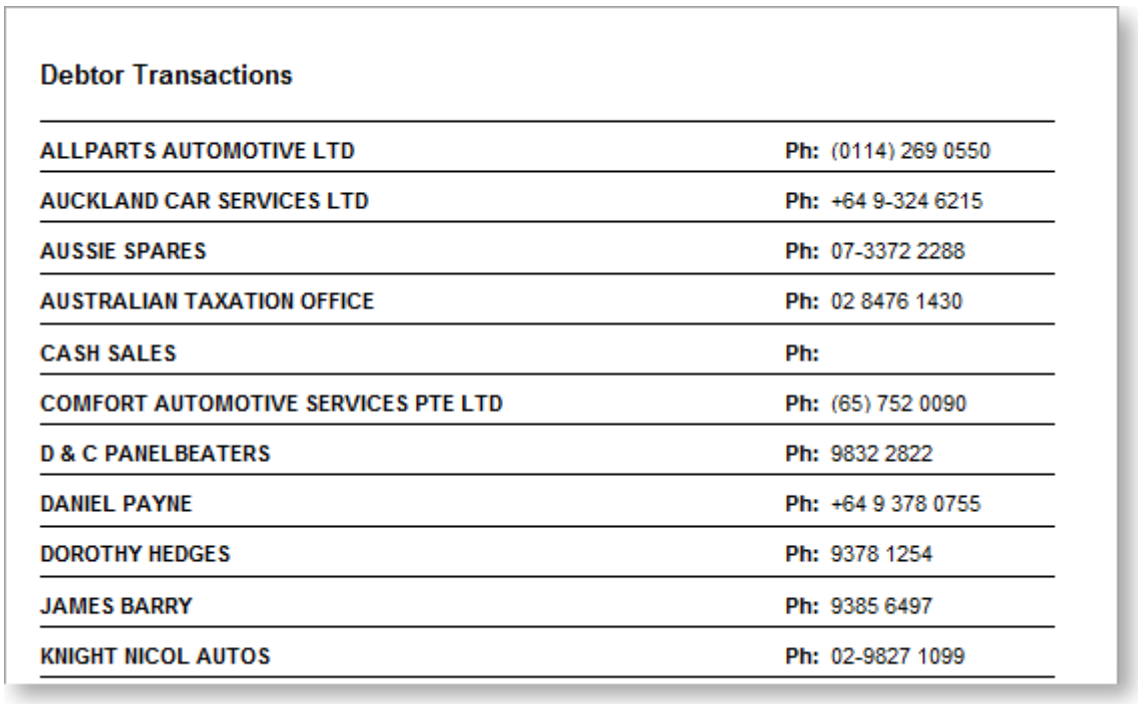
Drill-down Reports

Exo Clarity can build reports that display summary information to begin with, but give the ability to "drill down" to more detailed information if required. In these cases, the detailed information is placed in a sub report, which is hidden until a component of the main report is clicked on.

To make a sub report into a drill down report, click on the SubReport component on the main report and select a value for its **DrillDownComponent** property on the Report Tree:



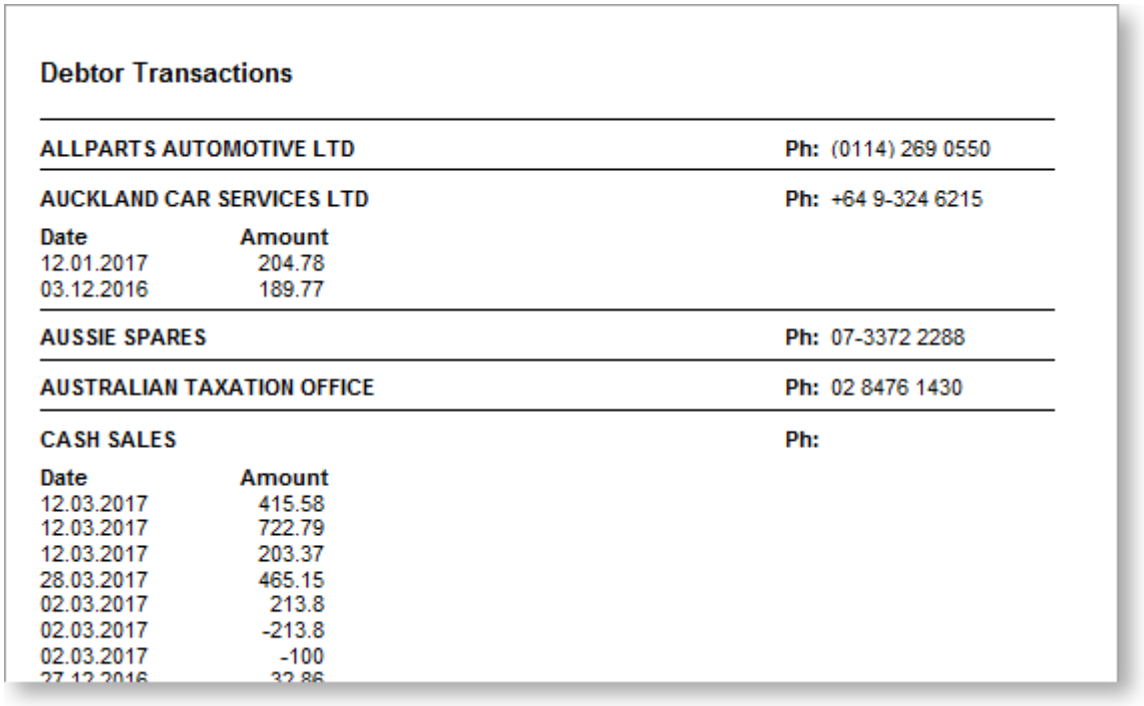
If **DrillDownComponent** is blank, the sub report is not a drill down report and will always be visible. To activate the drilldown functionality, you can select any of the other components on the main report from the dropdown here. The sub report will now be hidden and will only appear when the component you selected is clicked on. In the example above, the sub report showing transaction details has been made a drill down report, which is activated by clicking on the DBText component showing the Debtor's name ("DBText3: NAME"). When the report is previewed, only the Debtor details from the main report are shown to begin with:



Debtor Transactions

ALLPARTS AUTOMOTIVE LTD	**Ph:** (0114) 269 0550
AUCKLAND CAR SERVICES LTD	**Ph:** +64 9-324 6215
AUSSIE SPARES	**Ph:** 07-3372 2288
AUSTRALIAN TAXATION OFFICE	**Ph:** 02 8476 1430
CASH SALES	**Ph:**
COMFORT AUTOMOTIVE SERVICES PTE LTD	**Ph:** (65) 752 0090
D & C PANELBEATERS	**Ph:** 9832 2822
DANIEL PAYNE	**Ph:** +64 9 378 0755
DOROTHY HEDGES	**Ph:** 9378 1254
JAMES BARRY	**Ph:** 9385 6497
KNIGHT NICOL AUTOS	**Ph:** 02-9827 1099

Clicking on any of the Debtor names opens the sub report showing the transactions for that Debtor:



Debtor Transactions

ALLPARTS AUTOMOTIVE LTD — Ph: (0114) 269 0550

AUCKLAND CAR SERVICES LTD — Ph: +64 9-324 6215

Date	Amount
12.01.2017	204.78
03.12.2016	189.77

AUSSIE SPARES — Ph: 07-3372 2288

AUSTRALIAN TAXATION OFFICE — Ph: 02 8476 1430

CASH SALES — Ph:

Date	Amount
12.03.2017	415.58
12.03.2017	722.79
12.03.2017	203.37
28.03.2017	465.15
02.03.2017	213.8
02.03.2017	-213.8
02.03.2017	-100
27.12.2016	32.86

Tables

You can organise a report into a table layout by adding a TableGrid component, then placing other components onto the table. As the table is moved or resized, all components on the table will move and resize with it.

Adding Tables

To add a table, click the TableGrid button ( ) on the Advanced Components toolbar, then click on the report canvas in the place where you want the table to appear.

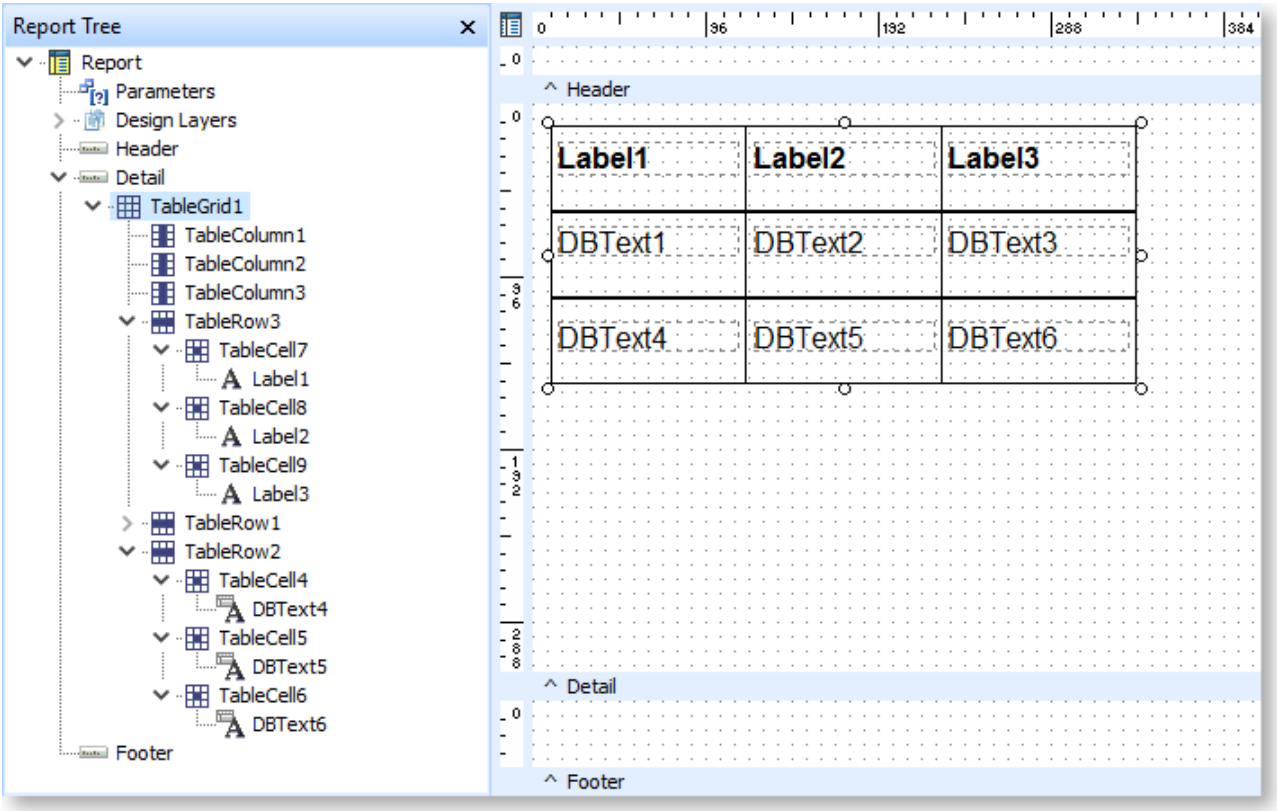
By default, a table contains a single row with three table cells in it. You can set up the table's layout using its right-click options:

Option	Description
Row	Submenu containing the following options: • Select - selects all cells in the row that you right-clicked on. • Add - adds a new row to the bottom of the table. • Insert - adds a new row above the row that you right-clicked on. • Delete - deletes the selected row. You cannot delete a row if it is the only one left in the table.
Column	Submenu containing the following options: • Select - selects all cells in the column that you right-clicked on. • Add - adds a new column to the right side of the table. • Insert - adds a new column to the left of the column that you right-clicked on. • Delete - deletes the selected column. You cannot delete a column if it is the only one left in the table.
Cell	Submenu containing the following options:

- Merge - merges the selected cells into a single cell. (Only available when multiple cells are selected.)
- Divide - splits the selected merged cell into its original cells.

Editing Tables

Once you have added the table, you can add components to it by clicking on a component's toolbar button, then clicking on the table cell you want to place it in. The Report Tree shows all cells of the table in a tree structure, including the components that are inside each cell:



You can click on the parts of the table in the Report Tree to select that part on the report canvas, e.g. in the screenshot above, clicking on TableGrid1 in the report tree will select the entire table, while clicking on TableRow3 will select all cells in that row. You can hold down SHIFT while clicking on parts of the table (on the canvas or on the Report Tree) to select multiple cells at once.

Note: You can also right-click on any part of the table on the Report Tree to access the right-click options for that row, column or cell.

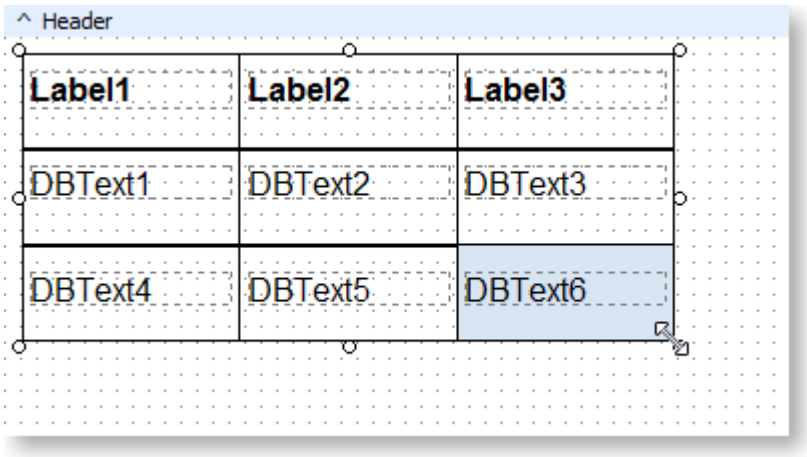
The controls on the Draw toolbar can be used to set the fill and line colours of the selected table cell(s).

You can move the table by clicking on any part of the table on the report canvas and dragging it into a new position. (Be sure to click on the table itself, not on a component that has been placed on a table cell - if you click on one of the child components, it will just move that component.)

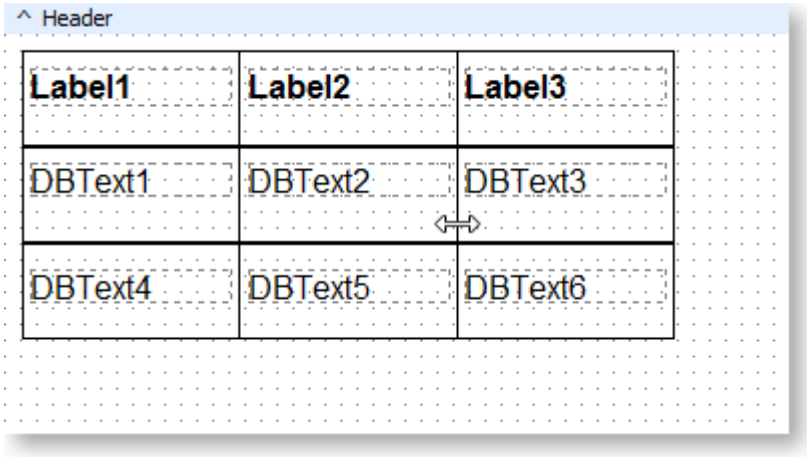
You can resize the table by clicking and dragging on the edges of the table. Resizing can be done in two ways:

- When any part of the table has been selected, circles appear at the edges of the table. Click and drag a circle to resize the entire table - all cells will be resized.
- When the table is not selected, you can hover the mouse over the borders of an individual cell, then click and drag to resize only that row or column.

In the screenshot below, the lower right cell of the table has been clicked on. The cell is highlighted, and circles appear at the edges. You can see that hovering the mouse over one of the circles changes the mouse pointer to show that you can click to resize the whole table:



In the following screenshot, no part of the table is selected. Moving the mouse over a cell's border lets you drag just that border to resize individual rows or columns. In this case, the pointer indicates that you can drag left or right to change the column's width:

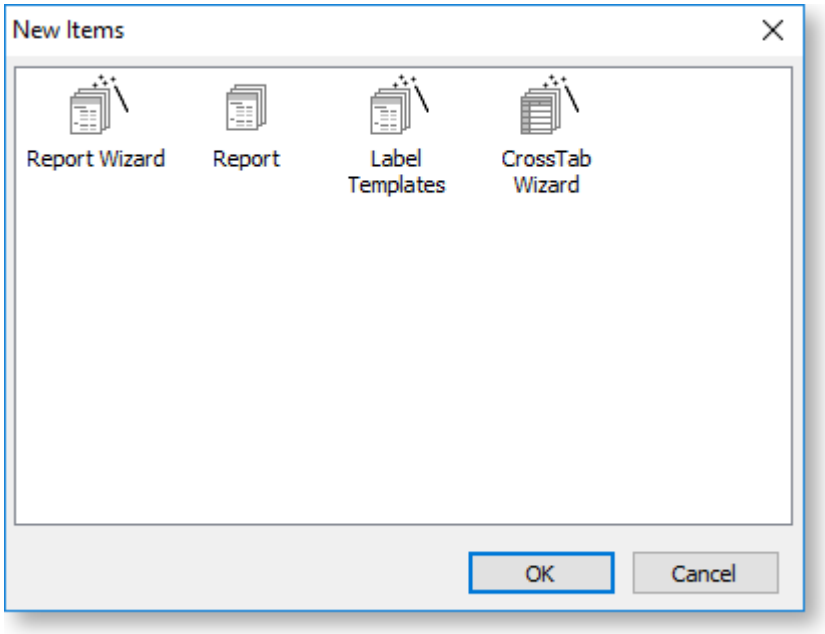


Printing Labels

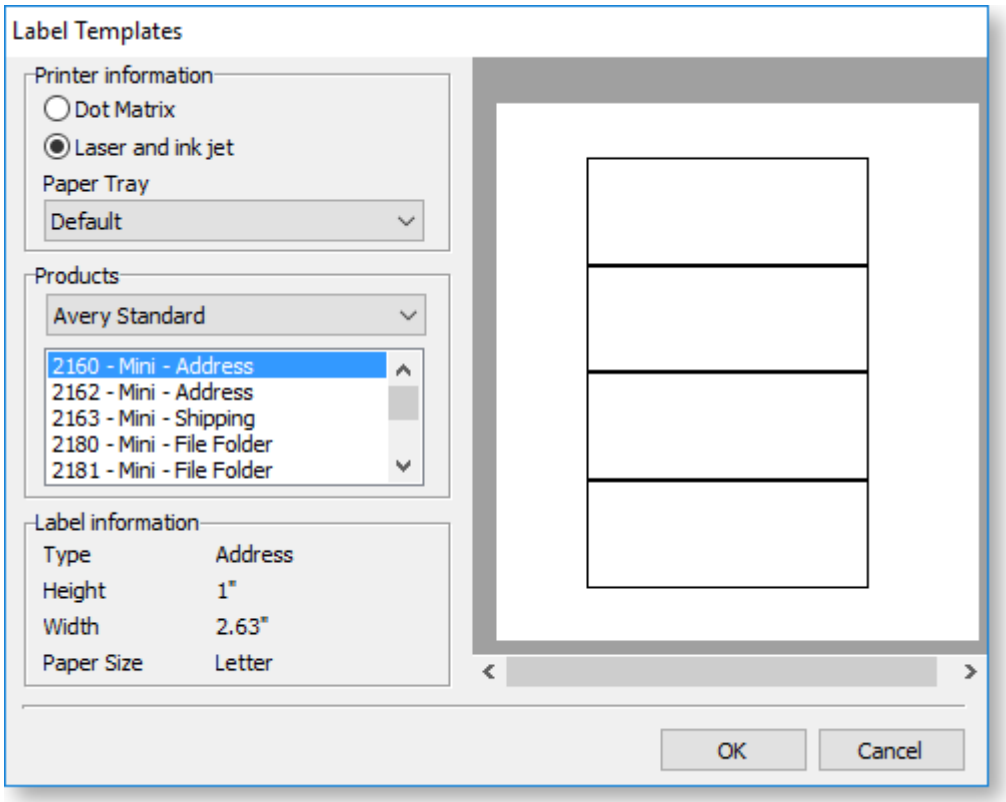
You can set up an Exo Clarity report to print out labels, for example address labels for mailouts to contacts or box labels for stock items.

Creating a Label Report

Select **New** from the File menu and chose **Label Templates** from the New Items window:



The Label Templates window opens:



Set the printer details, select a format for the labels from the Products section, then click **OK** to create the report.

Printing Multiple Labels per Document Line

To print multiple labels per line:

1. In the Report Tree, set the **BandsPerRecord** property of the Detail object to the quantity of labels required per line.

2. Keep track of the number of times the detail is printed so that the value may be assigned to the print count property of the detail in a later event. This may be done by using setting up a variable on the Calc tab to keeping a running balance of the number of times the detail is printed:

```
Detail.BandsPerRecord := Master['Ord Quant'];
Variable1.Value := Variable1.Value + 1;
```

3. Assign the value of Variable1 to the print count property of the detail:

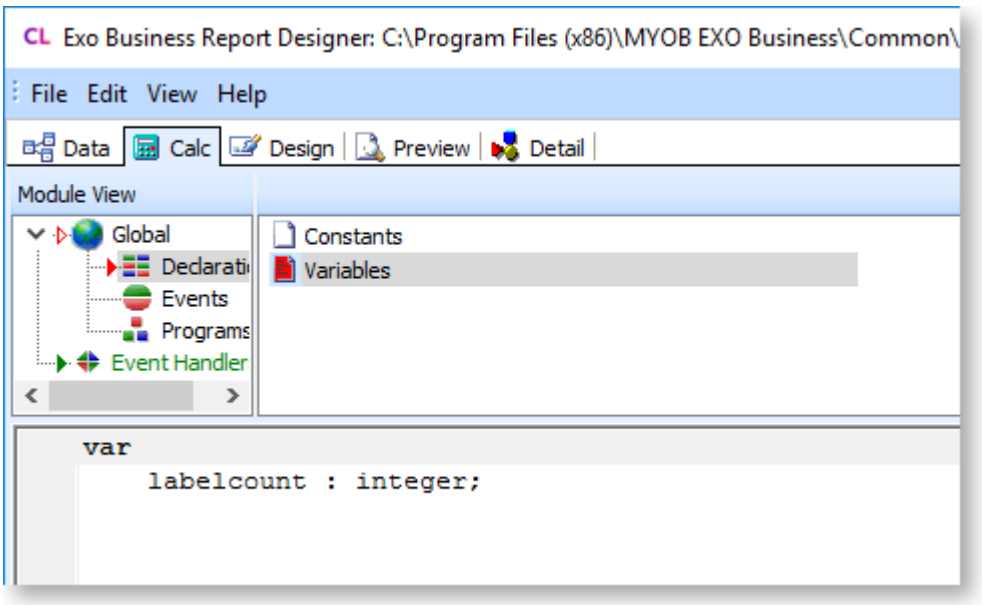
```
Detail.PrintCount := Variable1.Value;
```

Note: See the **SOLine.CLF** form in the Clarity Masters for an example of this function.

Printing Label Counts

As well as producing multiple labels per document line you may want to display a label count like "Box X of Y" on each label. To do this:

1. Declare a global variable called **labelcount**:



2. Increment the value of the global variable (to be displayed as X) until it equals the **bandsperrecord** property (to be displayed as Y) then set it back to zero for the next set to begin:

```
procedure DetailBeforeGenerate;
 begin
   labelcount := labelcount + 1;
   BoxXofY.text := 'Box' + inttostr(labelcount) + ' of ' +
inttostr(detail.bandsperrecord);
   if labelcount = detail.bandsperrecord then
     labelcount := 0;
     Variable1.value := Variable1.value + 1;
 end;
```

Mail Merge

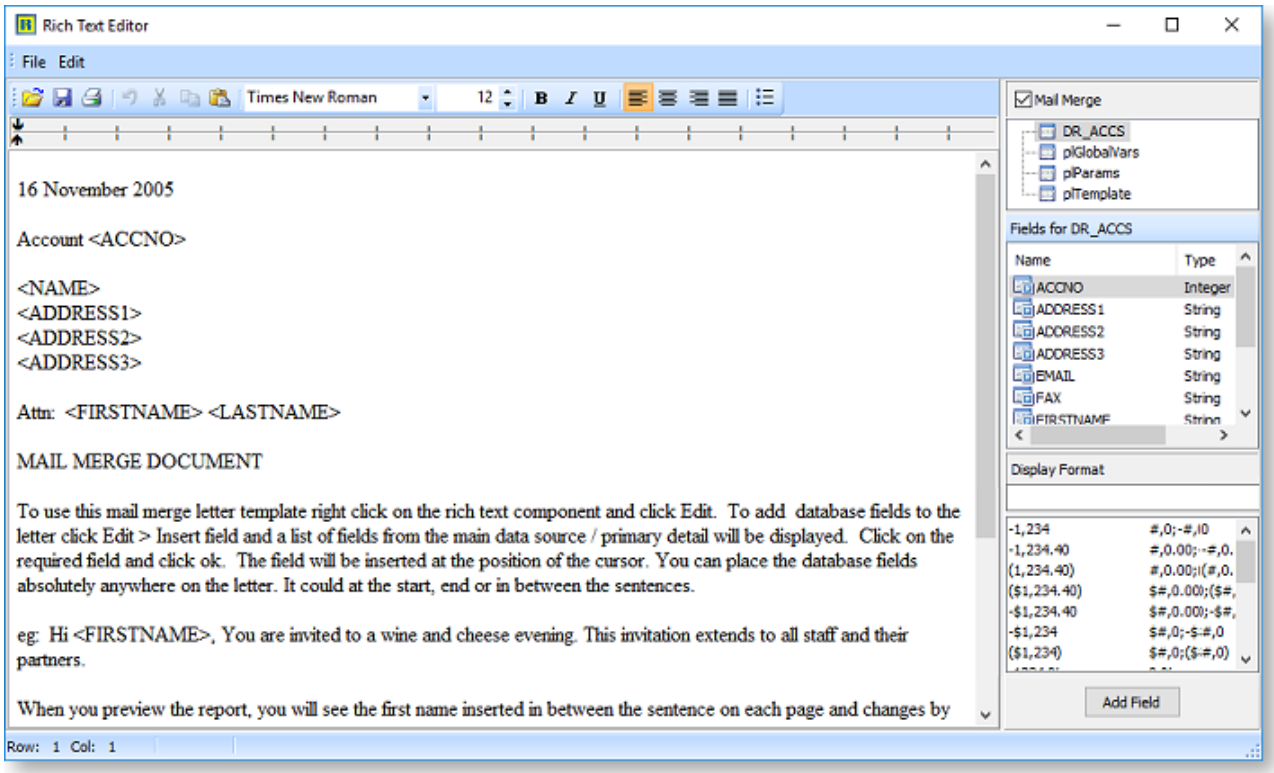
You can use Exo Clarity to create mail merge templates for mailouts to customers. The basic content of each mailout will be the same, but each will contain information that is specific to the individual recipient, such as name, address, or some other piece of personal data. For example, in a letter that announces a new product, your company logo and the text about the product will appear in each letter, but the address and greeting line will be different in each letter.

Using an Exo Clarity mail merge template you only need to set up the content once, then add database fields for the information that is unique to each mailout. Mail merge templates can be formatted as Rich Text for printing, or as HTML for sending as emails. Mail merge templates are saved with the extension .CLM for rich text templates and .CLE for HTML templates, and can be used when sending bulk communications in Exo Business using the mailshot function.

Note: Sample mail merge templates are available in the Clarity master folder.

Rich Text Mail Merge

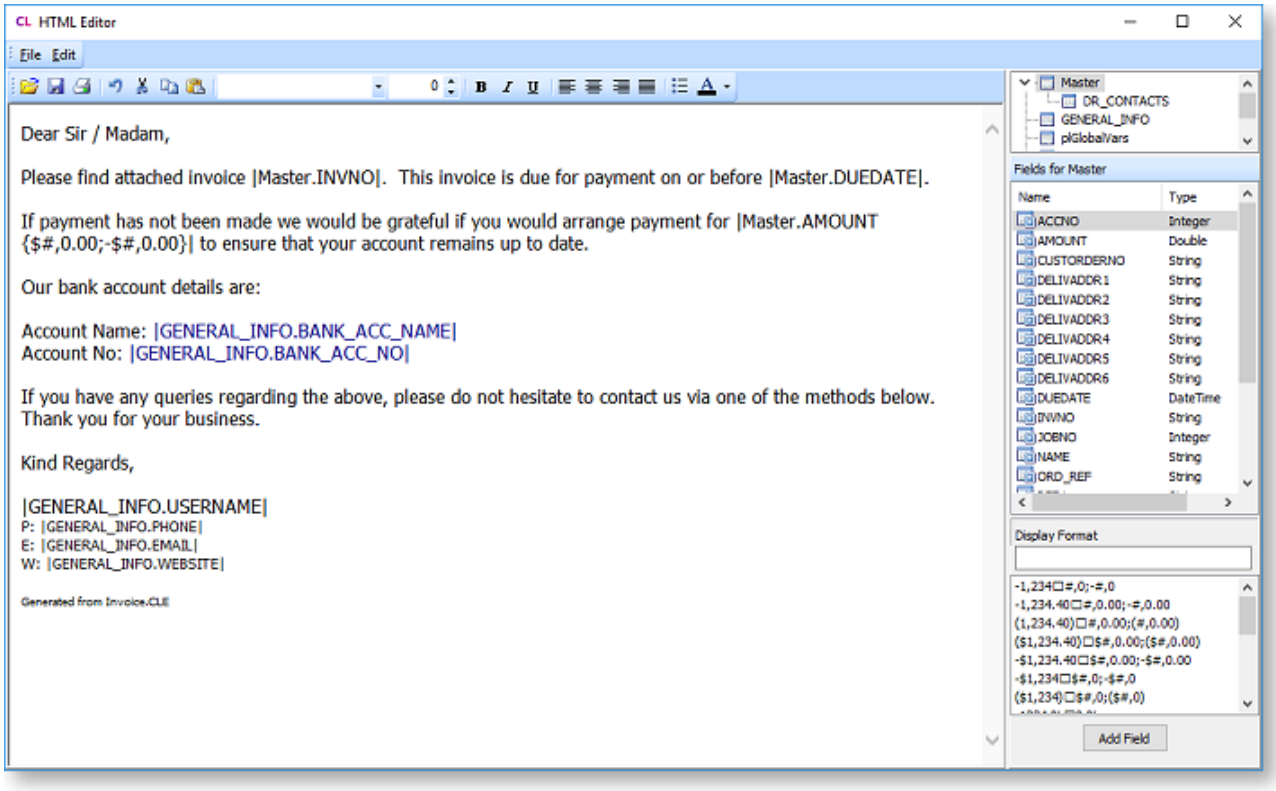
To create a CLM mail merge report, add a Rich Text component to the report. Resize it as necessary, then right-click and select **Edit**. The Rich Text Editor window opens:



Tick the **Mail Merge** box to tell Exo Clarity that this component is for a mail merge - doing so also enables the data controls on the right of the window. Enter the text of the mailout into the main window, inserting fields to be replaced with customer-specific details as necessary. To add a field, select a data source from the box at the top right of the window, then select a field from the box below. Choose a display format if necessary, then click **Add Field** to insert the field into the body of the mail merge at the current cursor position. You can also double-click on a field to add it to at the current cursor position.

HTML Mail Merge

To create a CLE mail merge report, add an HTML component to the report, and rename it to **HTML_EMAIL_BODY** in the Report Tree - the presence of a component with this name tells Exo Clarity that the template is an HTML template. Resize the component as necessary, then right-click and select **Edit HTML**. The HTML Editor window opens:



Edit the content of the mail merge in the same way as for Rich Text mail merges. Select a data source, field and display format from the controls on the right of the window, then click **Add Field** to insert the field into the body of the mail merge at the current cursor position. You can also double-click on a field to add it to at the current cursor position.

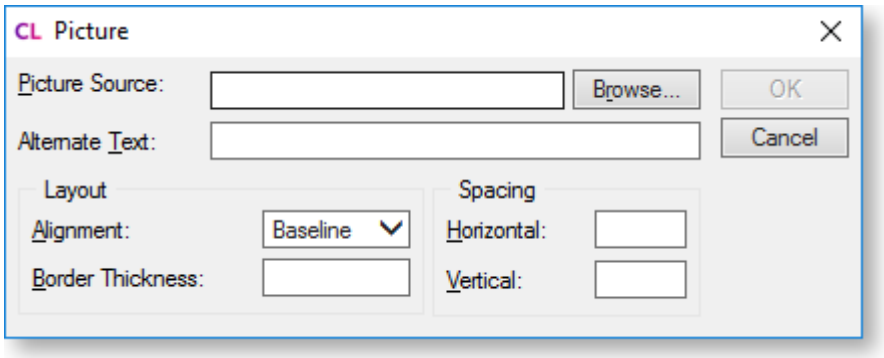
Editing HTML

The HTML Editor window is a WYSIWYG editor. Basic formatting options are available to change the font, size and colour of the text; to make the text bold, italic or underlined; or to organise text into bulleted lists.

> **Note:** When adding text formatting to a database field, it is best to apply the formatting to the entire text of the field, including the pipe characters.

Hyperlinks can be added by selecting text, right-clicking and selecting **Edit hyperlink** from the right-click menu.

Images can be added by right-clicking on the text and selecting **Insert image** from the right-click menu. A window opens for specifying the image source and properties:



Subject Line and Alternative Text

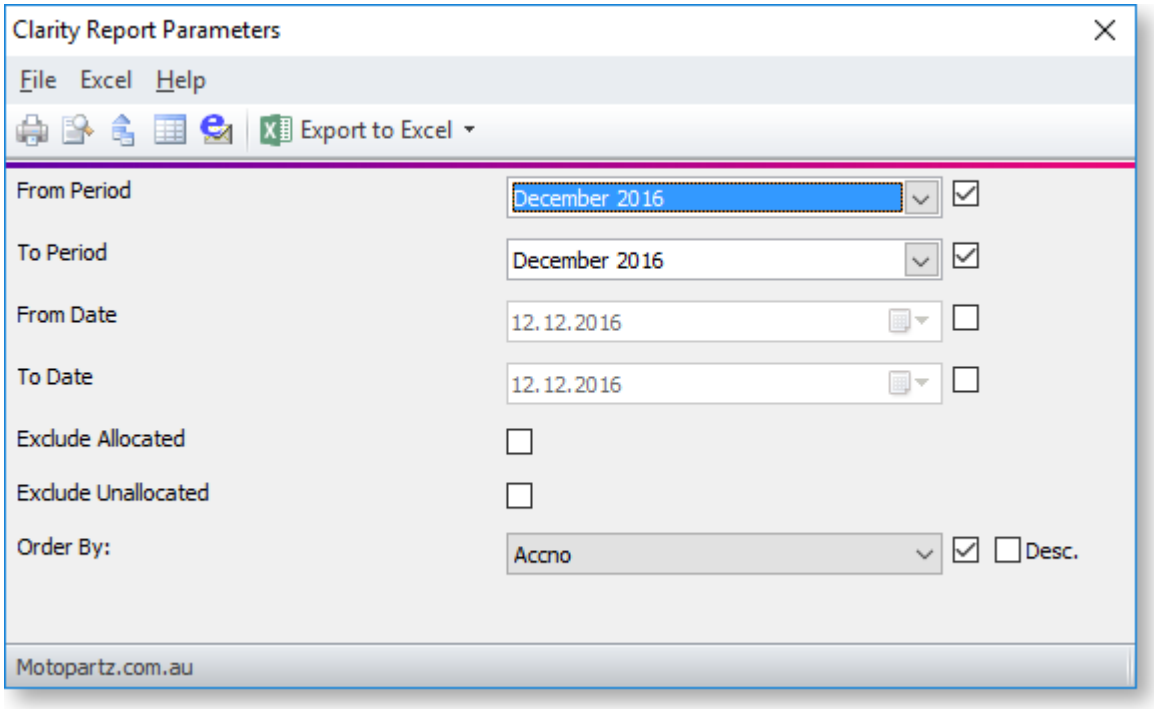
In addition to the **HTML_EMAIL_BODY** component, you can optionally add two Rich Text components to supply more information for the email template:

- You can add a Rich Text component called **EMAIL_SUBJECT** to specify the subject line of emails that are sent using this template. The component can contain data fields to customise the subject line, e.g. "Invoice for <COMPANYNAME>"
- You can add a Rich Text component called **EMAIL_TEXT_BODY** to contain alternative content to be used for plain text emails.

Report Parameters

About Report Parameters

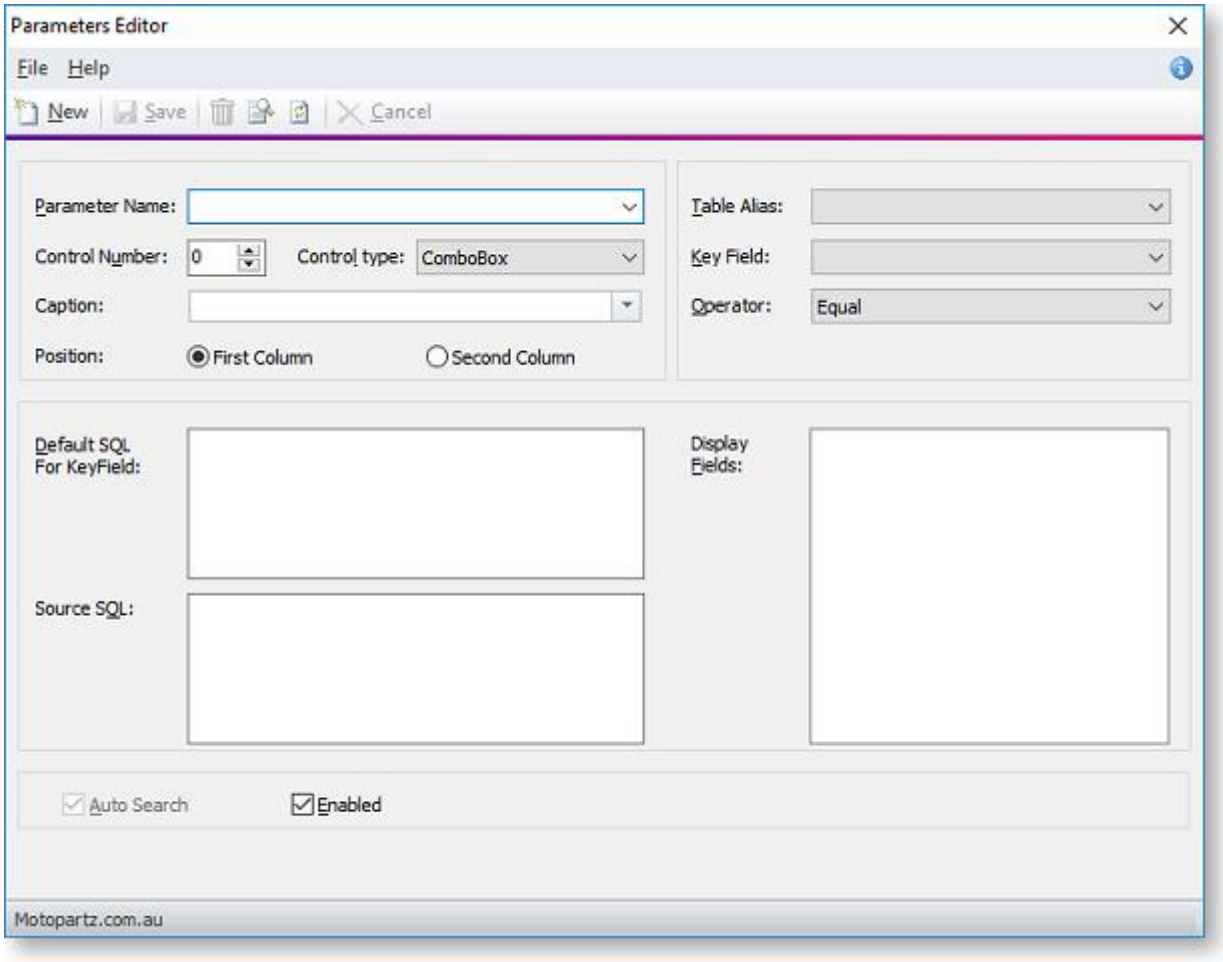
When a report is run, it can prompt the user for parameters that will be fed into the report. Common parameters include "From" and "To" options to set a date range for the report, or ordering options. Parameters are selected on the Clarity Report Parameters window, which appears when the report is run:



Note: If you choose to preview the report from this window, the parameter controls are displayed on the left of the Print Preview window - this avoids the need to close the preview and return to the parameters window if you want to change the selected values.

Adding Parameters

Parameters are added to a report and configured using the Parameters Editor window - open it by selecting **Runtime Parameters** from the Tools menu.



Note: Settings for all report parameters are saved in a separate file. This file has the same name as the report file, with the extension .CLS. The parameter settings file is a plain text file, which you can edit in any text editor.

Field	Description
Parameter Name	This is the name of the parameter that will be used when the parameter value is accessed in code on the Calc tab.
Control Number	This determines the order of the parameters when they show on the parameter dialogue box. 0 is first, 1 is second, and so on.
Control Type	The type of control that will appear on the Clarity Report Parameters window (text box, combo box, etc).
Caption	The label that will appear alongside the control on the Clarity Report Parameters window. The caption is important because it will describe to the user which field the selection criteria will control (see item 6 in the example below). **Note:** Captions can include the Display Names that are set up in Exo Business Config. Enter @PARAMETER_ID, where PARAMETER_ID is the field's ID on the Display Names screen in Exo Business Config. For example, if you enter @STOCKGROUP_PRIMARY in a caption, at runtime it will be replaced with the Display Name that has been set up for Primary Stock Group. The dropdown control on this field lets you select an @ parameter from a list of available ones.

Position	Parameters can be arranged into two columns on the Clarity Report Parameters window - select whether the parameter will be in the First (left) or Second (right) column. If all parameters are in the same column, they will all extend across the entire window.
Table Alias	The name of the data pipeline that the selection criteria affect.
Key Field	The field within the data pipeline that is filtered by the selection criteria.
Operator	The search operator that is applied to the key field, e.g. equal / greater than / less than.
Default SQL for Key Field	Used with Listbox, Combobox, Checkbox, Textbox, OrderBy, DebtorPeriod, CreditorPeriod, StockPeriod and GLPeriod parameters. The value that is entered into this box determines what is selected by default in the Listbox / Combobox when the Clarity Report Parameters window pops up. For OrderBy fields, this field becomes **"Order By" Fields** and contains a comma-separated list of fields that you can select to order your data by (the labels on this window change to reflect this when you select an OrderBy type parameter). For the period selection control types, a short SQL query must be used to specify the default, such as: SELECT SEQNO FROM Period_Status WHERE Ledger = 'D' and AGE = 1
Source SQL	Only used with Listbox, Combobox and OrderBy components. The SQL that will retrieve the values to be displayed in the Listbox / Combobox when the Clarity Report Launcher pops up. The name of the field for your Key Field must match one of the names of the fields in your source SQL statement. If there's a difference (e.g. your Key Field is HDR_SEQNO and your combo has the field SEQNO), change "SEQNO" to "SEQNO AS HDR_SEQNO". This is called aliasing a field and allows Clarity to match that field to the Key Field. For OrderBy parameters, this field becomes **Default**, and determines the default OrderBy field. For Search parameters, the field becomes **Filter SQL**, where you can enter conditions to filter the search results.
Display Fields	The fields that are highlighted in this box are the fields that will be displayed in a Listbox or Combobox when the Clarity Report Parameters window pops up. To enter values into this box, double click on the SQL statement in the Source SQL box. If it is a valid SQL statement, the list will automatically populate. The key field must be included in this list to be able to save the parameter. Of the selected fields, you can determine which ones will actually show by CTRL-clicking to select and de-select fields.
Enabled	Whether or not the parameter is enabled by default. Enabling / disabling can also be achieved at runtime by checking or un-checking the checkbox displayed alongside the parameter.
Visible lines	Used with Listbox parameters. The number of lines high the control should be on the Clarity Report Parameters window - if there are more options than this, a scroll bar will be available.

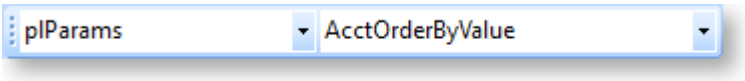
	Note: If the data is too wide to fit in the list box, there will be a horizontal scrollbar at the bottom of the list - this will take up one of the available lines.
Add to days	Used with Date parameters. Adds the specified number of days to the date that is selected and passes that date to the report.

Displaying Parameters on the Report

All of the parameters you set up will appear on the Clarity Report Parameters window; to make a parameter value appear on the report itself, you can add it as a DBText field. Instead of selecting a data source, select 'plParams' from the first dropdown on the Edit toolbar, and select the parameter to display from the second dropdown. For each parameter, there will be three options:

- ParameterName - displays the parameter's caption.
- ParameterNameEnabled - displays "Y" or "N" depending on whether or not the parameter is enabled.
- ParameterNameValue - displays the value entered for the parameter by the user.

For example, in the image below, the parameter is 'AcctOrderBy' - this option will display the value that the user entered for the 'AcctOrderBy' parameter.

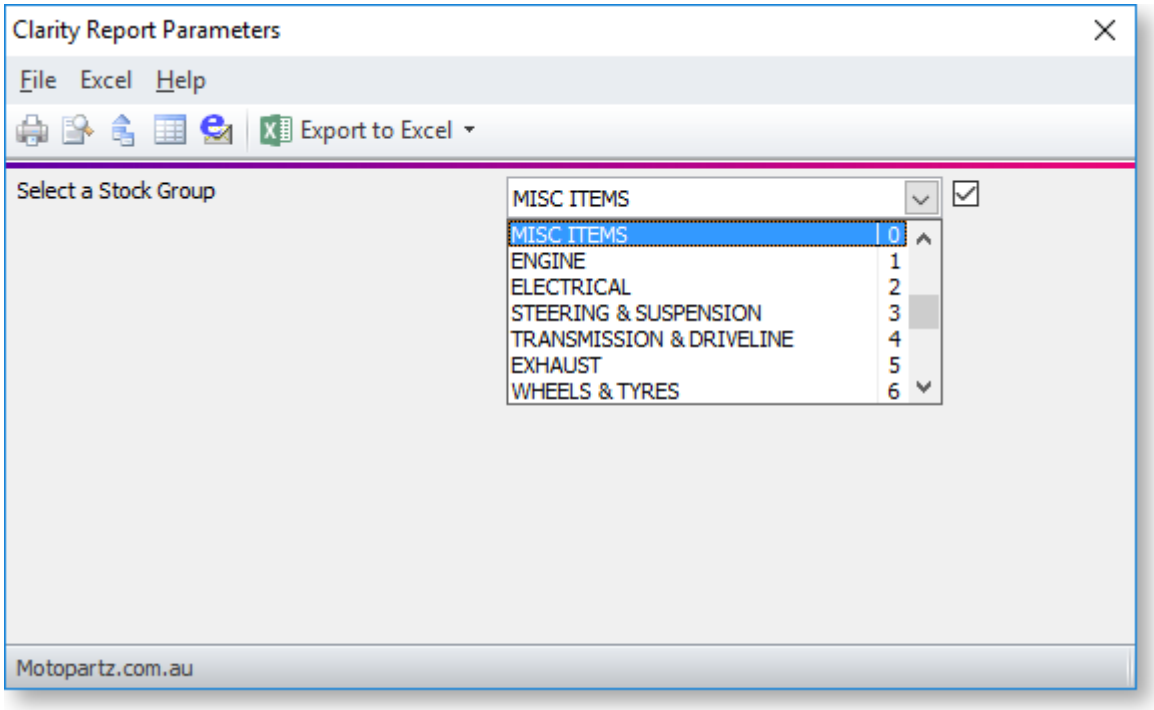


Parameter Control Types

The **Control type** selected for a report parameter determines how the user will be able to specify the parameter when the report is run. Each available control type is discussed below, with information on the settings that must be configured for each type.

ComboBox

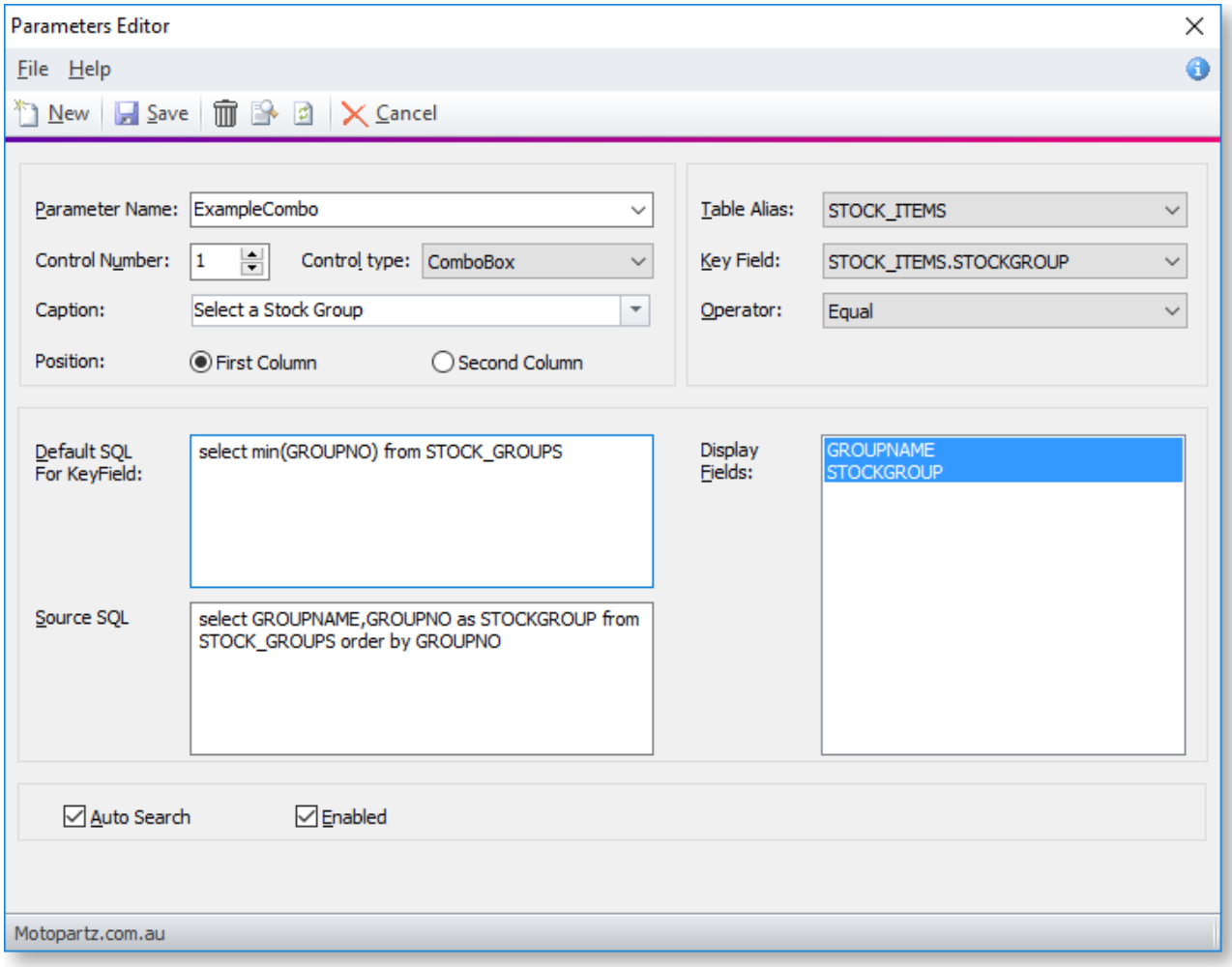
Displays a dropdown list that is used to select a single option:



The Parameters Editor window must be set up carefully to ensure that the parameter behaves as expected:

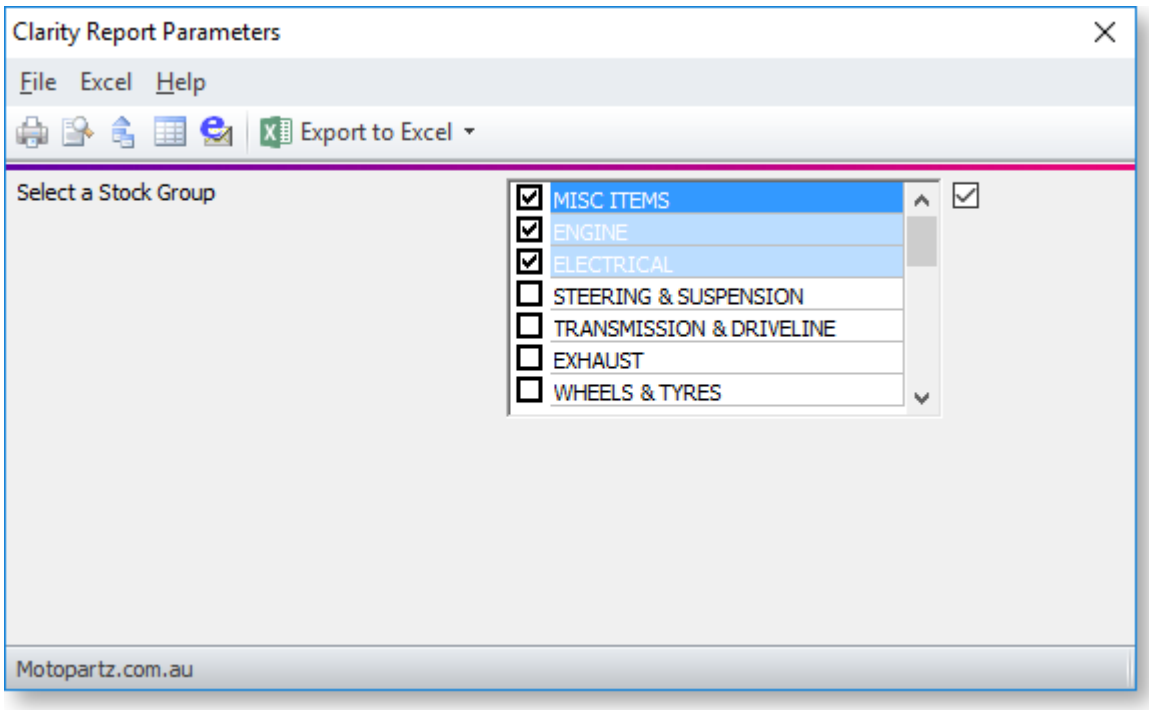
- The SQL statement in the **Default SQL for Keyfield** area determines which item will be selected by default.
- The SQL statement in the **Source SQL** area is used to populate the combo box.
- The selections in the **Display Fields** area determine which of the fields set up by the **Source SQL** will appear in the combo box. (After entering a **Source SQL** statement, double-click on it to populate the **Display Fields** area.)

The combo box in the screenshot above was defined with the following settings:



ListBox

Displays a list of options with check boxes, allowing multiple items to be selected:

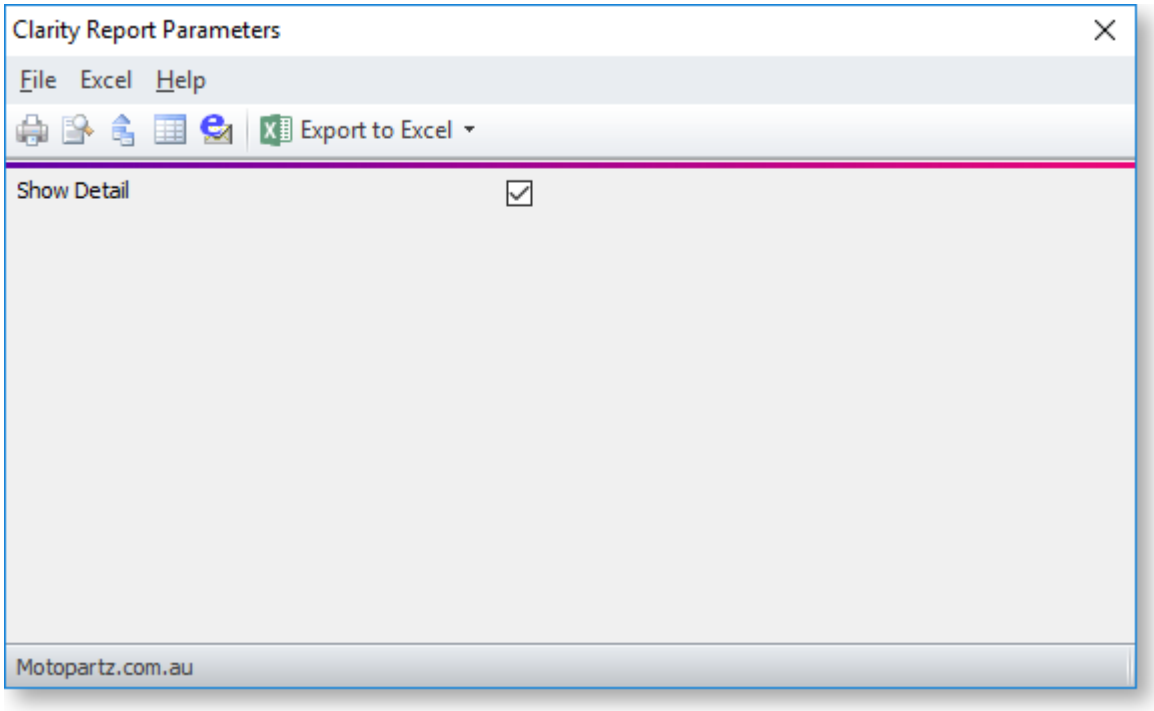


ListBox parameters are set up in the same way as ComboBox parameters. An extra **Visible lines** setting is available for ListBoxes - this controls how many lines tall the box will be (in the screenshot above, this is set to 7.)

Note: If the data is too wide to fit in the list box, there will be a horizontal scrollbar at the bottom of the list - this will take up one of the available lines.

CheckBox

Displays a checkbox that can be ticked or cleared:

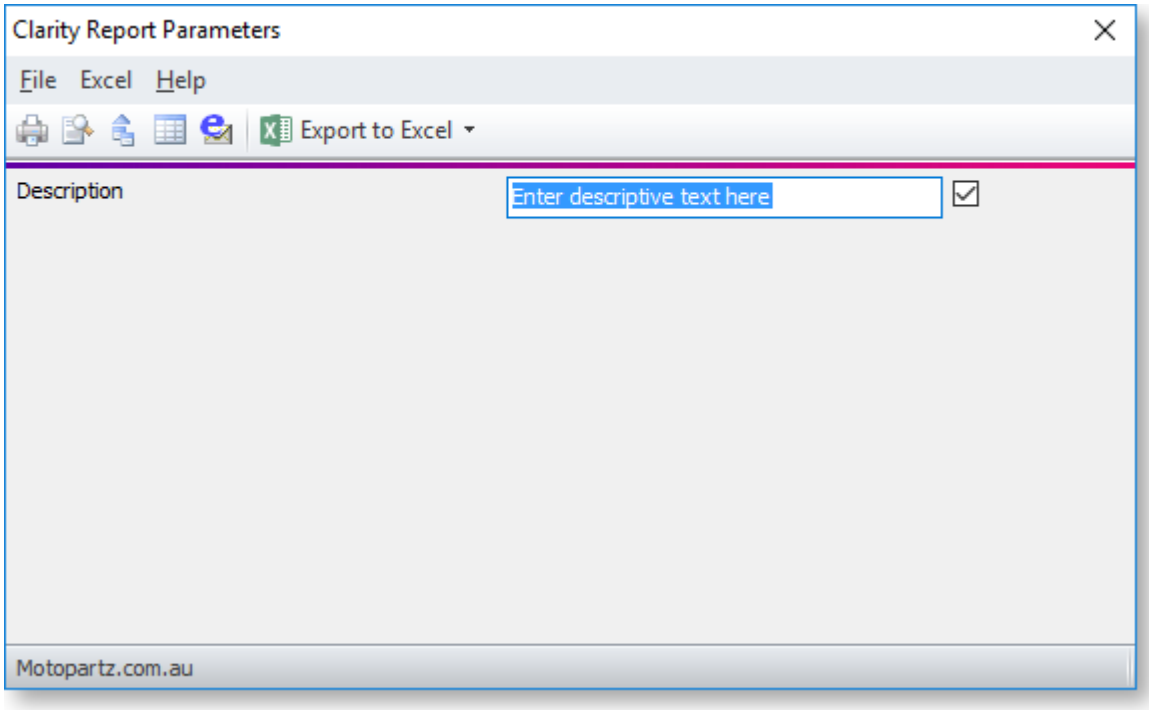


A Checkbox parameter returns a boolean value that is true if the box is ticked. CheckBox parameters are not used to filter the report results; they are intended to be used in code on the Calc tab. For this reason, you don't need to configure any other settings on the Clarity Report Parameters window.

Note: Parameter values are available under the "plParams" data source on the Calcs tab.

TextBox

Displays a field where text can be entered:

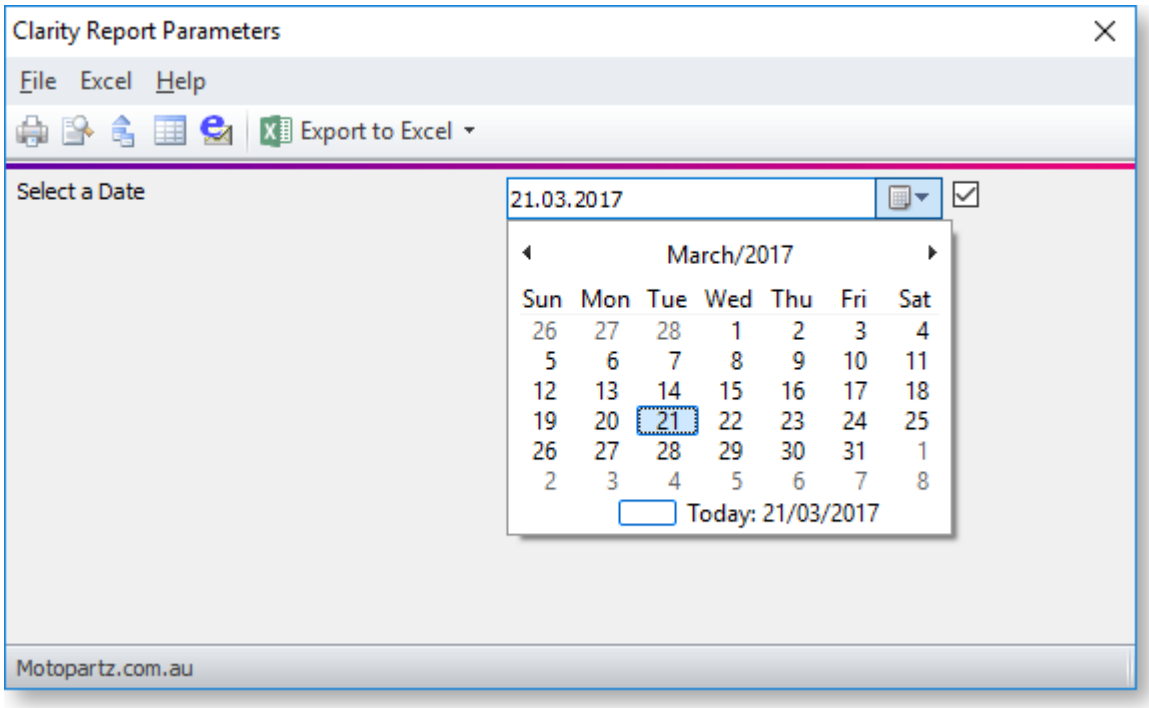


Like the CheckBox, this control type is not bound to a data field and does not filter report results, but is available for use by data-aware components and in calculations. For example, you could add a DBText component to a report that takes its value from the value of a TextBox parameter.

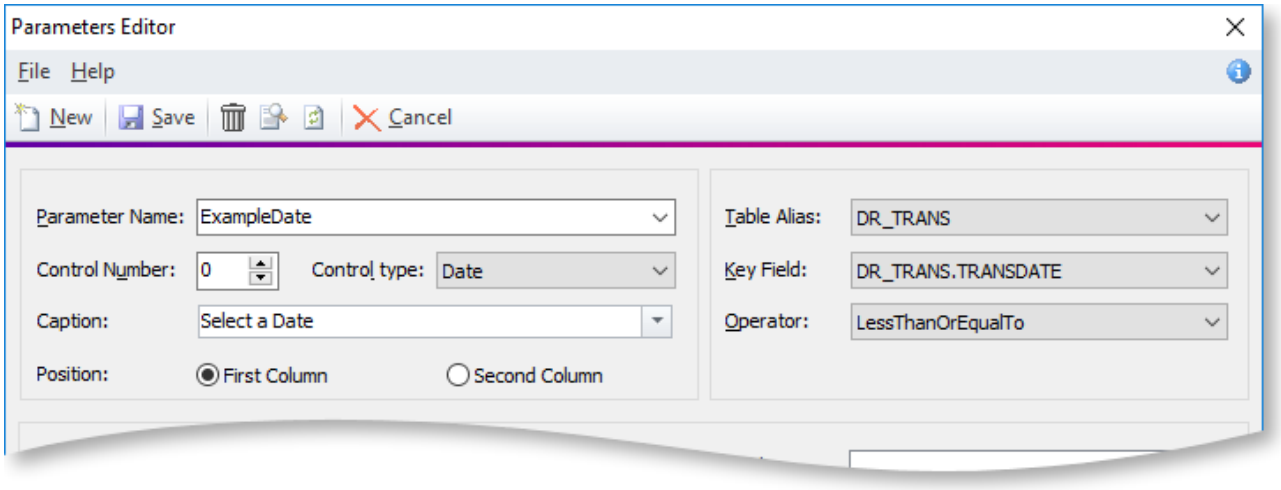
You can set default text for the parameter by entering it into the **Default SQL for Keyfield** field.

Date

Displays a standard date selection control that can be used to filter report results by date:

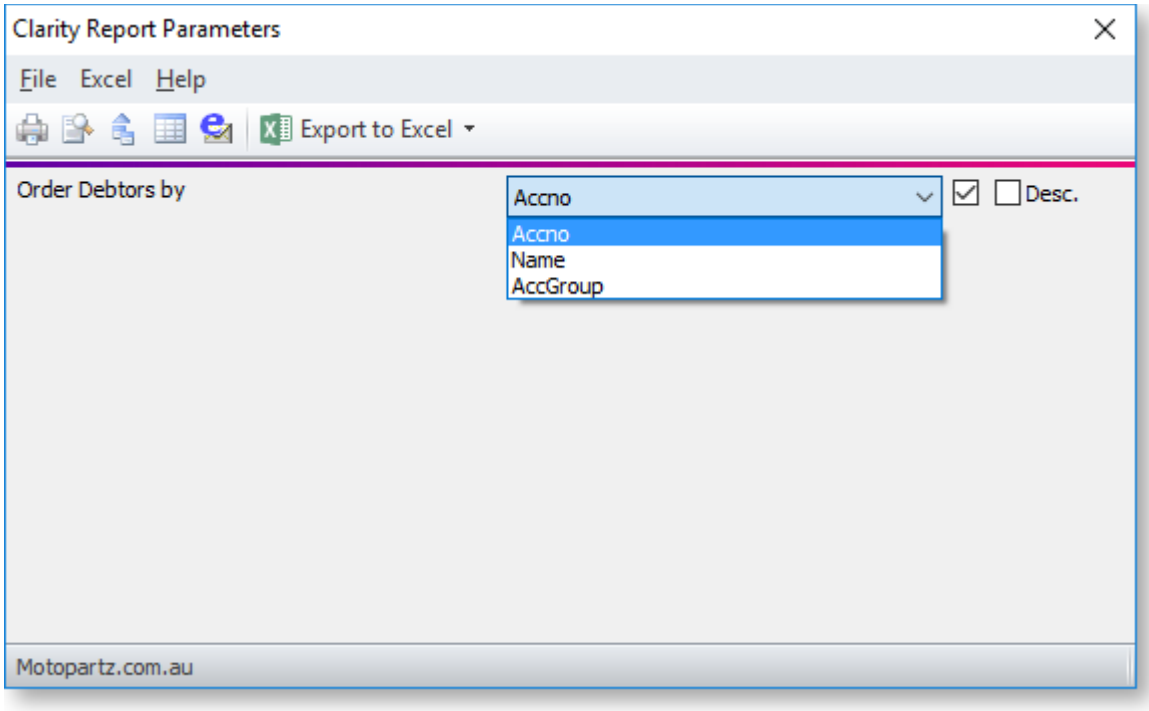


The **Table Alias**, **Key Field** and **Operator** settings let you control how results will be filtered, e.g. the following configuration will return all Debtor transactions from today or earlier:

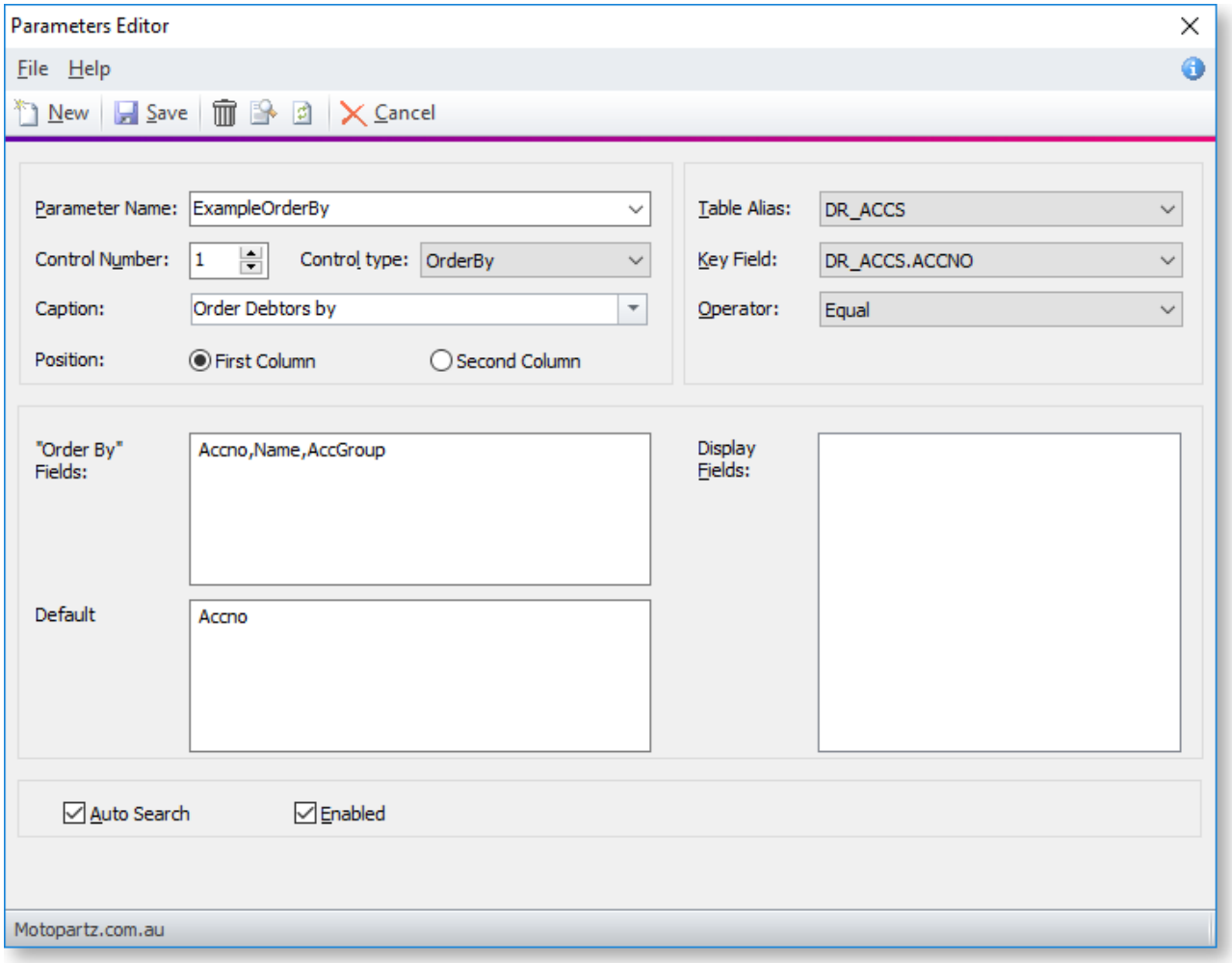


OrderBy

Displays a dropdown that lets the user choose which field to order report results by, including a **Desc.** box to set whether ordering should be in ascending (unticked) or descending (ticked) order:



When a parameter is set to "OrderBy", the **Default SQL for Keyfield** field becomes the **"Order By" Fields** field - this is where you enter the options for the dropdown. Enter fields from the selected table, separated by commas. These fields are not validated - any field that doesn't exist in the selected table will simply not appear in the dropdown. You can specify the default option in the **Default** field.

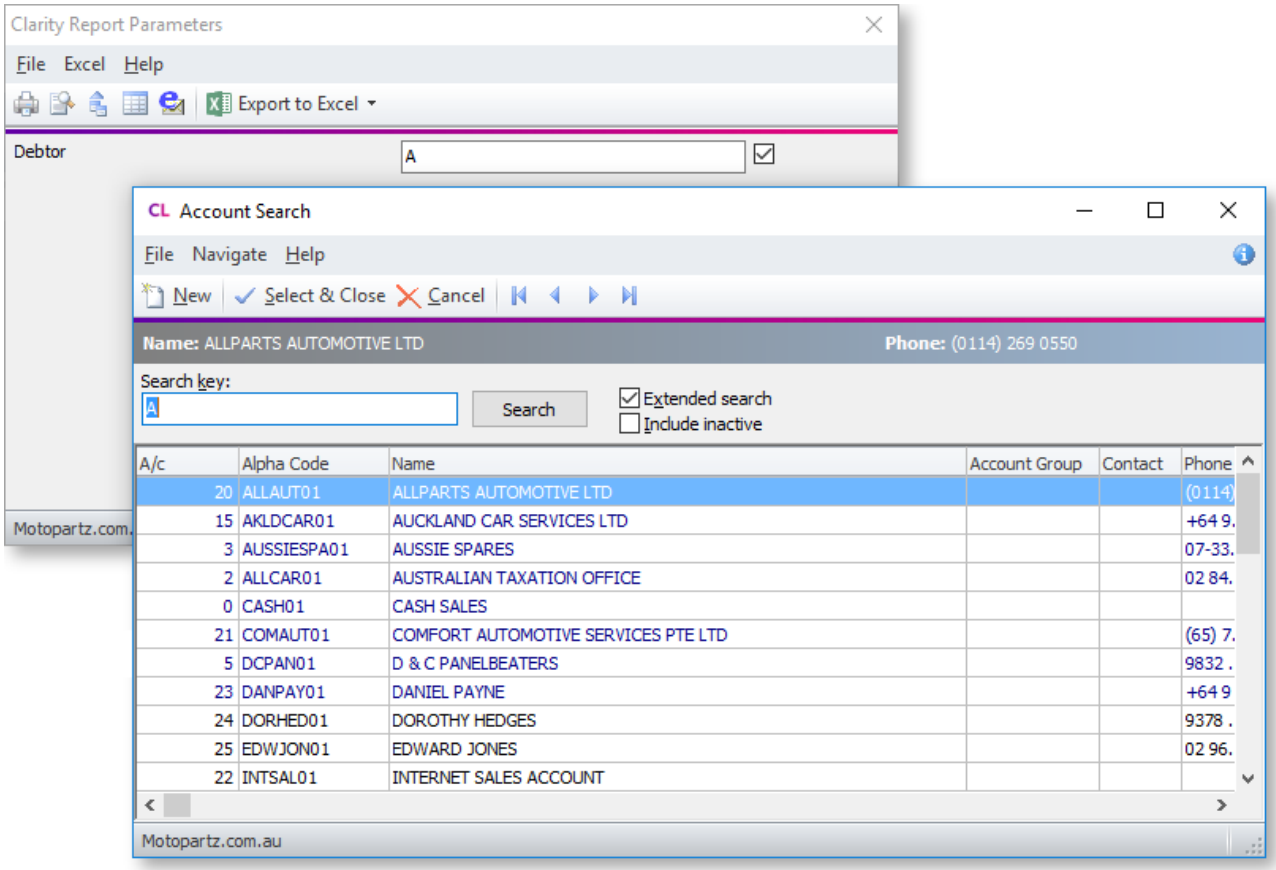


Search

Several control types are available for searching different type of records:

- DebtorSearch
- CreditorSearch
- StockSearch
- GLAccSearch
- GLAccGroupSearch
- AnalysisCodesSearch

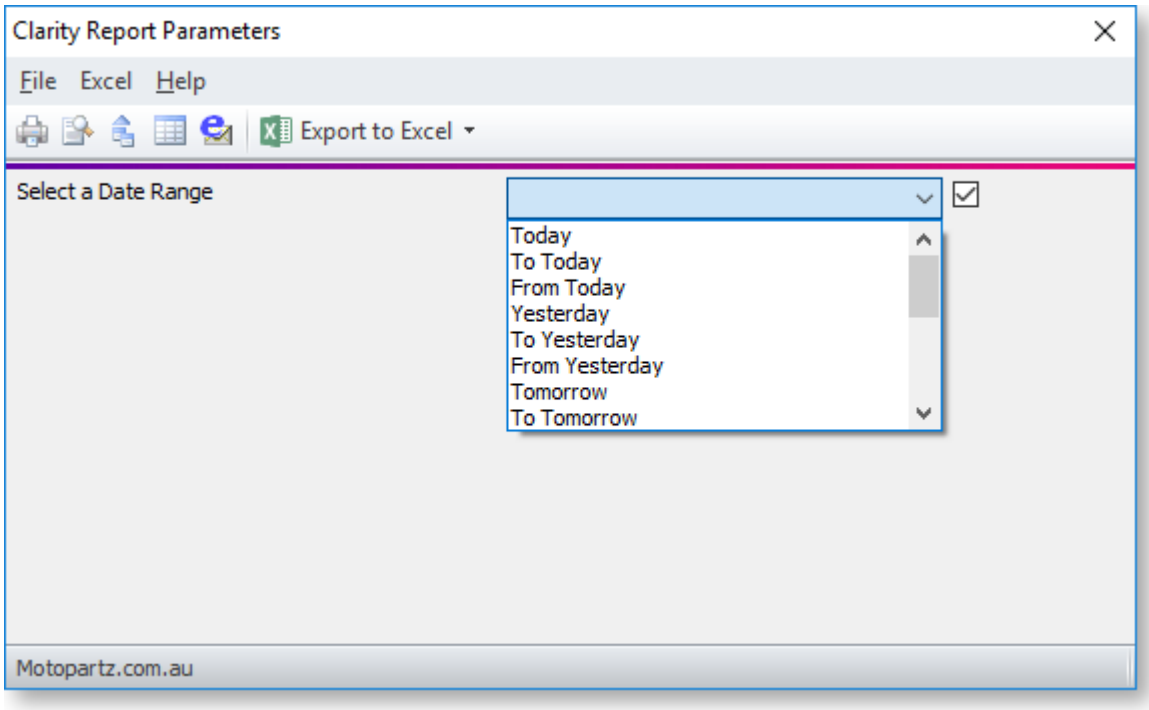
Each of these control types displays a text box where the user can enter search terms (or enter "?") and press ENTER to open a search window for the appropriate record type.



When a search control type is selected, the **Source SQL** field becomes the **Filter SQL** field, where you can enter conditions to filter the search results on, e.g. for a DebtorSearch parameter, enter ACCGROUP = 1 to filter search results to include only Debtor accounts that belong to Account Group 1.

DateRange

Displays a date range control that allows you to select one of a group of pre-set ranges relative to today's date, e.g. To Yesterday, From Today, This Week, Next Month.



Note: If you want to be able to specify a range between two specific dates, you will need to use two Date parameters: one for the From date and one for the To date.

Spacer

This control type can be used to control the layout of the Clarity Report Parameters window. It leaves a blank space at the position indicated by its **Control Number** - no other settings are needed.

Line

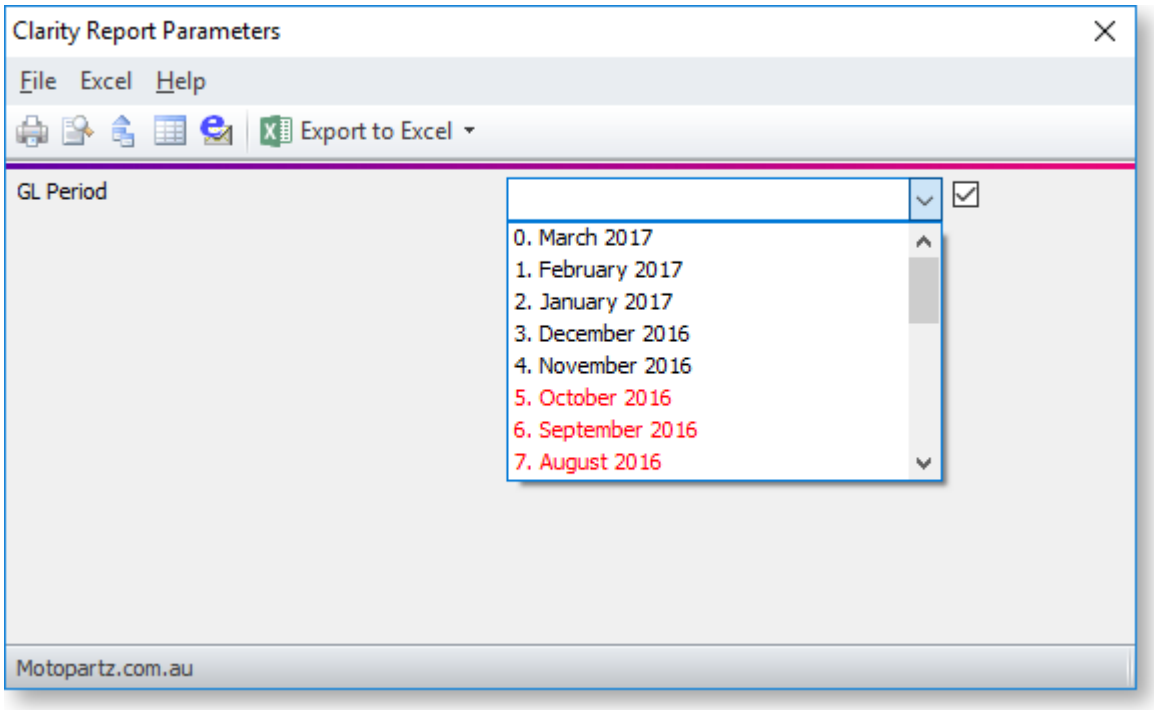
This control type can be used to control the layout of the Clarity Report Parameters window. It draws a straight horizontal line at the position indicated by its **Control Number** - no other settings are needed.

Period

Several control types are available for selecting an accounting period:

- DebtorPeriod
- CreditorPeriod
- StockPeriod
- GLPeriod

These control types display a period selection control for the appropriate ledger:



These control types are not bound to a data field and do not filter report results, but they available for use by data-aware components and in calculations, e.g. when setting up parameters to use the global Financial Period selection in the Exo Accountant's Assistant module.

Note: Period section controls always use the period ID (SEQNO). This means that it is not possible to enter a simple age (e.g. 1) in the parameter's **Default SQL For Key Field** field; instead a short SQL query must be entered, such as: `SELECT SEQNO FROM Period_Status WHERE Ledger = 'D' and AGE = 1`

Advanced Parameter Features

Using Parameters with Stored Procedures and Functions

If SQL editing has been enabled on the SQL tab of the Query Designer, it is possible to take data from a function or stored procedure. In this case, you can use a runtime parameter as an input parameter to the stored procedure/function. To do this:

1. Create the parameter as normal.
2. On the Data tab, create a data source using the Query Designer.
3. On the SQL tab of the Query Designer, right-click and select **Edit SQL** to enable manual SQL editing.
4. Enter the SQL query that uses the stored procedure or function. To substitute a runtime parameter, enter the parameter name preceded by a colon parameters, e.g. SELECT *FROM FN_CR_AGEDBALANCES_BACKWARDS(**:Age**).
5. Click **OK**.
6. Go to the Calc tab and find the OnInitializeParameters Report event in the Events list. Right click on this event and select **New**.
7. Enter a script similar to the following to copy the values set in plParameters to the report parameters:

```
procedure ReportOnInitializeParameters(var aCancel: Boolean);
begin
 aCancel := False;
     { set Age parameter to value entered by user }
 Report.Parameters['Age'] :=  plParams['AgeValue'];
end;
```

Filtering Parameters Based on Other Parameters

You can set up a Listbox or Combobox parameter whose options are filtered depending on the value of another parameter. For example, a report could contain a "parent" parameter that specifies a GL account and a "child" parameter that specifies a subaccount - selecting an account in the first parameter filters the second parameter to show only subaccounts of the selected account.

To do this, in the **Source SQL** of the child parameter, use the name of the parent parameter preceded with a colon as a parameter in the SQL statement, for example:

```
select distinct s.subaccno, s.name from glsubaccs s join glaccs g on
g.accno = s.accno where g.accno = :GLAccount
```

In this example, the name of the parent parameter is "GLAccount".

Using $COMPUTERID and $USERID

You can include the $COMPUTERID and $USERID variables in statements in the **Default SQL for Keyfield** field, e.g.

```
SELECT BRANCHNO WHERE BRANCHNO = $COMPUTERID
```

This will set the branchno to the default branchno stored for your computer and will run the report for that particular branch. When you preview the report, the Branchno will default to the ID of the computer.

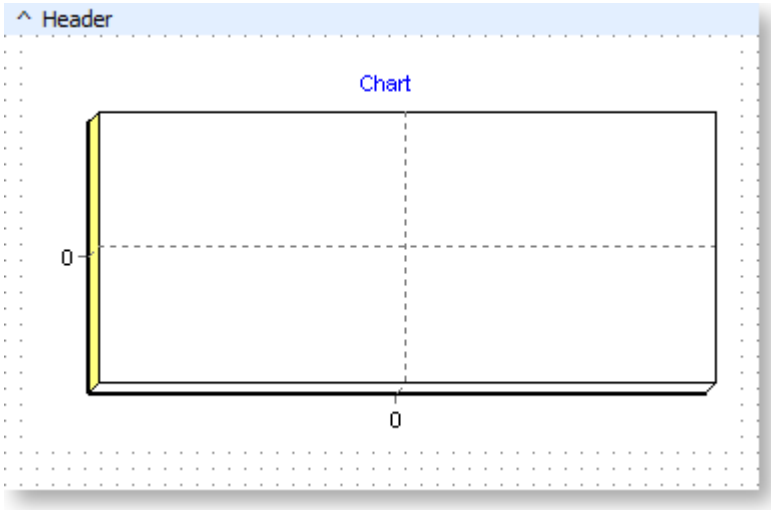
Charts

Adding Charts to a Report

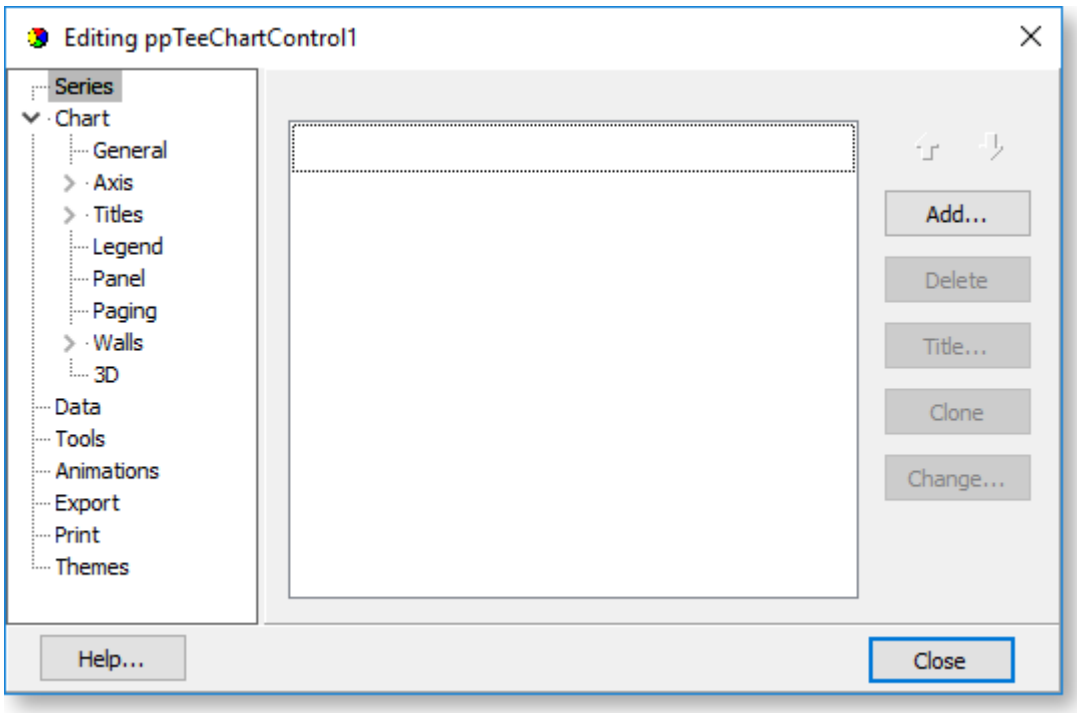
Exo Clarity lets you add graphical charts to a report, which can display data from a variety of sources. Add a chart using one of the following toolbar buttons:

- The Chart button on the Standard Components toolbar ( ) - this adds a chart component that is not bound to a data source.
- The DBChart button on the Data Components toolbar ( ) - this adds a data-aware chart component that can be bound to a data source that has been set up in the report.

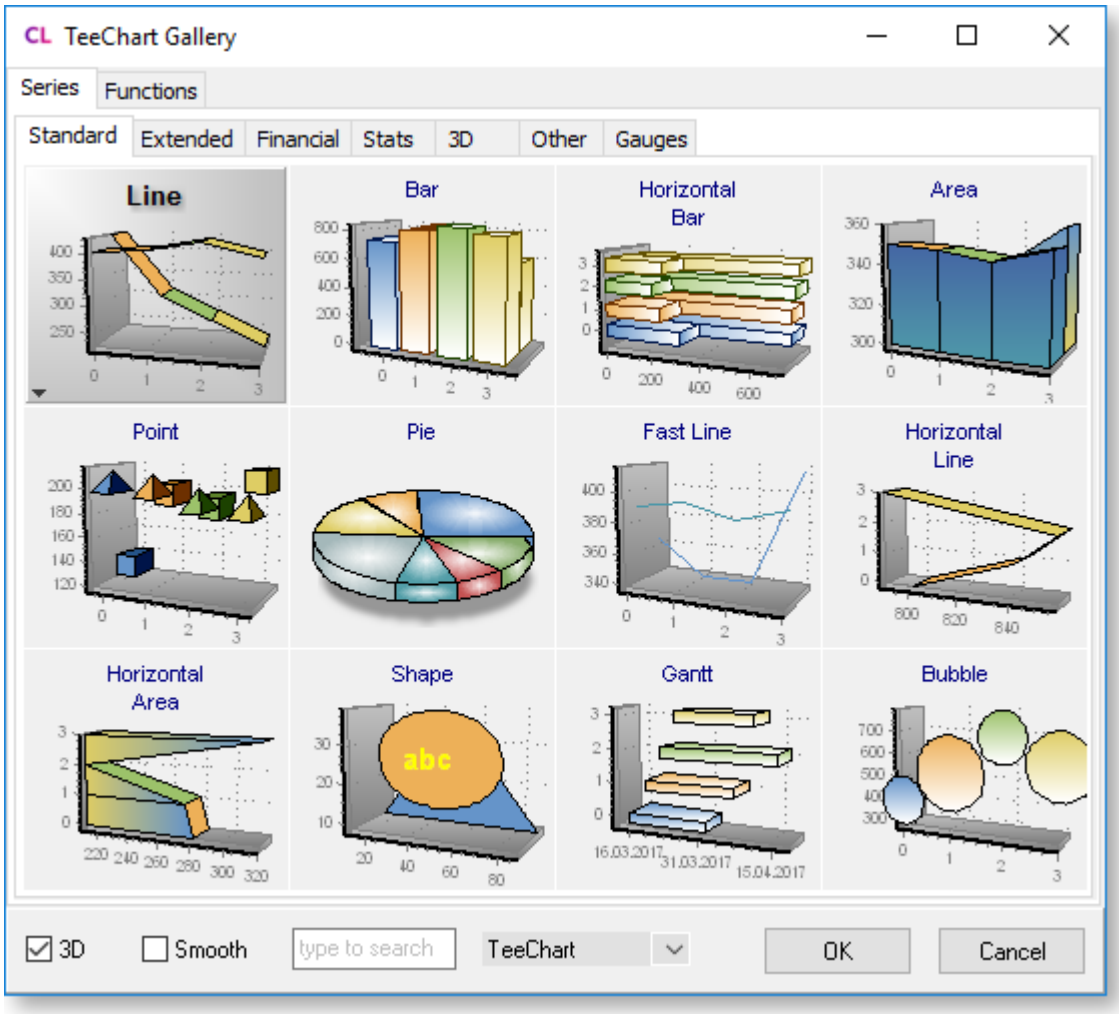
When a chart component is first added to a report, it contains no data:



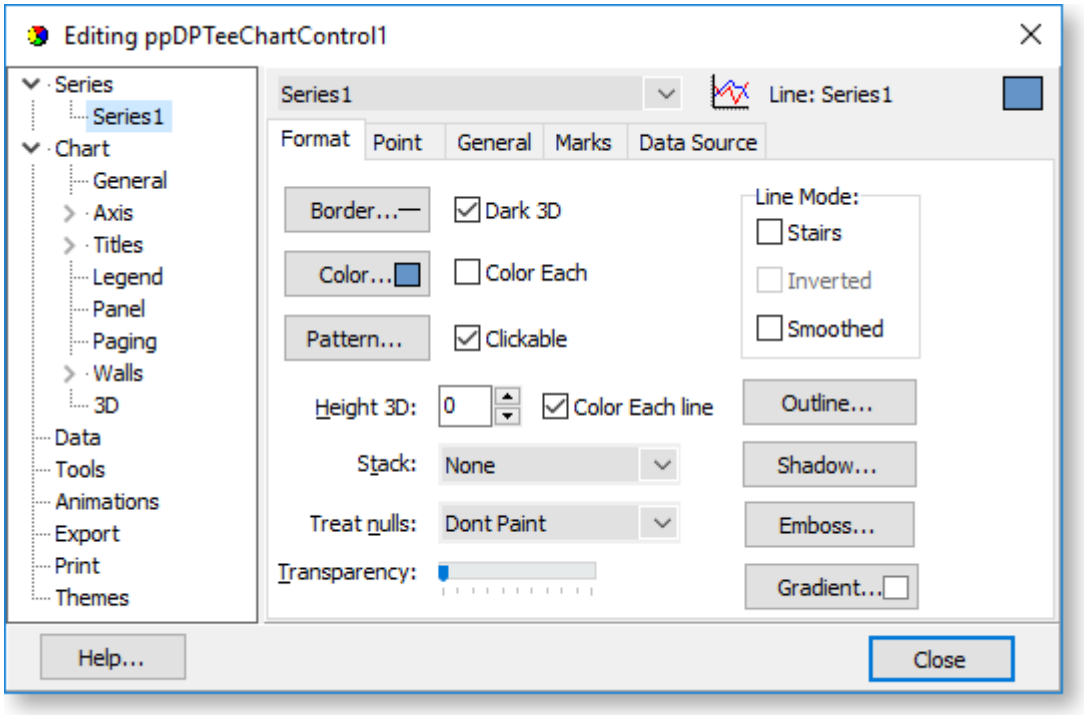
Right-click on the chart component and select **Edit Chart** to set up the properties of the chart. A window opens where you can set up the chart's properties:



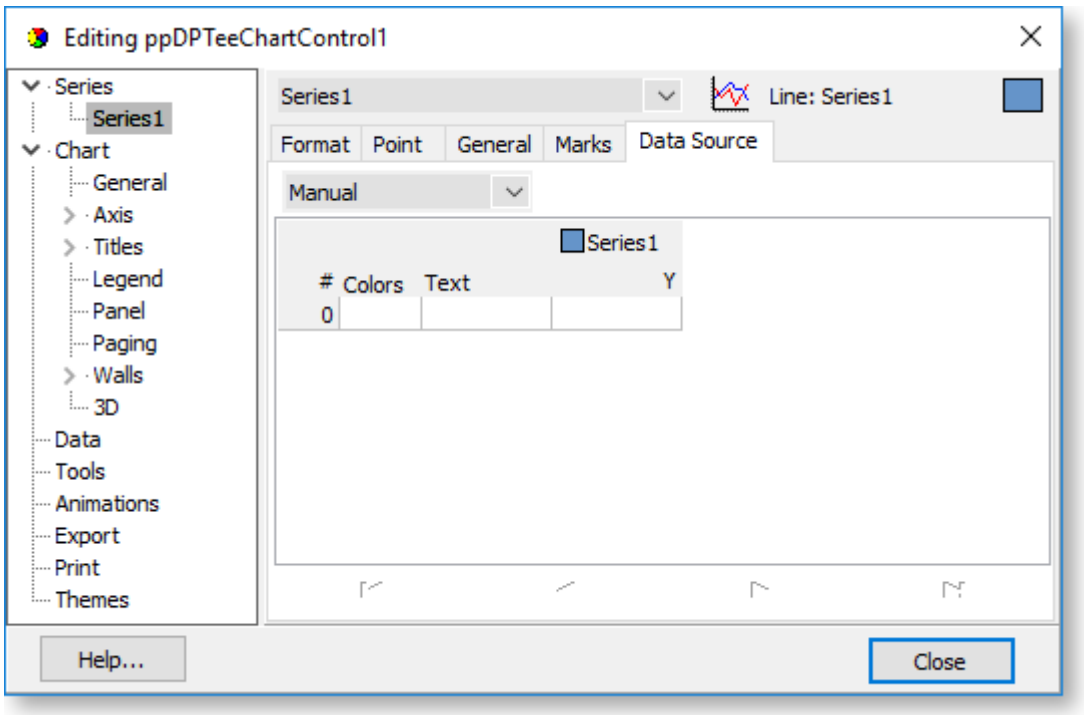
To be able to display any data on the chart, you must set up one or more data series. Click **Series** in the tree on the left of the window and click **Add**. A gallery of available chart types appears - select a type and click **OK** to create the chart series.



You are returned to the chart editing window, and the new series is available in the tree on the left. Select the series to edit its properties:



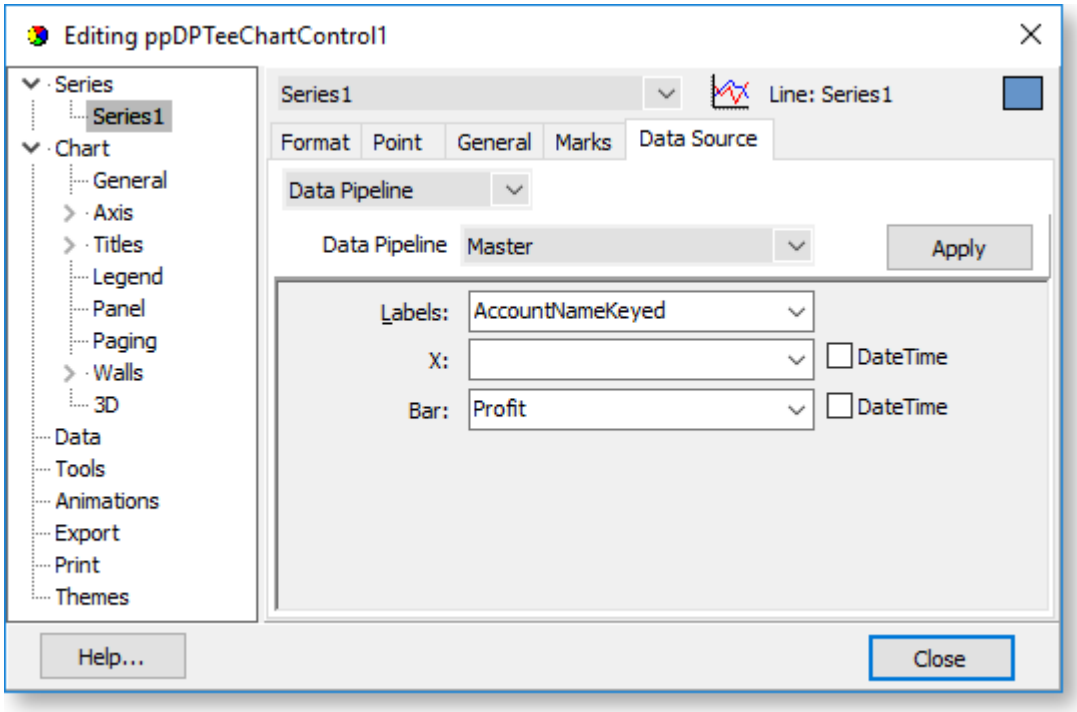
Properties for the series are divided into tabs. Most tabs control the appearance of the chart - to set up the data that appears on the chart, go to the Data Source tab:



Select the data source from the dropdown at the top of this tab. The following options are available:

- **Manual** - Enter data to display on the chart manually.
- **Random** - Generate random data for the chart. This can be used for demonstration purposes.
- **Series** - Not currently used.
- **Function** - Not currently used.
- **Data Pipeline** - This option is available for DBChart components. It lets you select a data source from the report
- **Excel** - Load data for the chart form an Excel file.

In the majority of cases, you will want to select the "Data Pipeline" option. When this option is selected, the lower part of the tab contains controls that let you select and set up the data source:

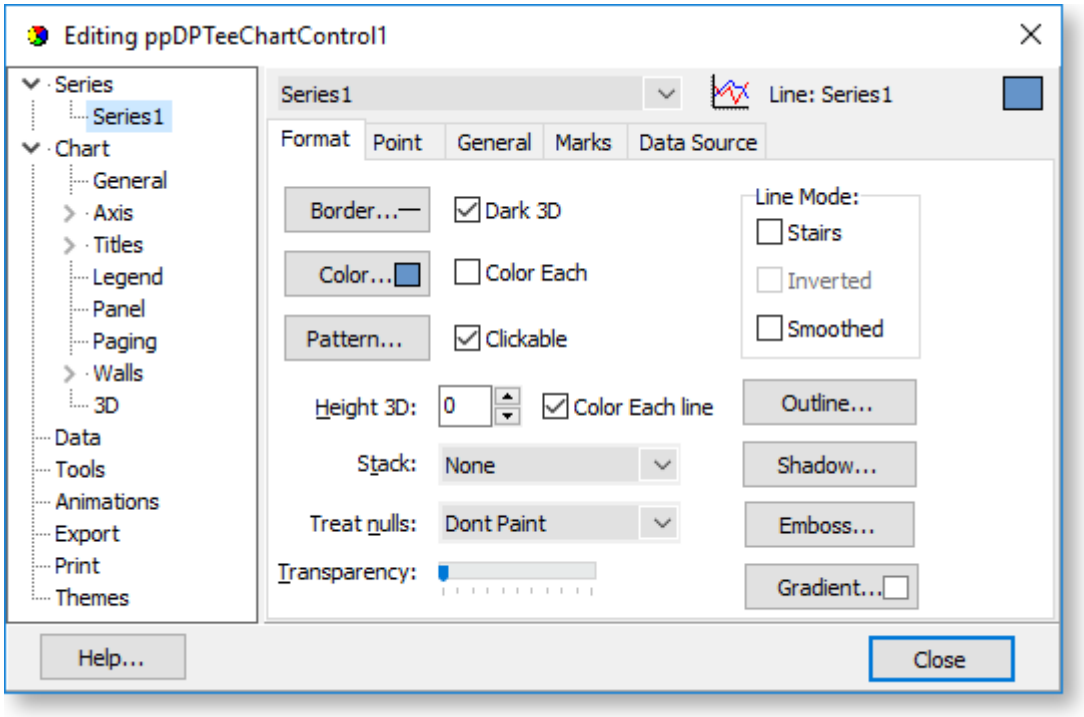


The **Data Pipeline** dropdown contains any data sources set up on the Data Tab, as well as other data sources that are built into the report, such as global variables and user-created report parameters. Select the data source to use, and the dropdowns below are populated with the fields, variables or parameters from the data source. Select which field should be used for which chart property. (The properties available on this tab differ depending on the kind of chart. The screenshot above shows the properties for a bar chart.)

Once the chart's data source is set up, you can use the other areas of the editing window to set up the chart's other properties.

Chart Properties

After you have set up a data series for a chart, you can edit the chart's other properties. Chart properties are divided into sections, which appear in the tree on the left of the window:



- General
- Axis
- Titles
- Legend
- Panel
- Paging
- Walls
- 3D
- Data
- Tools
- Animations
- Export
- Print
- Themes

Note: The property pages that appear on the right of the window as you click items in the tree layout will dynamically re-organise depending on what you've selected. For example, if you select the "Chart" node, you'll see tabs for General, Axis, Titles and so on, with all their respective property pages underneath each one. If you select "Axis", you'll only see the property pages under Axis, and so on.

General

This section contains options to allow scrolling or zooming on interactive charts (in preview mode or on the Dashboard) and set fonts for each of the text fields that make up the chart.

To turn zooming on or off, tick or clear the **Allow** option on the Zoom sub-tab.

To configure scrolling, select an option under **Allow Scroll** on the Scroll sub-tab.

Axis

This section contains options that affect the chart's six axes (Left, Right, Top, Bottom and two Depth axes for 3D charts). For each axis, you can set its scales, min/max values, ticks, labels, position and so on. Notable axis properties include:

- The **Inverted** option on the Options sub-tab can be ticked to invert the axis and all property point along it.

- The properties on the Minimum, Maximum and Inverted sub-tabs let you change the automatic min/max and increment calculations, which will force the chart to use the values you select.
- The Title sub-tab lets you set and format the titles for each axis.

Titles

This section lets you set and format for the chart's title, sub-title, sub-footer and footer text.

Legend

This section lets you determine whether or not to display a legend on the chart, and to format the legend if one is displayed.

To turn the legend on or off, tick or clear the **Visible** option on the Style sub-tab.

Panel

This section lets you set up the background of the chart, which can contain a solid colour, a gradient or an image.

Paging

This section controls multi-page charts, which are not common in Exo Clarity reports.

Walls

This section lets you set up the chart's walls, which are the planes that makes up the sides of a 3D chart. In the default 3D chart, there are three visible walls: the X axis wall, the Y axis wall, and the back wall. If you add a right axis then another wall will be created for that axis. Walls help give 3D charts their depth, but 2D charts also use the back wall.

3D

If you are using a 3D chart, this section contains settings that let you change its 3D properties including zoom, rotation, elevation and perspective.

Data

For charts that are not bound to a data source, this section displays the data that has been entered for the chart. You can edit the data here or on the Data Source tab of the Series section.

Tools

This section lets you add advanced reporting tools to the chart.

Animations

This section lets you add animations to the chart.

Export

For charts that are not bound to a data source, this section lets you export data directly from the chart editing window. You can export the chart as a bitmap image, a .WMF metafile or an .EMF enhanced metafile.

Print

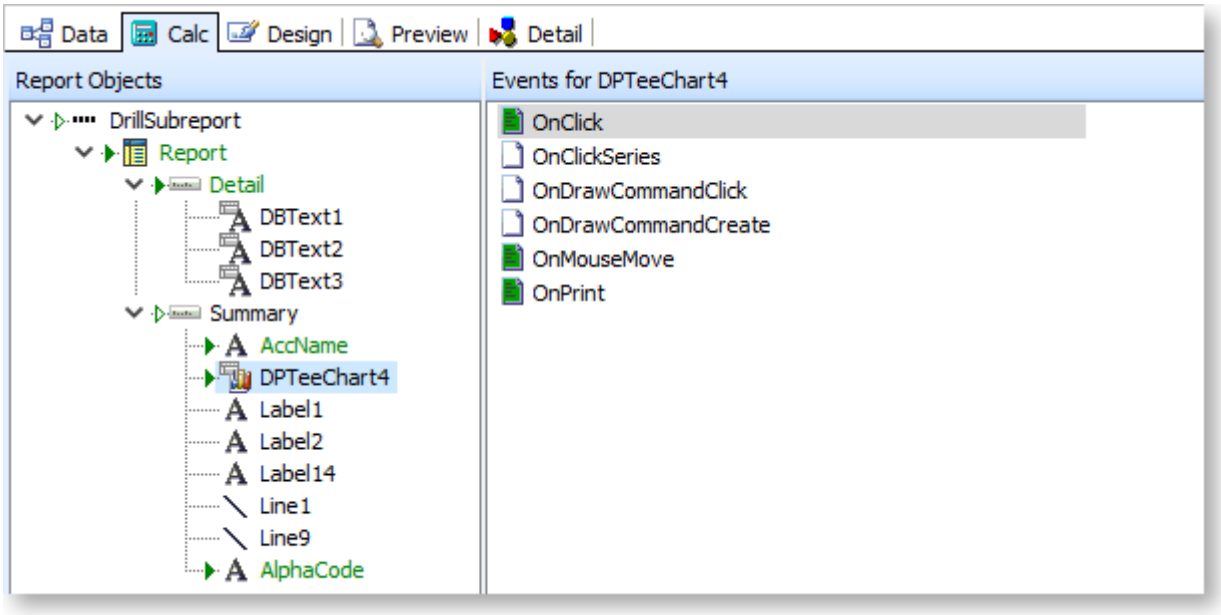
For charts that are not bound to a data source, this section lets you print the chart directly from the chart editing window.

Themes

This section lets you set a visual theme for the chart.

Chart Events

The following events may be useful when programming reports that include a chart. To enter code for a chart event, go to the Calc tab, ensure that the tree view is in Events mode and select the chart component from the tree. All available events for the chart appear on the right:



The OnClick, OnClickSeries, OnMouseMove and OnPrint events are specific to charts - these are discussed below.

OnClick Event

The OnClick event is useful if you just want to perform some action if a click is made anywhere on the chart component. This is particularly useful for line charts - you can add OnClickSeries events for a line chart, but it can be difficult for someone to move the mouse in the right place to actually click on the line. In this case, clicking anywhere on the chart may suffice (not if you want a particular data point though).

This event is used in the "Top" dashboards, e.g. **DashTopCustByMargin.CLR**, where if you are drilled into an item clicking on the line chart of the history will "drill" back up to the original view. It doesn't matter where on the line chart you click, the same event is fired.

Here is the procedure definition for an OnClick event handler:

```
procedure TeeChart1OnClick(ppCustomTeeChart: TppCustomTeeChart); begin
end;
```

Notice that a parameter is passed to the event handler which references the chart that was clicked. At this stage there is not much you can do with that parameter, but more functionality may be added in the future.

OnClickSeries Event

The OnClickSeries event is used whenever you want to perform a specific action when someone clicks on a data series element. Examples of data series elements are series lines in a line chart, bars in a bar chart, pie segments in a pie chart, and so on.

When an OnClickSeries event is fired, the event handler is passed a reference to the element which was clicked on. Here is the procedure definition for the event handler:

```
procedure TeeChart1OnClickSeries(SeriesIndex: Integer; ValueIndex:
Integer; Button: TMouseButton; X: Integer; Y: Integer);
begin
end;
```

Here we are passed some more useful data:

The SeriesIndex value is the index of the series that was clicked (remember you can have multiple series in your chart). The first series is 0, the second is 1, and so on.

The ValueIndex value is the index of the value in the series that was clicked. While this may not mean much if you don't know what data is in your data source, there are some ways that this can be useful. In many of the standard dashboards, a description is stored in a list that is later referenced using this value.

The Button value is a reference to which mouse button was clicked. It is outside the scope of this document. For the purposes of this explanation, all mouse clicks will fire the event (left, middle and right).

The X and Y values give the coordinates where the click was made. This will likely be of little value.

OnMouseMove Event

The OnMouseMove event is used to trigger changes in the cursor. All the bar charts that use OnClickSeries events also use an OnMouseMove event to make the pointer change to a link pointer when one of the bars is hovered over. There are not many other uses for this event at this time. Here is the code snippet:

```
procedure DPTeeChart1OnMouseMove(X: Integer; Y: Integer);
var
 ClickedPart:TExoClickedPart;

begin
 ClickedPart:=DPTeeChart1.CalcClickedPart(X, Y);
 if ClickedPart.Part=cpSeries then
   DPTeeChart1.Cursor := -21
 else
 DPTeeChart1.Cursor := 0;
end;
```

Notice the use of the function DPTeeChart1.CalcClickedPart(X, Y) to return a value which indicates which part of the chart was "clicked" (a "mouse move" event is internally handled much like a click event – hundreds of little events fire every time you move your mouse). The cursor is then changed to -21 if it is part of the Series (i.e. a bar , pie slice, etc.). The value -21 represents the link pointer. The value 0 represents the regular arrow mouse pointer.

CrossTab Reports

The CrossTab component can be added to an Exo Clarity report to display grouped or aggregated data data in a spreadsheet-style format. CrossTab stands for Cross tabulation, a process by which totals and other calculations are performed based on common values found in a set of data (this style of spreadsheet is also called a pivot table).

> **Note:** The General Ledger Account Summary report (**GLaccSummaryCrossTab.clr**) is an example of a report that uses a CrossTab to organise data.

How CrossTabs Work

Assume you have a set of data that describes the sales for a company. Each sale is represented by a row of data. Each row of data contains a customer name, company type, geographical area, sale date or period, and sale amount. If you want to know the total sales for each month by area, you could present the data:

Area: Sydney

Year: 2017

Total Sales: $2577

State: Auckland

Year: 2001

Total Sales: $3548

This format shows the data you need, but it makes a state-to-state comparison difficult. Another format could be:

	Region		
Year	**Auckland**	**Sydney**	**Total**
2016	$7816	$5327	$13,143
2017	$10,500	$9750	$20,250

This format is easier to read and more compact - there is more information in less space. It is easy to make state-to-state and year-to-year comparisons. This format is a CrossTab. The Year and Region values are called "dimensions" because they orient the data in rows and columns. The values in the cells are the calculations created when the sales data is summarized and are sometimes referred to as measures.

The simple CrossTab above be taken a couple of steps further to create a very informative report. For example, if we wanted to know the sales by Customer Type within the region, as well as the total number of distinct sales per Customer Type, we could present this information by adding a Customer Type dimension to the columns and another calculation to the values (count of sales). The resulting CrossTab would look like the one below:

		Region									
		Auckland					Sydney				
Year	Data	Dairies	Super-markets	Mini-marts	Take-aways	Region Total	Dairies	Super-markets	Mini-marts	Take-aways	Region Total
2016	Count Sales	43	27	19	103	192	23	24	12	83	142
	Sum of Sales	$2076	$1764	$1524	$1672	$7816	$1001	$1502	$1423	$1401	$5327
2017	Count Sales	41	56	14	164	275	30	36	10	102	178
	Sum of Sales	$3084	$4500	$1029	$2097	$10,500	$2124	$4500	$1029	$2097	$9750

Notice the new subtotal columns for each Region. This CrossTab shows all of the information of the initial CrossTab, plus more detailed information by Customer Type. You can see that CrossTabs can express a lot of information in a very small amount of space.

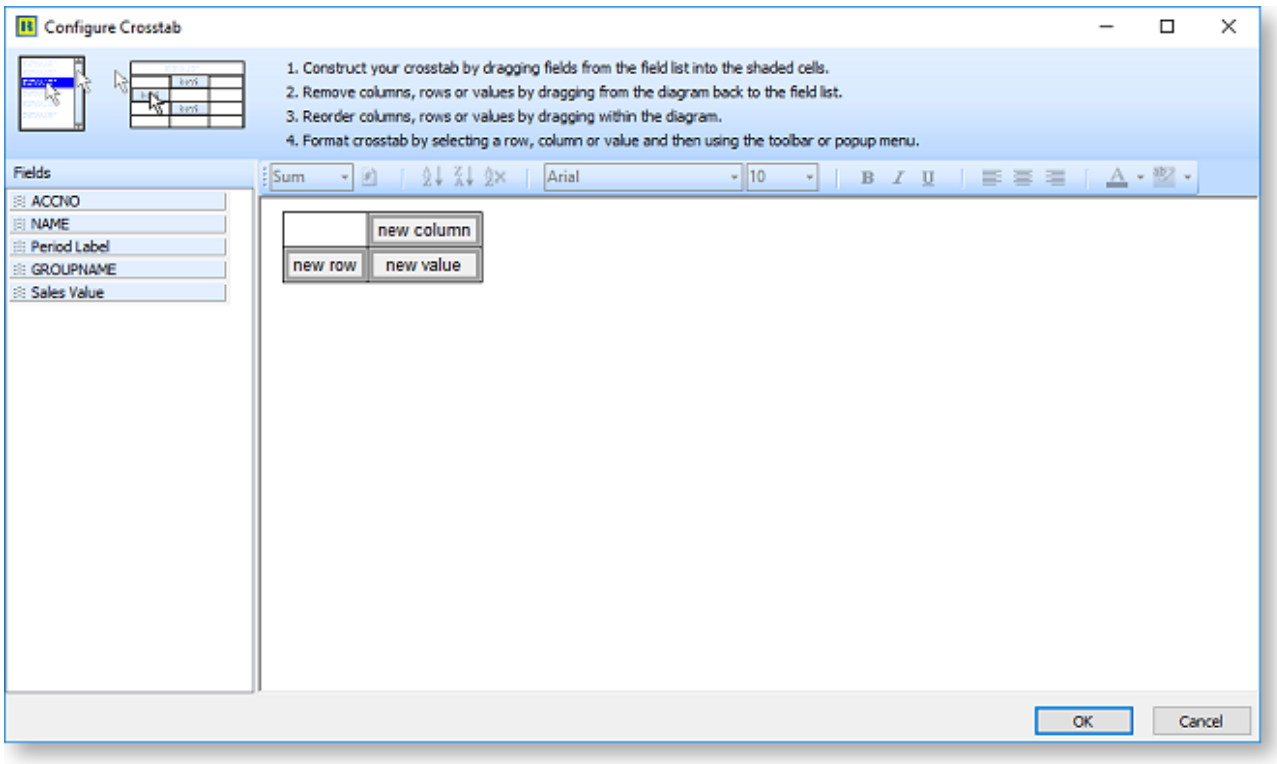
Adding a CrossTab to a Report

Before adding a CrossTab to a report, you must set up a suitable data source for it on the Data tab.

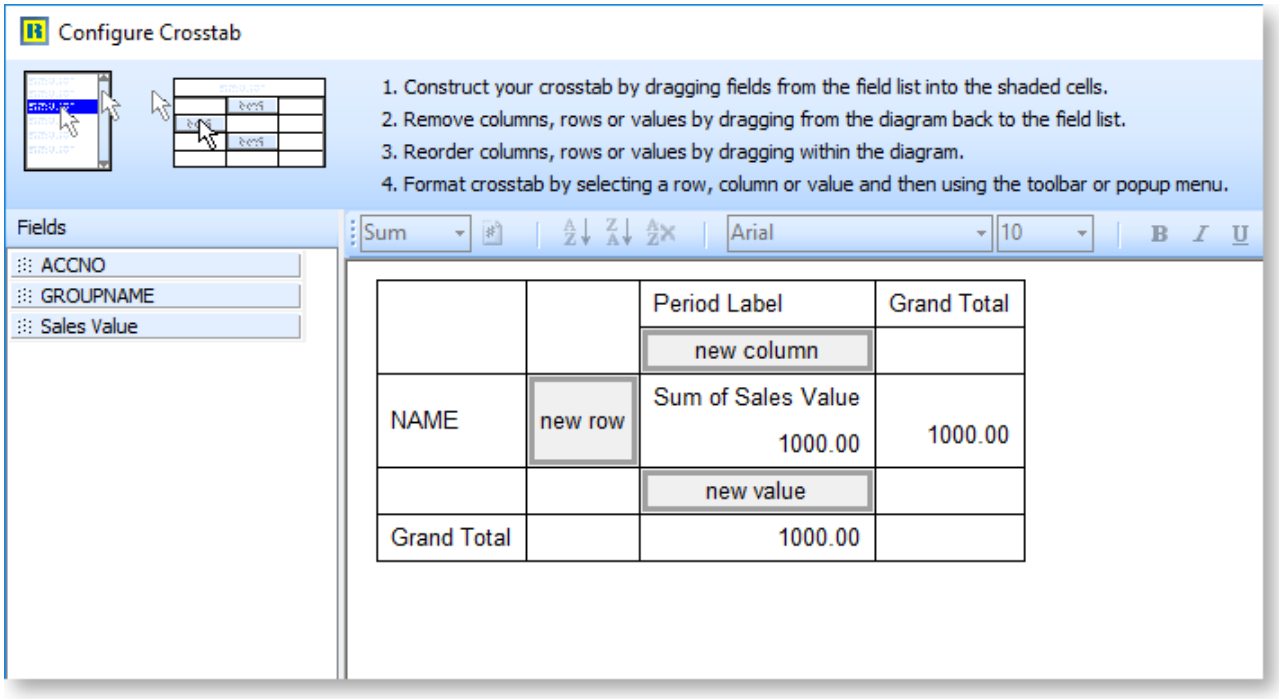
> **Note:** When setting up a data source for a CrossTab, you should be careful to filter out unnecessary transactions. On a large database, calculating a CrossTab could take a long time, and it could affect performance if you are running it on the database server. It may be a good idea to perform some small test runs first to analyse the performance of the data query.

To add a CrossTab to a report, go to the Design tab and click the CrossTab button () on the Advanced Components toolbar. Click on the main report canvas to add the CrossTab component at that position.

To edit the CrossTab, first use the Edit toolbar to select a data source for the CrossTab, then right-click on the CrossTab component and select **Configure**. The Configure Crosstab window appears:



All fields from the CrossTab's data source are available on the left of the window. Drag fields onto the **new column**, **new row** and **new value** controls on the main area of the window to set up the data:

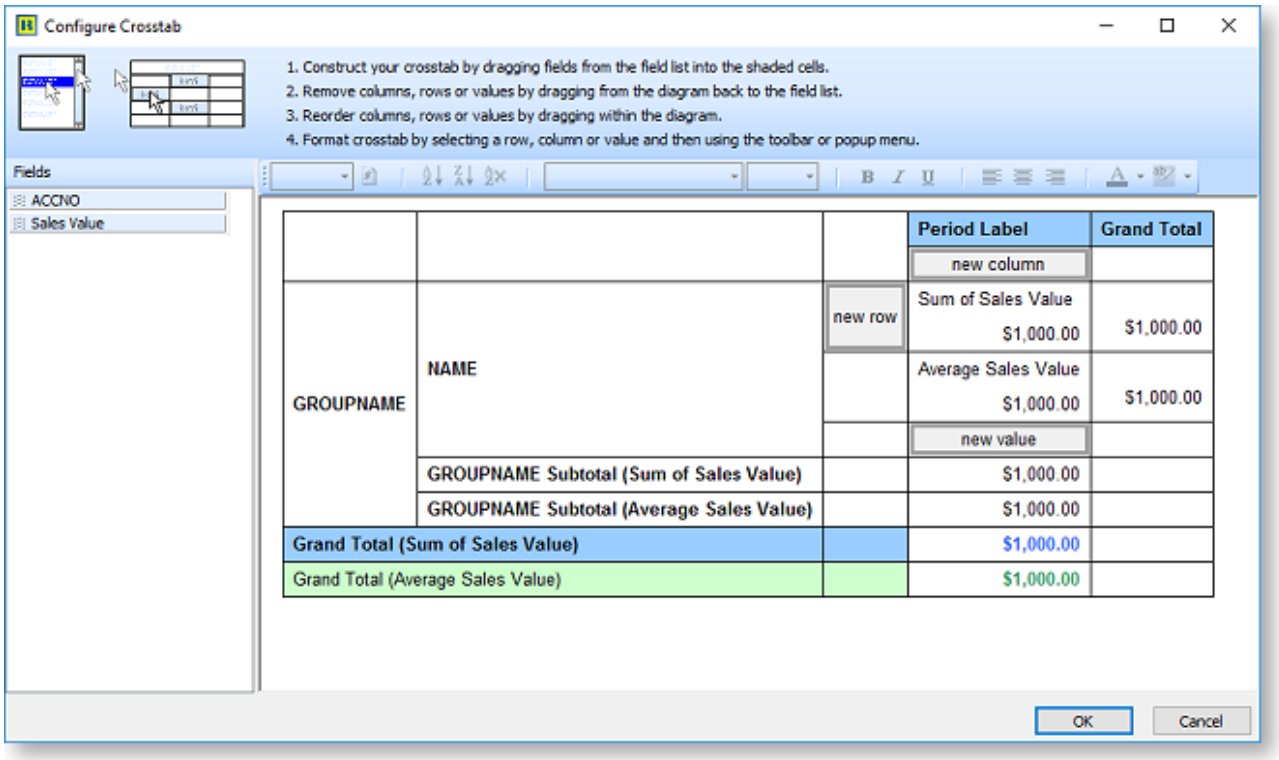


To remove a field from the CrossTab, drag it from the main area back to the list on the left.

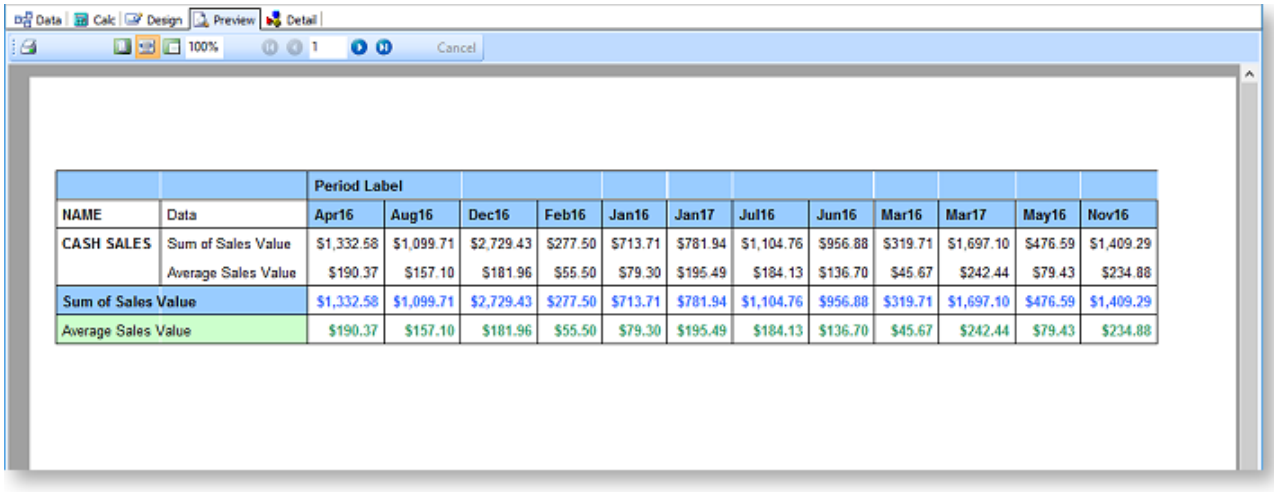
You can add multiple rows, columns and values to the CrossTab and reorder them by dragging them into different positions - by experimenting with different data layouts, you can see how powerful and flexible the CrossTab representation can be.

Note: When a field is dragged onto a row or column, it is removed from the list on the left; however when a field is dragged onto a value, it remains in the list. This means you can add the same field as a value more than once, for example if you want to show the same value summed and also averaged.

Once you have the data fields in place, you can use the toolbar buttons and right-click options on the Configure Crosstab window to set up the CrossTab's appearance and behaviour.



When you have finished configuring the CrossTab, click **OK** to return to the main report. You can now see how the CrossTab will look on the Preview tab:

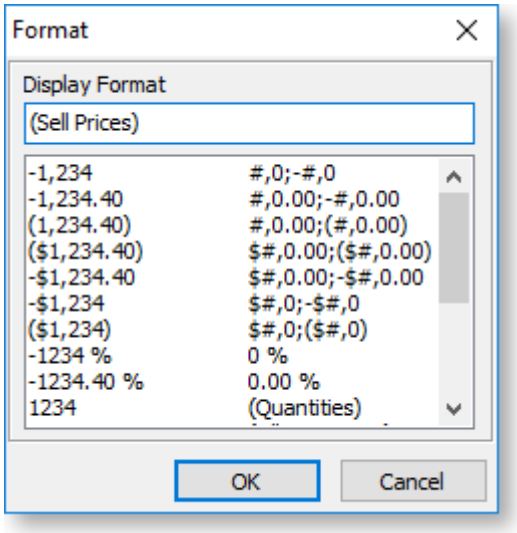


		Period Label											
NAME	Data	Apr16	Aug16	Dec16	Feb16	Jan16	Jan17	Jul16	Jun16	Mar16	Mar17	May16	Nov16
CASH SALES	Sum of Sales Value	$1,332.58	$1,099.71	$2,729.43	$277.50	$713.71	$781.94	$1,104.76	$956.88	$319.71	$1,697.10	$476.59	$1,409.29
	Average Sales Value	$190.37	$157.10	$181.96	$55.50	$79.30	$195.49	$184.13	$136.70	$45.67	$242.44	$79.43	$234.88
Sum of Sales Value		$1,332.58	$1,099.71	$2,729.43	$277.50	$713.71	$781.94	$1,104.76	$956.88	$319.71	$1,697.10	$476.59	$1,409.29
Average Sales Value		$190.37	$157.10	$181.96	$55.50	$79.30	$195.49	$184.13	$136.70	$45.67	$242.44	$79.43	$234.88

CrossTab Toolbar Buttons

The tools on the Configure Crosstab window's toolbar let you change the behaviour appearance of the rows, columns and value elements on the CrossTab.

Button	Name	Description
Sum	**Calc Type**	This tool is available when the title of a value element is selected, e.g. "Sum of Sales Value" in the example above. It allows you to select the calculation that will be performed on the value, which can be one of: • Count • Sum • Minium • Maximum • Average Selecting a new calculation updates the value's title, e.g. if "Average" is selected, the title will becomes "Average of Sales Value".
	Display Format	Opens a window where you can select how the data should be displayed. The available options depend on the type of data, e.g. for currency data, there will be options for currency formats, for date/time data, there will be options for date formats, etc.



	Sort Asc **Sort Desc** **No Sort**	These tools are available when a row or column element is selected. They determine how the data in the row/column will be sorted: in ascending order, in descending order, or no sorting, i.e. the data will displayed in the order it appears in the data source.
Arial	**Font**	Select the font face for the selected element.
9	**Font Size**	Select the font size for the selected element.
B	**Bold**	Sets all text in an element to **bold**.
I	**Italic**	Sets all text in an element to *italic*.
U	**Underline**	Sets all text in an element to underline.
	Left Justify	Aligns text to the left side of the element's bounding box.
	Center	Aligns text to the right side of the element's bounding box.
	Right Justify	Centers text relative to the element's bounding box.
	Font Colour	Sets the colour of a element's text.
	Highlight Colour	Sets the background colour that will appear behind the element's text.

Right-click Options

You can right-click anywhere on the main area of the CrossTab to access the following options, which apply to the currently selected element of the CrossTab. Some options are only available for some elements.

Option	Description
Tips	Turns the tips at the top of Configure Crosstab window on or off.
Font	Opens a window where you can set the font face, size, style and colour for the selected element.

Colour	Opens a colour picker window where you can set the background colour of the selected element.
Display Format	Opens a window where you can select how the data should be displayed. The available options depend on the type of data, e.g. for currency data, there will be options for currency formats, for date/time data, there will be options for date formats, etc.
Skip Nulls	Select whether or not to display null values for the selected element.
Null Order	If **Skip Nulls** is off, select whether null values should appear first or last in the data.
Visible	Select whether or not to display the element when the CrossTab is generated on the main report.
Gridlines	Select whether or not to display gridlines when the CrossTab is generated on the main report.

CrossTab Settings on the Report

Options that affect how the CrossTab appears when the main report is generated are available from the right-click menu on the Design tab.

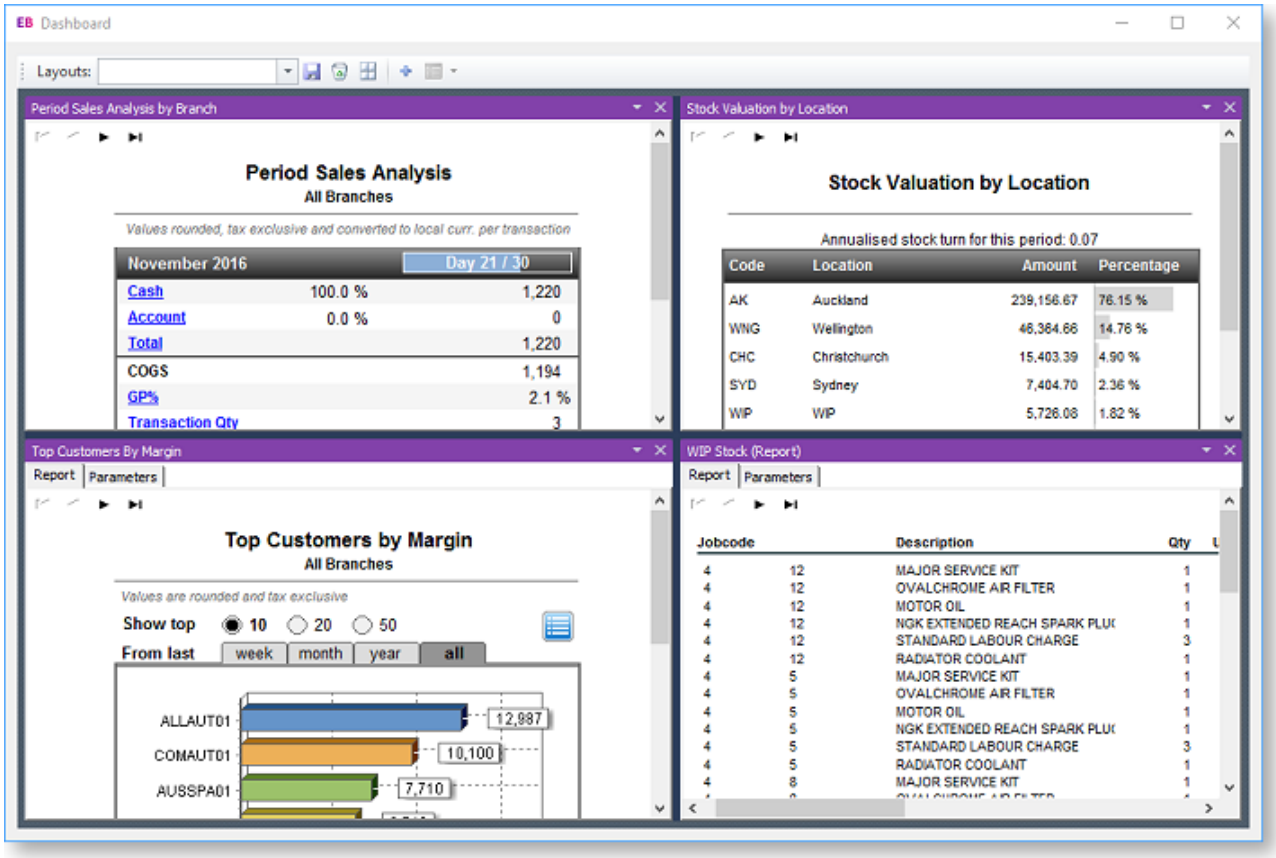
Right-click on a CrossTab component and select **Pagination** to control the order in which the pages print when the Crosstab cannot fit on a single page: Across then Down or Down then Across.

Right-click on a CrossTab component and select **Style** to choose how report captions (row or column headings) should appear when the Cross Tab cannot fit on a single page:

- Standard: Captions appear on the first page only.
- Repeated Captions: Captions appear on every page.

Dashboards

Dashboards are customisable displays showing one or more "widgets". Widgets are small, self-contained information panes; in many cases they are summary reports, making the Dashboard an at-a-glance overview of the entire MYOB Exo Business system.



Dashboard widgets can display several kinds of content, including web pages and data from an SQL query, but most commonly they display an Exo Clarity report.

To display a report on a dashboard, create the report in Exo Clarity as normal, then create a dashboard widget to contain it in Exo Business Config.

Creating a Report for a Dashboard Widget

Exo Clarity reports that appear in dashboard widgets are created and set up in the same way as any other report; however, several points specific to dashboard widget should be taken into consideration:

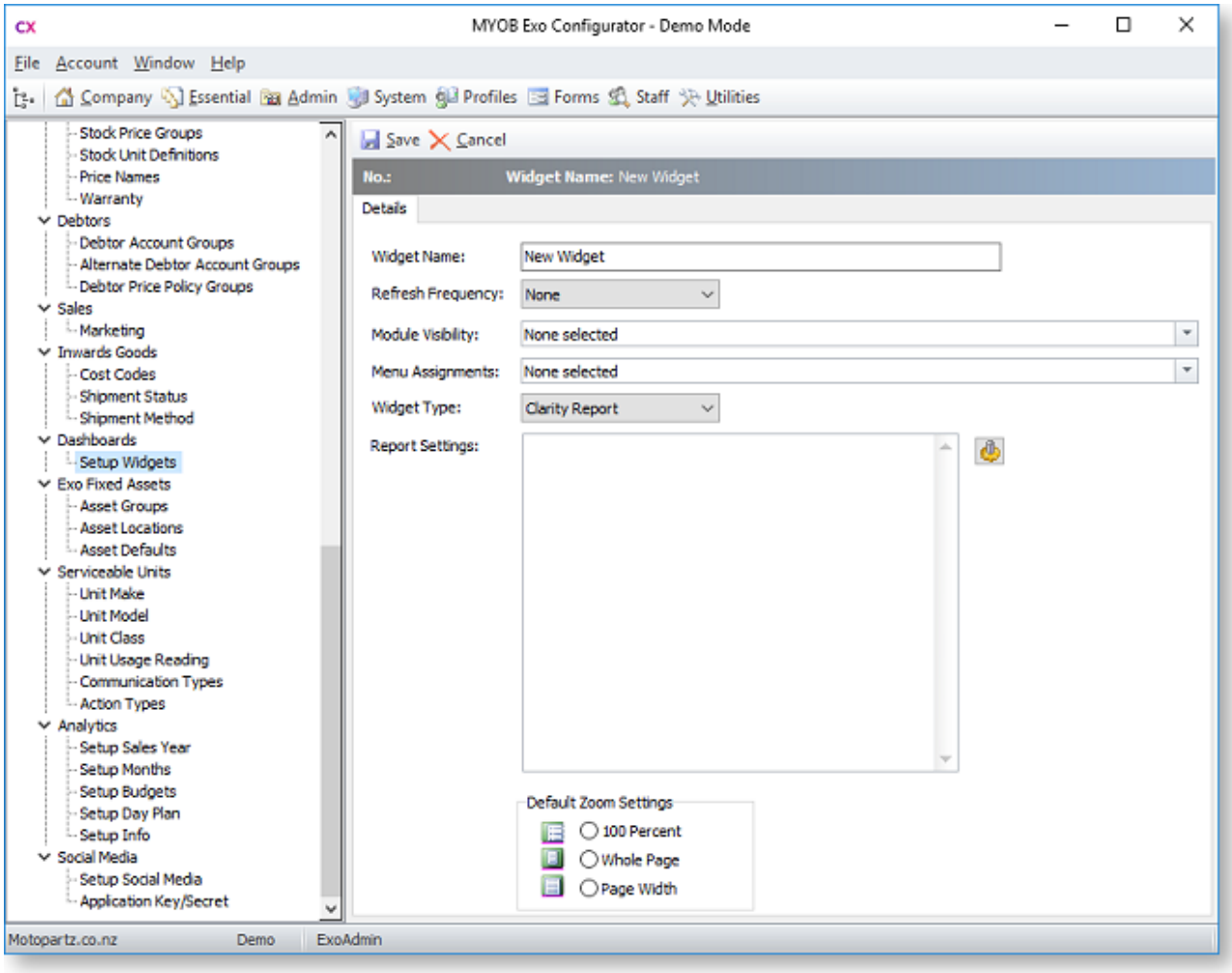
- Dashboard widgets are displayed in a relatively small frame - make sure that the report is not too large; otherwise it will either not fit in the widget pane without scrolling, or will need to be zoomed out to the point that the text becomes hard to read.
- Dashboard widgets generally display summary data - space constraints mean that detailed reports are usually not suitable for a dashboard widget.
- Dashboard widgets are intended to provide information at a glance - large text, use of colour and, depending on the data, use of charts is recommended.
- Report parameters can be set on a separate Parameters tab in the widget. If it is necessary for users to set parameters before the report is displayed, use the /A=Y runtime parameter to make the Parameters tab appear by default when the widget first opens.

The standard dashboard reports supplied with Exo Business should give you some ideas for how to set up your own reports for use in dashboards.

Setting up a Dashboard Widget

Once a report has been created and saved, you must create the dashboard widget that will contain it, using Exo Business Config. Widgets are set up in the Admin section at **Dashboards > Setup Widgets**.

This section displays all widgets that currently exist in the system. Click **New** to create a new widget.



To set up the widget:

1. Enter a name for the widget.
2. Select the **Refresh Frequency**, which determines how often the data in report is automatically refreshed.
3. Select which Exo Business module(s) this widget should be available for. Choose from:
    - <All>
    - Exo Clarity
    - Exo Job Costing
    - Exo Business Analytics
    - Exo Business CRM
    - Exo Acountant's Assistant
4. Widgets are not available to users until they have been added to their dropdown menus. You can add the widget to users' menus automatically by selecting one or more from the **Menu Assignments** dropdown. Widgets can be manually added to menus later in the Staff > Menus > Dropdown Menu section of Exo Business Config.
5. Select "Clarity Report" for the Widget Type.

Note: Once a widget is saved, its **Widget Type** cannot be changed.

6. Click the  button to select the report file to display. Once a report is selected, this button opens the Clarity Report Parameters window - the parameters you select here become the default parameters when the widget first appears. These defaults appear in the **Report Settings** field. You can also edit the Report Settings field directly to enter any other runtime parameters that may be required.

   Note: Some errors are suppressed in the dashboard, and it could be irritating to have messages or parameter windows popping up each time the dashboard refreshes - when designing dashboard reports, rather than giving an error if the correct parameters are not supplied it is smarter to use some sensible default values and continue running the report.

7. Select **Default Zoom Settings**, which determine how large the report appears inside its frame.
8. Click **Save**.

The widget can then be added to dashboards from the Exo Business module(s) it is available in.

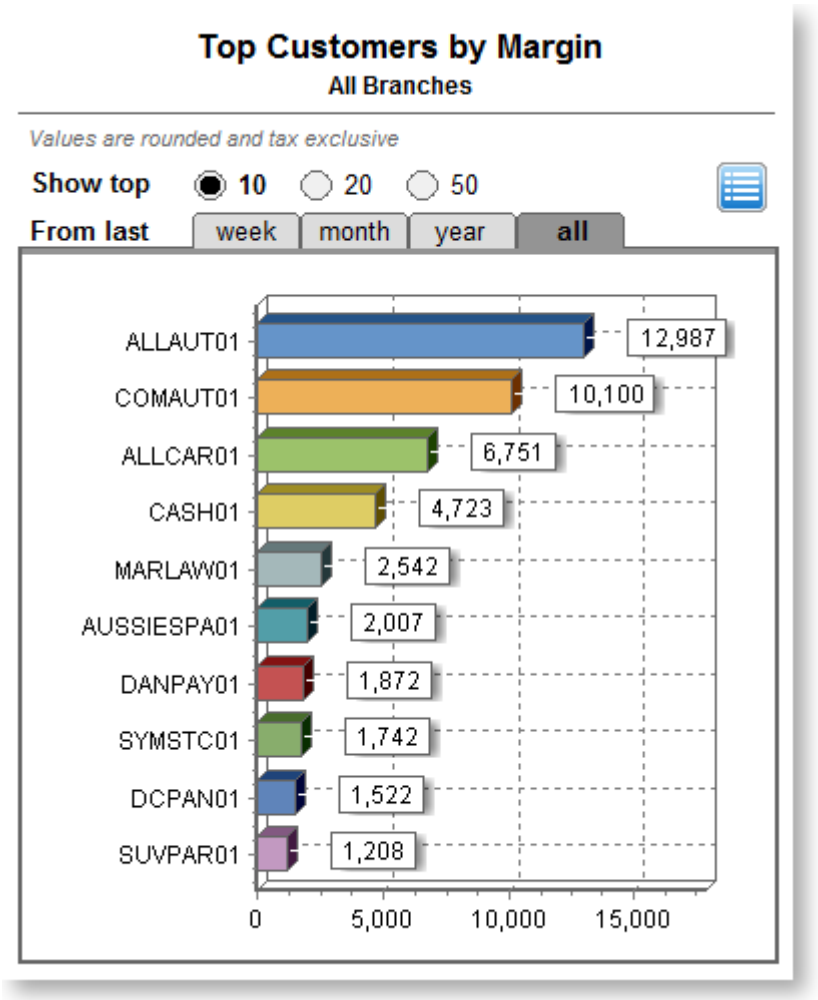
Standard Dashboards

A library of dashboard widgets is included by default with MYOB Exo Business. Widgets are available for the core Exo Business module, as well as several of the additional modules like Exo Job Costing, Exo CRM and the Exo Accountant's Assistant. All dashboard widgets are available in the **Admin > Dashboards > Setup Widgets** section of Exo Business Config (sort this view by **Widget Type** to find the widgets that are based on Exo Clarity reports). Before designing dashboard widgets of your own, it can be useful to browse through these to get ideas for how information can be presented on a dashboard display.

Common Dashboard Features

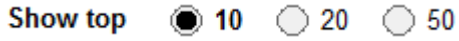
Many of the dashboard widgets supplied with Exo Business include interactive graphical features that let users change the information displayed by the widget or change the way the information is displayed. These features can be a good way of including more information on a small widget - by showing only a subset of the available data but giving users the ability to customise the data that appears, you can make a lot of information available without taking up a lot of space.

For example, the Top Customer by Margin widget (which displays the report **DashTopCustByMargin.CLR**) offers radio buttons and a tabbed interface to control how much data is displayed, and the option of toggling between a graphical and tabular presentation of the data:



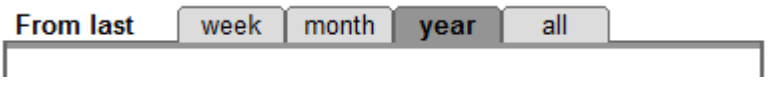
These features and others are discussed below, to give you an idea of how might set up and use them in your own dashboard widgets. Some of them use custom programs created specifically for the Exo Business dashboards to achieve their effects.

Radio Buttons



These are used in the range selection for the "Top" reports, to show for example, the Top 10 Suppliers by Spend. While radio buttons are used extensively in Windows forms and web pages, Exo Clarity does not have a radio button tool, so these were created with circle Shape components, text Label components and click events. When an option is selected, the circle fill colours are changed, the selected option's text is shown in bold, the relevant filter options are changed, and the report is refreshed.

Tabs



These give the appearance of a tabbed sheet, where clicking a tab brings that sheet to the front. This feature is created simply using rounded rectangle Shape components with some text placed on top. The rectangles have an OnDrawCommandClick event handler which changes the colour of the rectangles, sets the new filter value and refreshes the report. The colours are designed to merge into the panel below when selected.

Text/Chart Buttons

Several reports include a button to switch between graphical and tabular views of the data. These buttons are Image components placed in the same position. When clicked, they simply toggle a global variable which stores the current state, then hides what shouldn't display and shows what should.

Progress Meter

While it would be possible to implement a progress bar like this with a Chart, it can be done easily with simple graphical components. This feature was implemented as a few rectangle Shape components and a text Label component. The length of the bar is determined as a percentage of the total length of the meter, based on the length of the period and the current day of that period.

Charts

The standard dashboard widgets make extensive use of Chart components to show data in a compact format that shows information at a glance. Many dashboard views use more than one chart, and most of them use more than one data series in each chart, showing and hiding the series as required.

Custom Dashboard Programs

Some of the standard Exo Business dashboard widgets include custom programs that could be useful in your widgets. Open the .CLR file for a widget in Exo Clarity to see the code for these programs and how they're used in the report.

DrillDown(ValueIndex)

This procedure was designed to be called when a chart is clicked on, with the index of the clicked value. While different reports work slightly differently, the procedure basically re-filters a drill-down pipeline, sets some variables, then shows and hides various parts of the report to actually make the drilldown happen. Finally the page is resized, as the drilldown view is rarely the same height as the original view.

ResizePage

This procedure does all the calculations to work out how high a report page should be, sets the height, then refreshes the report. This procedure differs somewhat between a few of the reports, because each report has a different page layout that requires a different height calculation.

SetDateFilterParm(Period, Refresh)

This procedure (designed for "Top X" reports) takes two parameters: a string of the period (Week, Month, Year or All) and a parameter stating whether the report should refresh after setting the filter parameters or not.

This calculates the "From" date to apply as the TRANSDATE field filter based on the selected period and the current date. The clicked "tab" is marked as selected, then the AutoSearch criteria is then set and the report is optionally refreshed. The optional refresh flag is in place to avoid unnecessary report refreshes which slow down the report significantly.

SetRecordCount(Count, Refresh)

Similar to above, but it sets the record count, edits the radio buttons and optionally refreshes the report.

SwitchBack

This procedure performs the reverse of the Drilldown procedure, hiding what was shown and showing what was hidden. Re-filtering is not necessary. It also takes into account whether the original view was on the chart of table view and shows the appropriate components.

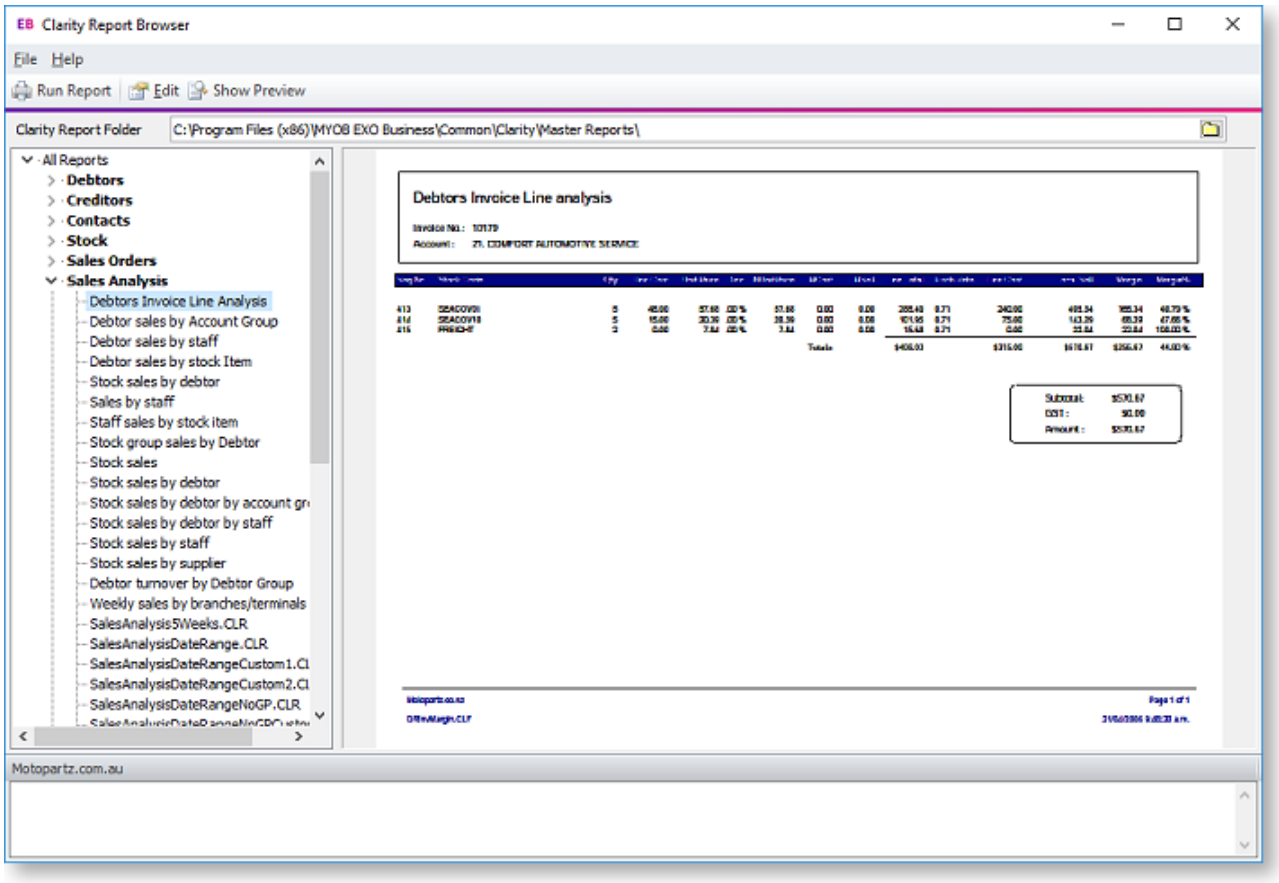
SetProgressBar(Region, Percent, Text)

This procedure sets the bar in the progress indicator to the desired length. You pass in a percentage (0-100) and it will calculate the length for you. If you pass an empty text string it will simply show the percentage as a string.

Running Reports

Clarity Report Browser

The Clarity Report Browser displays a structured list of all available reports. Reports can be run directly from the browser, or they can be opened for editing in the Clarity Report Designer. Open the Report Browser from the main Exo Business application by selecting **Clarity Report Browser** from the Reports menu:



Running and Editing Reports

If the **Show Preview** button is toggled on, clicking on a report displays a review of the report on the right of the window.

Select a report and click **Run Report** to run the report.

Select a report and click **Edit** to open the report for editing or customisation in the Clarity Report Designer.

Sorting Reports

MYOB Exo Clarity looks for Clarity reports and forms in the directories specified by the Computer Profile settings **Directory for Custom Clarity Forms(.CLR,.CLF and .FMT)** and **Directory for Standard Clarity Forms(.CLR,.CLF and .FMT)**. All Reports detected by Exo Clarity are displayed in the browser, sorted into categories.

Information on the report's category is stored in the file properties of each report file. If this information is not present for a report file, the report will be classified into the "No Category" category. Right-click on a report and select **Properties** to view that report's file properties.

Note: Exo Clarity keeps a record of all standard reports that are available by default. Selecting **File > Reclassify All Reports** sorts all known reports that are missing their classification data into the appropriate categories.

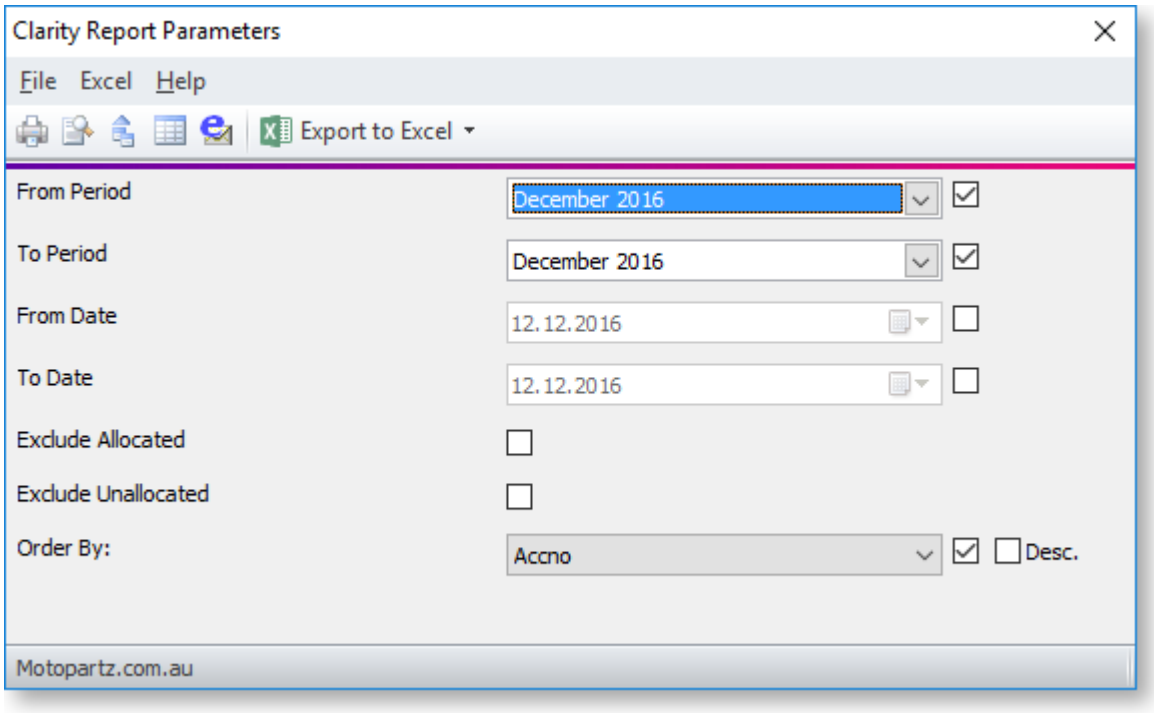
Running a Report

Select a report to run from the Report menu or from the Clarity Report Browser. After running the report, you must enter any report parameters required. You can then output the report to the required destination in the required format.

Specifying Report Parameters

If no parameters are required, the report is generated immediately. A report preview is displayed, which can then be printed.

Many reports use a standard Clarity Report Parameters window for gathering parameters (some or all of which may be optional):



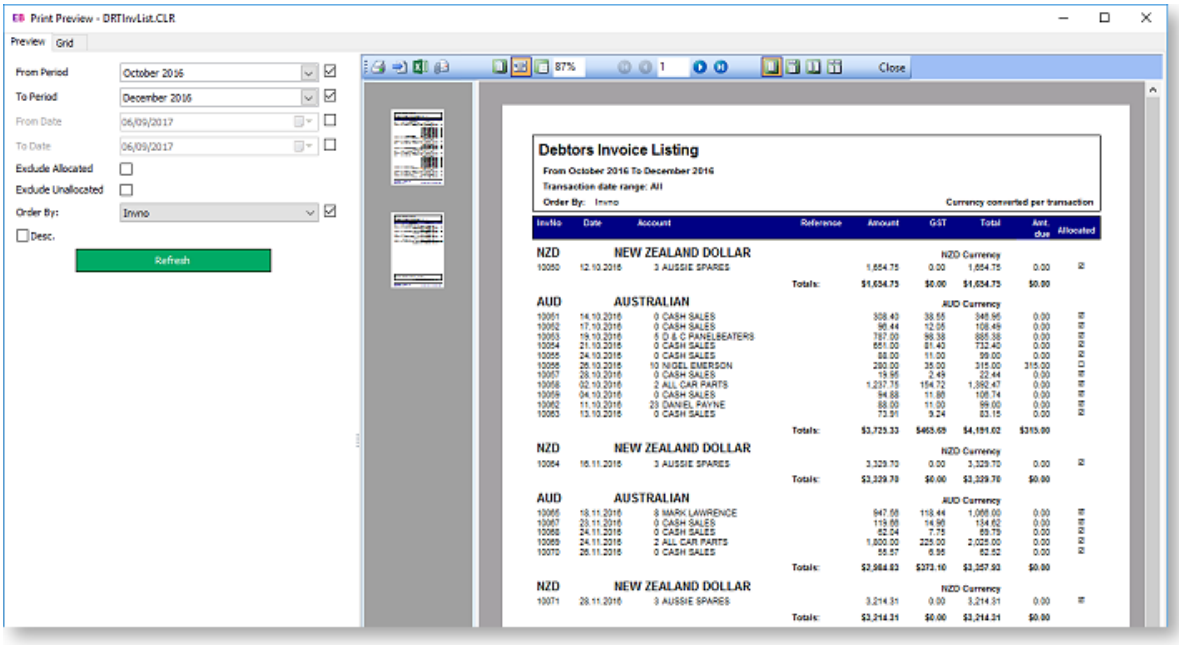
The exact parameters available differ from report to report, but the layout of the Clarity Report Parameters window and the available options are common to all reports that use the window.

Note: Some reports use a more complex custom window for gathering parameters. These windows are specific to their report.

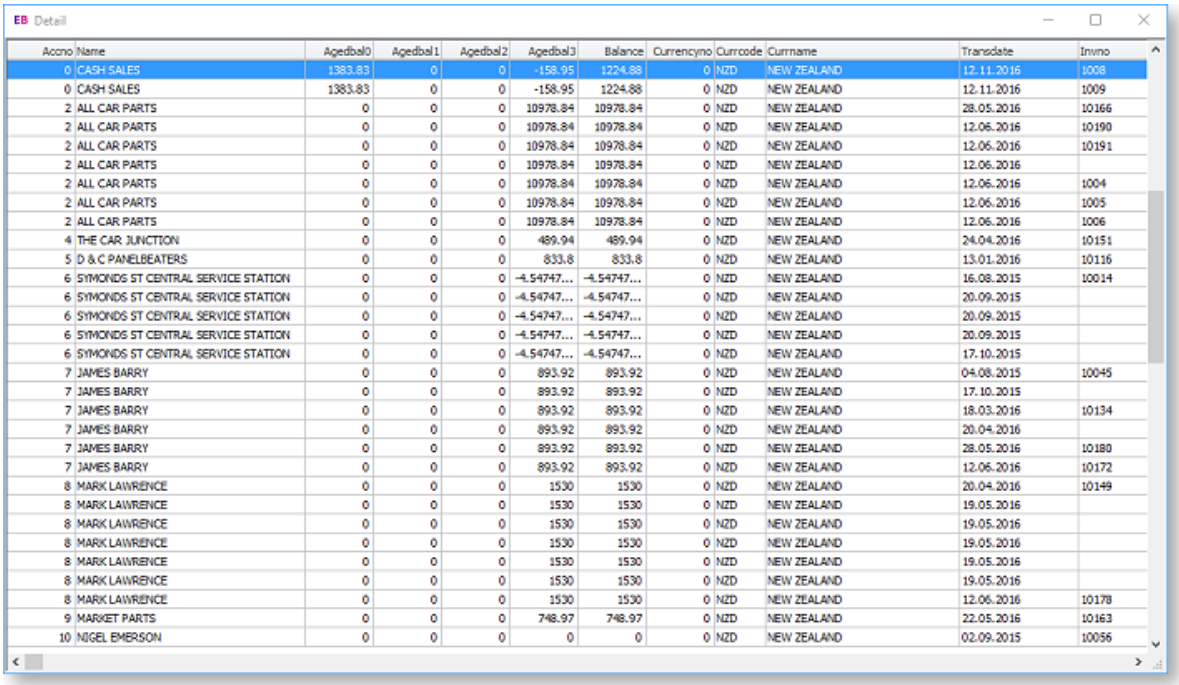
Viewing, Printing and Exporting Reports

Once the required parameters have been entered, the report can be generated and then output to a variety of destinations. The Clarity Report Parameters window offers the following options are available from its File menu or toolbar:

- **Print** - generates the report and prints it to the selected printer.
- **Preview** - Generates the report and displays it in an onscreen viewer. The report can be printed directly from the preview, or you can close the preview to return to the Clarity Report Parameters window.



- **Export to File** - Generates the report, then displays a Print to File window, allowing you to select the file format to export the report to.
- **View Grid** - Displays a grid containing the data that the report is generated from. This view is also available on the Grid tab of the Preview window.



- **Email Attachment** - Generates the report, then attaches it to a new email as a PDF. The default email client is used for composing and sending the email.
- **Export to Excel** - Generates the report as a Microsoft Excel report, then opens it in Excel. This dropdown has three options:
    - **Export to Excel 2007 and later (XLSX)** - uses the XLSXData output format to produce an Excel file in the .XLSX format, with the data laid out in columns for easy data manipulation.
    - **Export to Excel 97 - 2003 (XLS)** - uses the XLSData output format to produce an Excel file in the .XLS format, with the data laid out in columns for easy data manipulation.

- **Export to XLS Report** - uses the XLSReport output format to produce an Excel file in the .XLS format, with the data laid out as it appears on the printed report.

- **Note:** When using Excel 2010, the report will open in Protected View mode. To prevent this from happening, open the Trust Center in Excel and add the MYOB Exo Business temporary directory as a Trusted Location. The temporary directory is specified by the **Directory location for temporary files** Computer-level profile setting.

Exporting to Excel

Clarity reports can be exported as Microsoft Excel files (XLS). Three different output types are available for Excel export:

- **XLSReport**, which presents the report in a pictorial format, with the emphasis on preserving the appearance/layout of the report. To export to this format, select **Export to Excel > Export to XLS Report** on the Clarity Report Parameters window.
- **XLSData**, which presents the report in a strictly columnar format, with the emphasis on ensuring the data is formatted to allow further data manipulation. This format preserves the formatting of the original of the original report, including column widths, colours, text formatting and any data formatting like the number of decimal places or percentage signs. To export to this format, select **Export to Excel > Export Data to Excel 97 – 2003 (XLS)** on the Clarity Report Parameters window.
- **XLSXData**, which functions in the same way as the XLSData format, but produces an .XLSX file, rather than an .XLS file. To export to this format, select **Export to Excel > Export Data to Excel 2007 and later (XLSX)** on the Clarity Report Parameters window.

When exporting to XLSData or XLSXData, Exo Clarity analyses the report and automatically formats it. By setting up the Excel output in Exo Clarity, you can override the automatic process and specify exactly how the output Excel file should be formatted.

Note: Video-based Help is available for this topic on the **Exo Business Education Centre**.

Setting up Excel Output

You can set either of the XLS formats as the default by setting the report's **DefaultFileDeviceType** property to "XLSReport", "XLSXData" or "XLSData". This property is available under **Generation** on the Design tab.

Additional settings that relate to XLS output are available in the **XLSSettings** section under **Output - File**.

XLSReport Settings

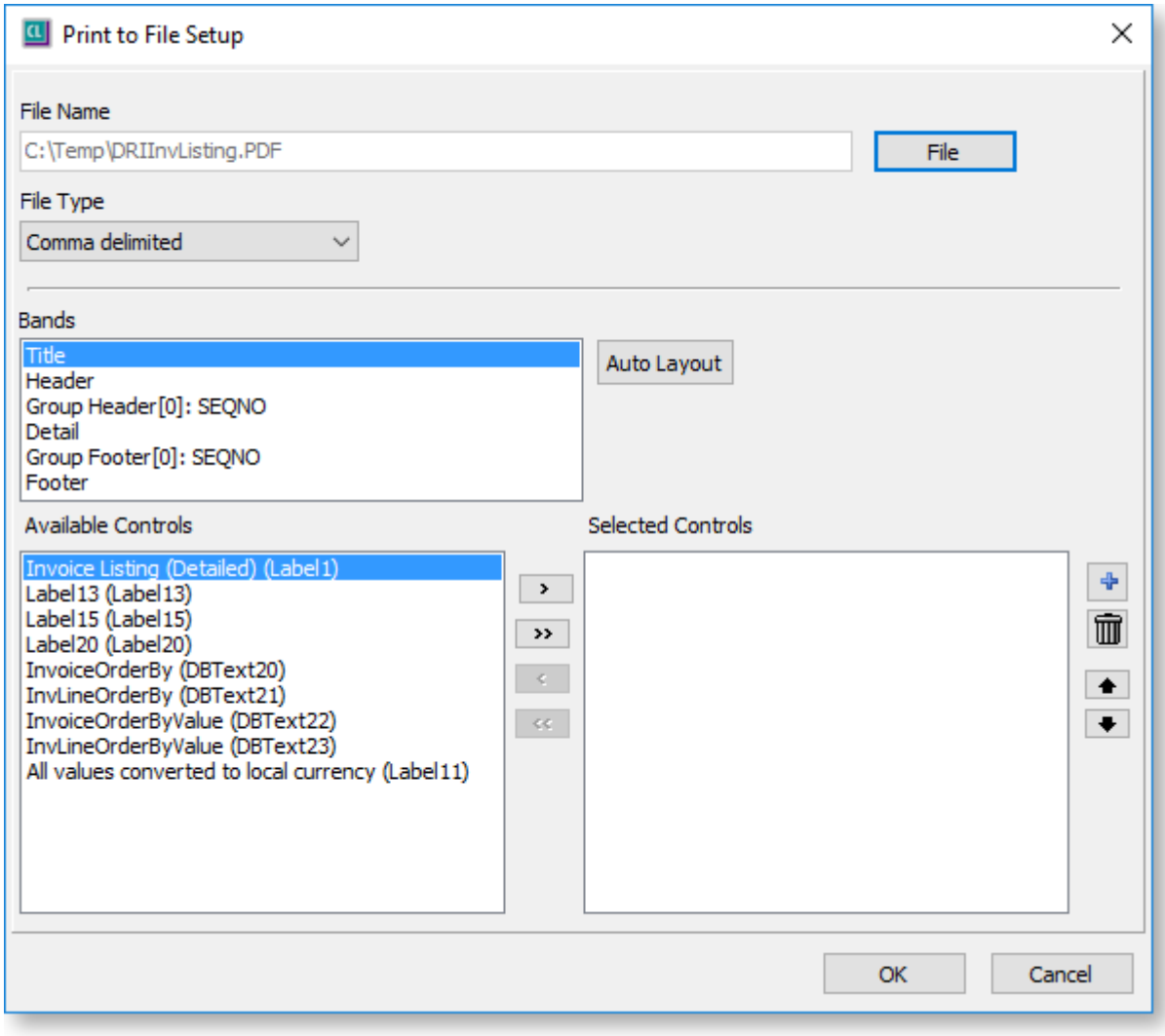
The XLSReport format uses the setup on the Design tab. When printed, the XLS output replicates the original report, including merged cells.

The file type to export to is specified by the **DeviceType** property under **Generation**.

The default filename and location for the exported file are specified by the **TextFileName** property under **Output - File**.

XLSData and XLSXData Settings

The XLSData and XLSXData formats are set up using the Print to File Setup window. This window is opened by selecting **Print to File Setup** from the File menu when on the Design tab.



To specify which columns should appear in the Excel file when the report is exported in the XLSData or XLSXData formats:

1. Select a report band from the **Bands** section. All components that appear in that band appear in the **Available Controls** section below.
2. Select the components that should appear as columns in the Excel file and use the arrow buttons to move them to the **Select Controls** section on the right.
3. Repeat for each band that has columns you want to include in the Excel export.

You can use the **Auto Layout** button to automatically populate the Selected Controls pane. The system analyses the report, then selects and orders columns for optimal output. The report data columns are sorted to the top of the pane (meaning they will appear on the left of the Excel file) and the titles, labels, etc. are sorted to the end. You can then delete any unnecessary columns or re-order them as necessary.

The and buttons let you add and remove blank columns in the Excel output. This can be useful for bands that contain section totals - you can include only those columns that you want to be totalled, and tell the output to skip the other columns by adding blank columns in their place.

For each field included in the export, the **SaveLength** property on the Design tab under **Output - File** can be used to specify that field's column width in characters. If this property is left at the default value of zero, the column will autosize based on the maximum content length. If the property is set to a value that is shorter that the maximum content length, the content will appear truncated in Excel (the content is not actually truncated – it can be viewed by resizing the column in Excel). The minimum (non-zero) value for the property is 8 characters.

GL Reports

When exporting a GL Report to Excel (i.e. when the **Deliver to Excel** option is ticked on the Run GL Report window), if the selected Clarity report's **DefaultFileDeviceType** property is set to "XLSData", the report will be exported to XLSData; otherwise it will be exported in the XLSReport format.

Automatic Formatting Process

Exo Clarity uses an automatic process to determine how Excel files should be formatted when exporting in the XLSData or XSLXData formats. This process is used when:

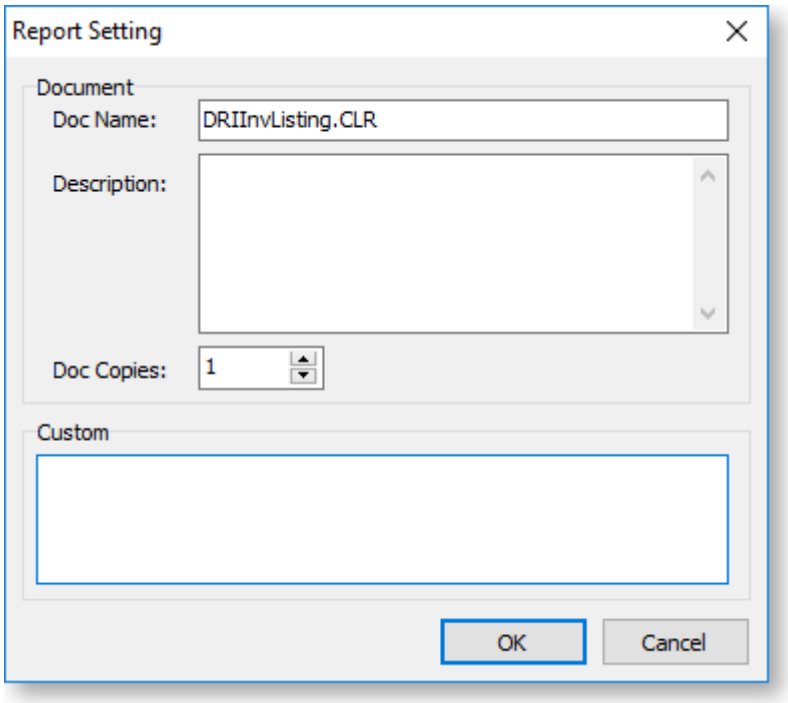
- A report is exported using one of these formats, if no configuration has been set up on the Print to File Setup window.
- The **Auto Layout** button on the Print to File Setup window is clicked (this sets up the configuration of the window, which you can then customise as necessary).

The automatic process works as follows:

1. Exo Clarity sets up the columns. It starts by looking at the bottom of the Header band, which is where column headings are most commonly located. It goes from left to right, looking for label components, which are added as columns to the Excel file in the order they appear.
2. It then works its way back up Header band and adds any other components it finds as extra columns to the right of the file, on the assumption that these are usually titles and parameter labels that aren't part of the main report data.
3. The data fields from the Detail band are then taken from left to right, and used to populate the values under each column heading.

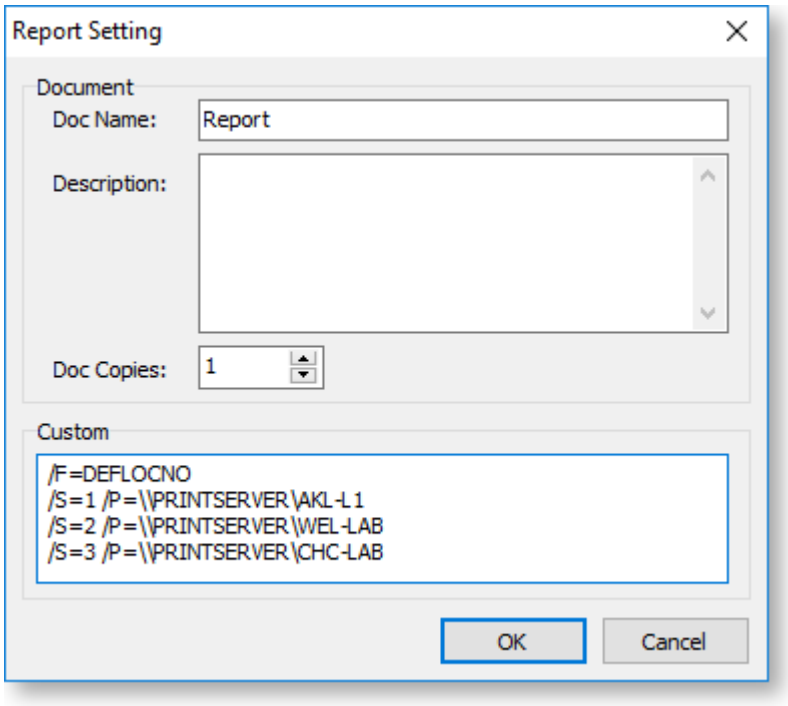
Report Printer Settings

The **Custom** field on the Report Setting window lets you enter advanced options that control the report's output. Open the Report Setting window by selecting **Setting** from the Design tab's Tools menu.



Setting the Printer Based on a Field

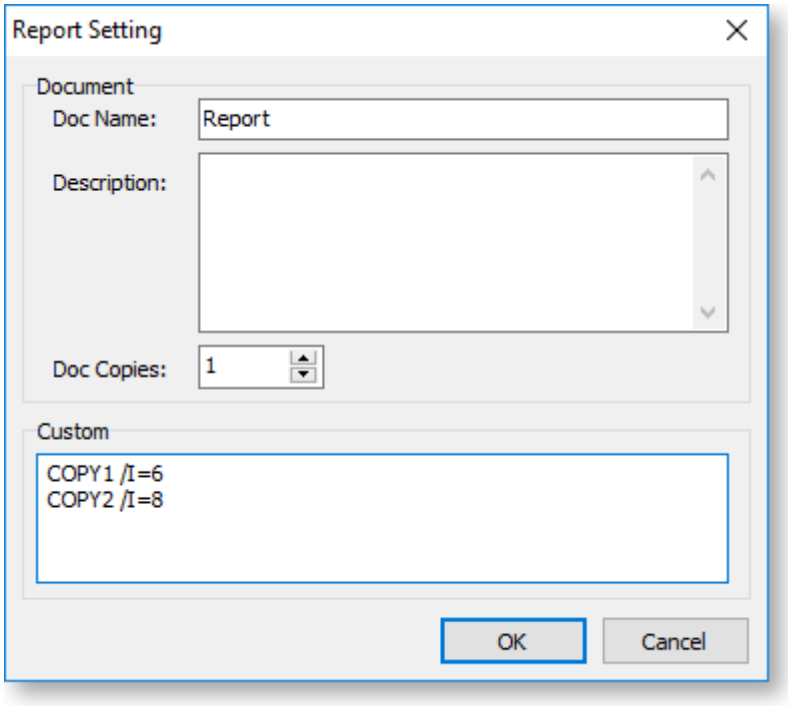
You can enter custom report settings to dynamically map printers based on the value of a field in the master pipeline. For example, you could map printers based on the value of the Database field DEFLOCNO, so that all invoices from Auckland print in the Auckland office, even if they were entered elsewhere. To do this, use the /F, /S and /P flags, as in the following example:



- /F specifies the field to use - in this example, the DEFLOCNO field is being used to determine which printer to use, but any field from the primary data source can be used.
- /S specifies the value of the field.
- /P specifies the UNC name of the printer to map to.

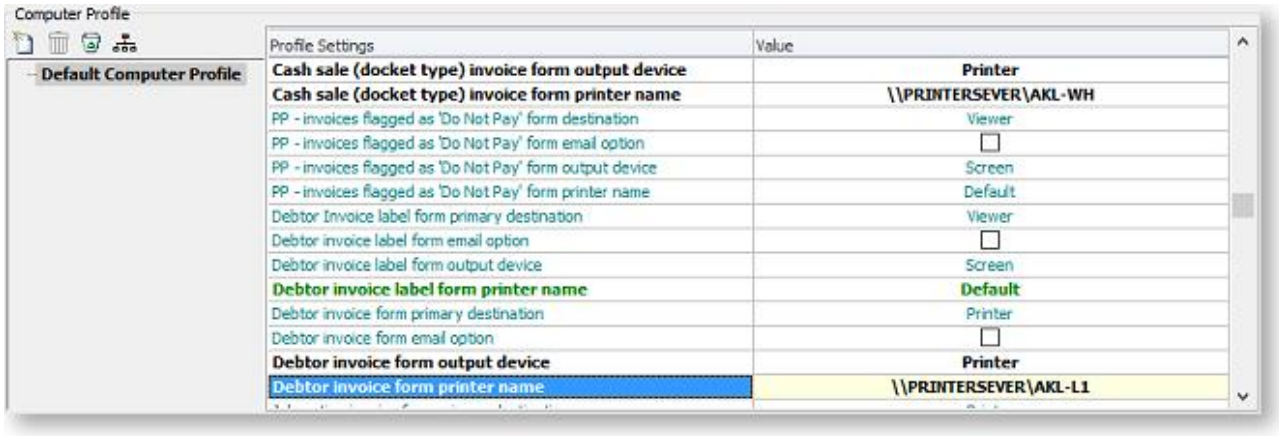
Setting the Printer for Different Copies

Custom report settings can also be used to specify a different printer for each copy of the report. For example, you could print packing slips in one office with second copy to the warehouse printer. To do this, use the COPYx and /I flags, as in the following example:



- COPYx specifies the copy, i.e. COPY1 is the first copy, COPY2 is the second and so on.
- /I specifies the Exo Business form to inherit settings from. This works in the same way as the /I runtime parameter - see Available Parameters for the list of available form IDs.

After you have associated each copy with a form, you can specify the printer to use by configuring the profile settings for the appropriate forms. For example, in the screenshot above, copy 1 is associated with form ID 6, which is the Cash sale docket form; copy 2 is associated with form ID 8, which is the Debtor Invoice form. You can map printers to these forms by setting the **Cash sale (docket type) invoice form printer name** and **Debtor invoice form printer name profile settings** in the Forms section of Exo Business Config:



Note: You can enter a single COPYx entry to manually redirect the report to print on a different printer.

Runtime Parameters

Using Runtime Parameters

As well as opening Exo Clarity reports from the Report Browser, you can launch them in other ways:

- Reports can be opened from the command line or from a batch file, which means you can schedule the automatic generation of reports.
- Reports can be opened from a custom button on an Exo Business window, giving you access to commonly used reports from the contexts where they are needed.
- Reports can be added to dropdown menus in Exo Business modules, so you can have a library of common reports that can be accessed quickly without having to go through the Report Browser. Because multiple report layouts can be set up and assigned to different Exo Business users, you can offer each user only those reports that are relevant to their role.
- Reports can be displayed as dashboard widgets.

When run in one of these ways, you can specify runtime parameters to select the report to run and to control the behaviour of the report, including the report destination and output format. Runtime parameters are also available to direct the report to ask for report parameters, or to specify these parameters directly so that user intervention is not required.

Note: If you are generating reports automatically, check the value of the **OpenFile** property under **Output - File** in the **Report Tree**. This value determines whether the report should open after it has been generated - for automatic or unattended report generation, this property should probably remain unticked.

Command Line

Reports can be run from the command line using a call to the **Clarity.exe** executable, followed by parameters. The format to use when running reports from the command line is:

```
Clarity.exe Connection Username Password Reportname Parameters
```

Where:

- `Connection` is the name of the Exo Business database connection to use.
- `Username` is the login name of the Exo Business user to run the report as.
- `Password` is the specified user's password.
- `Reportname` is the filename of the report to run, e.g. DRIInvListing.clr.
- `Parameters` is a list of report parameters in the format Parameter=Value. Parameter/value pairs are separated by spaces.

Example:

```
Clarity.exe EXONET_DEMO ExoAdmin ExoAdmin MyReport.clr /d=File
/f=c:\test.pdf /m=PDFFile
```

Custom Button

Custom buttons can be added to many windows in the Exo Business by setting up the appropriate **custom button** profile settings. For example, to add a custom button to the Debtor Account Details window, you need to enter values for the **Debtor custom button 1 caption** and **Debtor custom button 1 command line** settings. When editing the **command line** setting for the button you want to add, you can specify a report, along with any necessary runtime parameters, so that the report runs when the button is clicked.

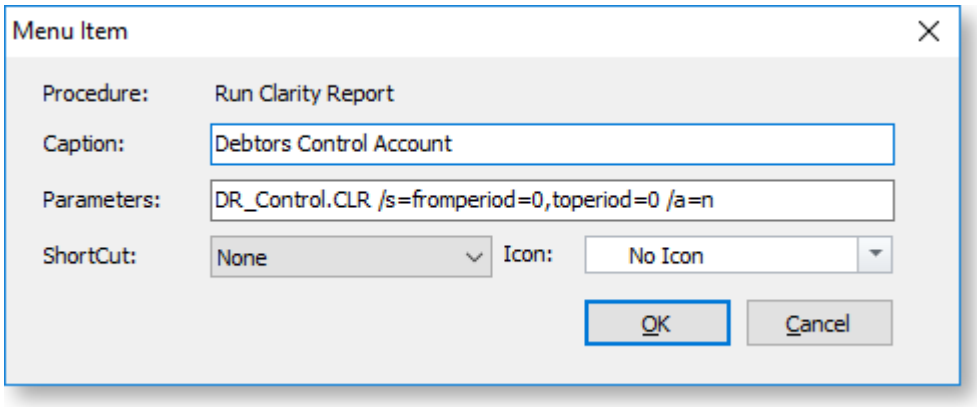
Unlike the command line scenario, you do not need to specify a connection and login details, since the user will already be logged in - you only need to include the name of the report and any parameters, e.g.

```
DRTLedger.clr /S=Accno=<CURRENT>
```

When setting up a report for a custom button, you can enter <CURRENT> as a parameter value - at runtime, this will be replaced with the current key identifier for the window that the button is on, e.g. on the Debtor Account Details window, this will be DR_ACCS.ACCNO; on the Contacts window this will be CONTACTS.SEQNO.

Menu Item

Exo Clarity reports can be added to the menus of Exo Business modules in the Staff > Menus > Dropdown Menus section of Exo Business Config. Open the menu you want to add the report to, then drag the **Run Clarity Report** procedure onto the menu. A window opens where you can specify the report to run and optionally any runtime parameters:



Note: A similar process is used when **adding a report to a dashboard widget**.

Dashboard Widget

You can specify runtime parameters when setting up a Clarity dashboard widget, in the same way that you can when settings up a menu item.

Available Parameters

The following parameters can be entered when running a report from the command line, a custom button, a menu or a dashboard widget. Some parameters are specific to only one or some of these scenarios.

Parameter	Description	Value	Default Value	Notes
Connection Name	The Exo Business database connection to use.			Only required when running a report from the command line, in which case this must be the first parameter specified.
Username	The login name of the Exo Business user to run the report as.			Only required when running a report from the command line, in which case this must be the second parameter specified.

Password	The specified user's password.			Only required when running a report from the command line, in which case this must be the third parameter specified.
Report Name	The file name of the Exo Clarity report to run.			When running from the command line, this must be the fourth parameter specified; in all other cases, it must be the first
/D or /d	The report destination to output to.	Printer, File, Viewer, Email, Grid	/d=Viewer	The File and Email options require the /F and /M parameters. The Printer option requires a valid /P or /I parameter. If not specified, the Viewer (Print Preview) window will appear, ignoring any inheritance direction values set by the /I parameter. It is good practice to always have this in any Exo Clarity command line call.
/F or /f	The filename for the output file.	A valid file name		The file extension must match the format specified by the /M parameter e.g. *.DOC or *.PDF
/M or /m	The output file mode.	TextFile, XLSData, XLSReport, XLSData, GraphicFile, HTMLLayerFile, HTMLFile, PDFFile, RTFFile, ReportTextFile	/m=RTFFile	Specifies the output format: • TextFile - TXT • ReportTextFile - TXT • XLSXData - XLSX • XLSReport, XLSData - XLS • GraphicFile - JPG, BMP or TIF • HTMLLayerFile - HTM • HTMLFile - HTM • PDFFile - PDF • RTFFile - RTF
/P or /p	The printer to print the report to.	'Default' or Printer name with the network drive portion in UNC format	/p=Default	This parameter is needed when the /D parameter is set to Printer. It is not used if a /I parameter is specified. "Default" is the Windows default printer. If the path is incorrectly entered, Exo Clarity will print to the default printer.

				The name to use is not the name of the printer as it is called on the server sharing the printer; it needs to be name as it is known on your machine, i.e. the printer name as displayed in the Control Panel > Printers and Faxes. The print server can be an IP address.
/C or /c	The number of copies to print.	1 or more	/c=1	Only used when the /D parameter is set to Printer. The copy counter and total copies are available for use in the report as global variables in the plGlobalVars data source.
/A or /a	Open the Clarity Report Parameters window to ask for report parameter values.	Y or y, N or n		Add /A=N to skip the confirmation of any parameters that exist for the report. Otherwise the report will run on defaults or values specified by the /S parameter. When used with a Clarity report dashboard widget, adding /A=Y will make the widget's Parameters tab appear by default when the widget first opens. If /A is set to N, or if it is not included, the Report tab is displayed when the widget opens.
/L or /l	Hide the Exo Clarity splash screen.	N or n		Only applies when running a report from the command line and asking for parameters (/A=Y). If this parameter is specified, the Exo Clarity splash screen will not appear when opening the report.
/S or /s	Specifies report parameter values.	Comma separated parameter=value pairs		Used internally for passing multiple parameters to a .CLF type file, but can also be used with a CLR file and associated CLS file. You must specify the parameter names as they are defined in the CLS file. If a parameter name is omitted or misspelled, then the parameter dialogue will appear at runtime, regardless of the /A parameter. Double quote marks that are part of the string are repeated to distinguish them from the quotes that surround the entire launch parameter string. If /A and /S are used together, any specified values will be pre-populated in the Clarity Report Parameters window.

/E or /e	Opens the report in the Clarity Report Designer for editing.	Other parameters are ignored.

/I or /i	Inherit the profile settings the .CLF file for an Exo Business form. Pas s in a form ID value based upon a predefine d list of constants in the Exo Business code. Can be used instead of specifyin g each individual ly if a similar report exists in Exo Business that has profile-controlle d settings.	Valid integer value	0	Principally for internal use by Exo Business developers. Current form IDs are: 0=riNone, 1=riBackord, 2=riBRemittance,3=riBStatement,4=riCRA ccount, 5=riCRInvoice, 6=riDocket, 7=riDRAccount, 8=riInvoice, 9=riJobInvoice, 10=riJobPreview, 11=riJobQuote, 12=riJobSheet,13=riManifest, 14=riPackslip, 15=riPOLine, 16=riPOQuote, 17=riPosReciept, 18=riPurchOrd,19=riPurchrec, 20=riQuote,21=riRemittance, 22=riSalesOrd,23=riSOLine, 24=riStatement,25=riStockItem, 26=riStockReceipt,27=riWorksOrd, 28=riIGReceipt,29=riShipment, 30=riIGCosting,31=riInvLabel 32=riGLAccSummary, 33=riCountSheets, 34=riStkVariance,35=riBomBatch, 36=riBillomat,37=riViewInv, 38=riChequeRepF, 39= riDDRepF, 40=riAMPurchases,41=riAMSales, 42=riAMBookValue, 43=riAMTaxValue, 44=riAMBookDepreciation, 45=riAMTaxDepreciation, 46=riAMServiceSchedule, 47=riAMListingByGroup, 48=riAMRevaluations, 49=riAmTransactionByAsset, 50=riPricePolicy, 51=riDRPaymentBatch, 52=riDrPmtReceipt, 53=riSupplyHist, 54=riBankBatchList 55=riPickingList, 56=riPickingRequisition,57=riStockReque st, 58=riSalesOrderLabel,59=riJobMisc, 60=riExotrackJob, 61=riExotrackJobList, 62=riExotrackJobDetail, 63=riSubsSummary, 64=riSubsDetail, 65=riSubsBillRunRep, 66=riSubsStockGroupSummary, 67=riFinPPManCheqWr, 68=riFinPPChqPayment, 69=riFinPPChequeRep, 70=riFinPPManChequeRep, 71= riFinPPDoNotPay, 72=riFinPPChqRunRep, 73=riFinPPEftRmit, 74=riFinPPCheqWr, 75=riFinPPSeqChqRep, 76=riFinPPDCRep, 77=riFinBRAPExceptionRep, 78=riFinBRUnRecRep,

79=riFinBRCloseBalRep,
80=riFinBRGLUnRecRep,
81=riFinBRDishonDDRep,
82=riFinBRRecAutoMatchRep,
83=riFinDDBatchRep,
84=riFinDDRemitance, 85=riShiftEnh,
86=riShiftSales,87=riShiftAudit, 88=riSTSale, 89=riSTReceipt, 90=riSTTfer,
91=riSTAdjustIn, 92=riSTAdjustOUT,
93=riSTCostAdjust, 94=riIGLabel,
95=riShift, 96=riStockTrans,
97=riPOSBankBatch,
98=riBatchSalesOrderProcessing,
99=riGLBatch,100=riSTKPricePolicy,
101=riACCPricePolicy,
102=riGLBatchTrans, 103=riBatchInvoice,
104=riCrPmtReceipt,105=riPickSlip,
106=riConfSlip,107=riStockValInactive

Note: /I should never be used in conjunction with other output / destination type command line parameters like /P or /D as it gets overridden by them.

Tips & Troubleshooting

Common Errors

Why are my runtime parameters not being passed though?

Check that the report's CLS file is in the default Exo Clarity directory (Custom or Masters, depending on which report you're trying to run).

Check that you don't have spaces in your parameters, e.g /S = MYPARAM = 12 will not work; you must remove the spaces: /S=MYPARAM=12.

Why do I get the message "Can't find Report Clarity.EXE"?

You can get this message when trying to run a Clarity report from a button or from a menu option if you have entered "Clarity.EXE" in the Command Line or Parameters field when setting it up. In the case of menu items and custom buttons, these fields should contain the report name and parameters only.

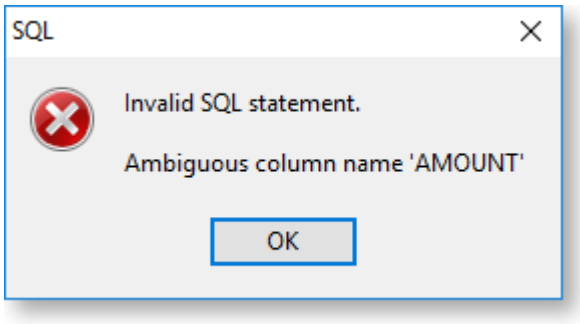
Why do reports with the /I parameter still go to the default printer?

Check that you are using a valid UNC pathname, e.g. \\PRINTSERVER\PRINTER NAME. If an invalid printer definition is given, Exo Clarity will always resort to the default printer.

UNC names are derived from the list of available printers on the network as they appear in Windows Explorer.

You may also need to specify the Direct to parameter (/D) when using the /I parameter on a command line to tell Exo Clarity to skip the printer/file window that it calls by default.

What does the error "Invalid SQL statement, Ambiguous column name" mean?



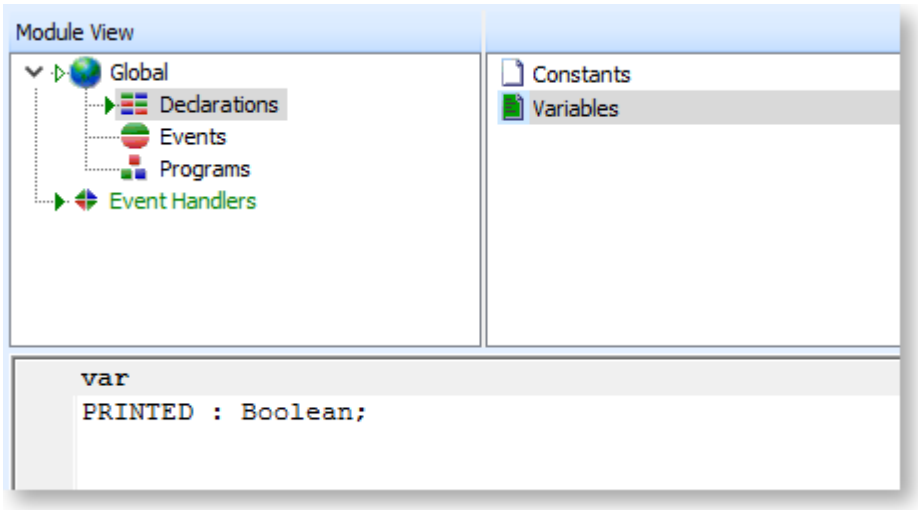
This error occurs when you enter a query that is not able to recognise which table the field belongs to, usually because you have only specified the field name, and there are two or more fields with the same name in the tables that you have selected for the data source. In this case you must explicitly specify the table name, e.g.

Sum(dr_trans.subtotal / dr_trans.exchrate)

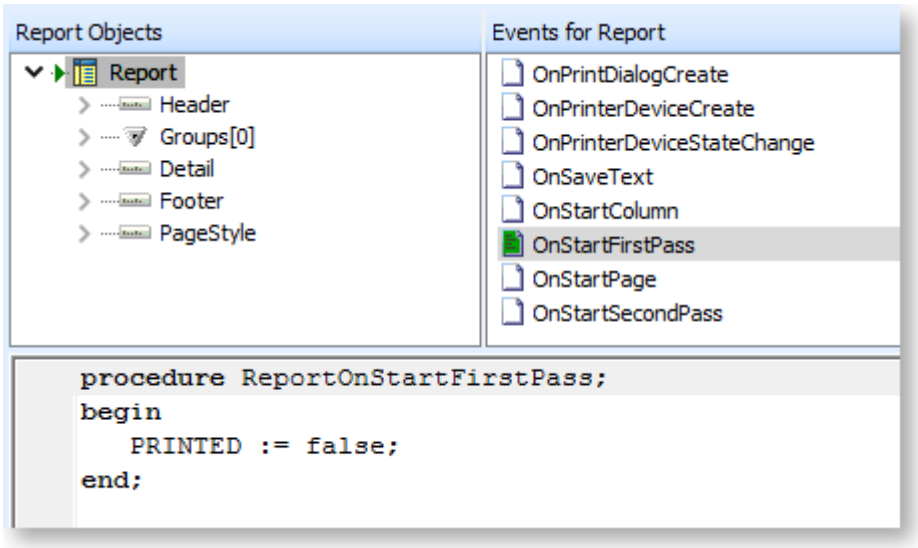
Printing Tips

How do I set an image to print only on duplicate copies?

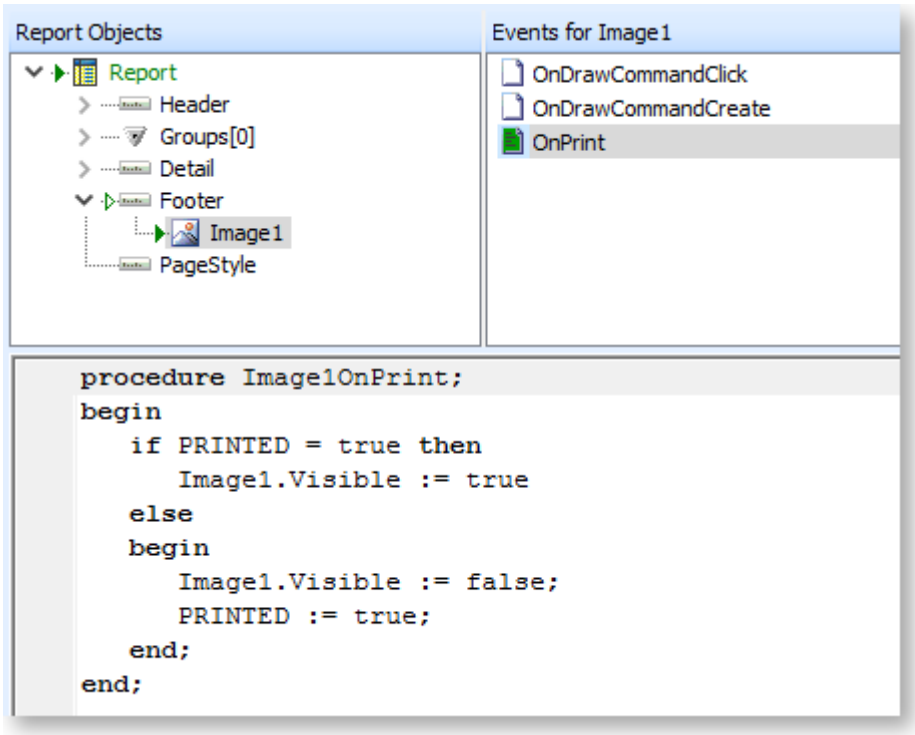
1. Declare a boolean global variable to hold the printed value.



2. Set the PRINTED variable to false at the start of the report.

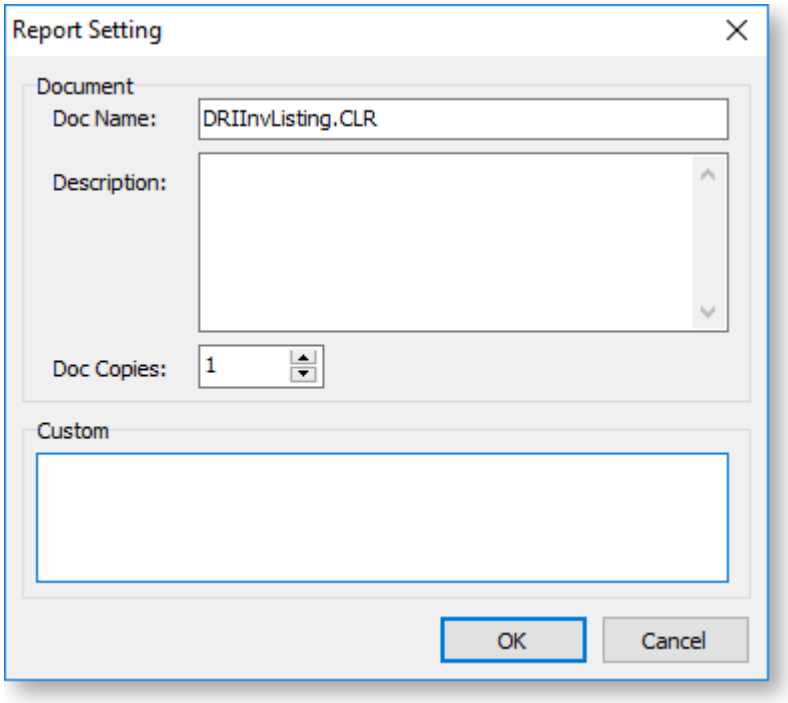


3. Set the set the image to visible if PRINTED is true, otherwise hide the image and set PRINTED to true.



How do I set the number of copies to print at design-time?

This can be set by entering a value for the **Doc Copies** property on the Report Setting window:



Open this window by selecting **Setting** from the Tools menu on the Design tab.

How do I make the second or subsequent copies print slightly differently?

Within one print session you can use the Copies and CopyCounter global variables to control the visibility of a label or content a variable type object on your form. One use of this is to print "Original" or "Copy" on various copies that are printed.

Example:

Add a label to your report and change the size, colour and position of the label to suit your needs. Change the label name to ShowCopy. Add this code in the ShowCopy labels' OnGetText event handler:

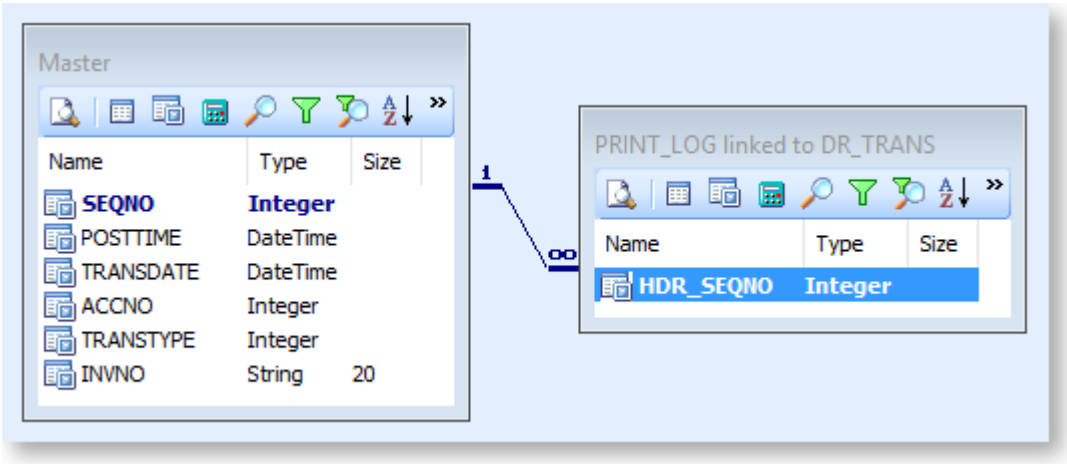
```
if plGlobalVars['CopyCounter'] = 0 then
   ShowCopy.Visible := false; {Customer copy shows nothing}
if plGlobalVars['CopyCounter'] = 1 then
 begin
    ShowCopy.Visible := true;
    Text := 'ACCOUNTS COPY';
 end;
if plGlobalVars['CopyCounter'] = 2 then
 begin
    ShowCopy.Visible := true;
    Text := 'FILE COPY';
 end;
```

With regards specifically to Tax Invoices and Copy Tax Invoices printed at a later date, you need to establish if there is a previous entry in the print log for this transaction from a previous print session.

> **Note:** This is quite different from duplicates within a single print session, where the print log entry is not made until the whole print session was successful.

Example:

Where the PRINT_LOG table is joined onto the DR_TRANS.SEQNO, you can use this to establish if there has been a previous successful printout of an invoice document:



Then using the value returned, you can incorporate this into a variable that provides the text for the title of your invoice document:

```
procedure InvoiceTextOnCalc(var Value: Variant);
begin
  if PRINT_LOG['HDR_SEQNO'] <= 0 then
    begin
      if Master['AMOUNT'] >= 0 then
        Value := 'Tax Invoice ' + Master['INVNO']
      else
        Value := 'Tax Credit ' + Master['INVNO'];
    end
  else
    begin
      if Master['AMOUNT'] >= 0 then
        Value := 'Copy Tax Invoice ' + Master['INVNO']
      else
        Value := 'Copy Tax Credit ' + Master['INVNO'];
    end;
end;
```

How do I dynamically map a printer based on the value of a field?

You can use the **Custom** field of the Report Setting window to specify different printers based on the value of a field in a data source.

How do I print a second copy to a different printer?

You can use the **Custom** field of the Report Setting window to specify different printers for different copies.

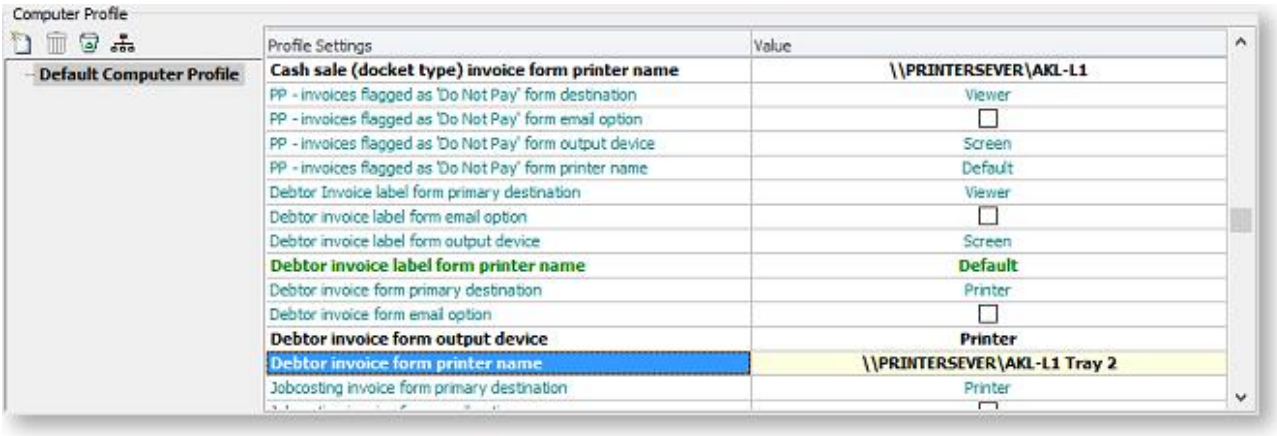
How can I direct a report to a different printer trays on a network?

The main difficulty with printing to different *trays* of a printer in lies with the convention of naming of printers in Windows, and the inability to be able to attach the same shared network printer twice on the client PC.

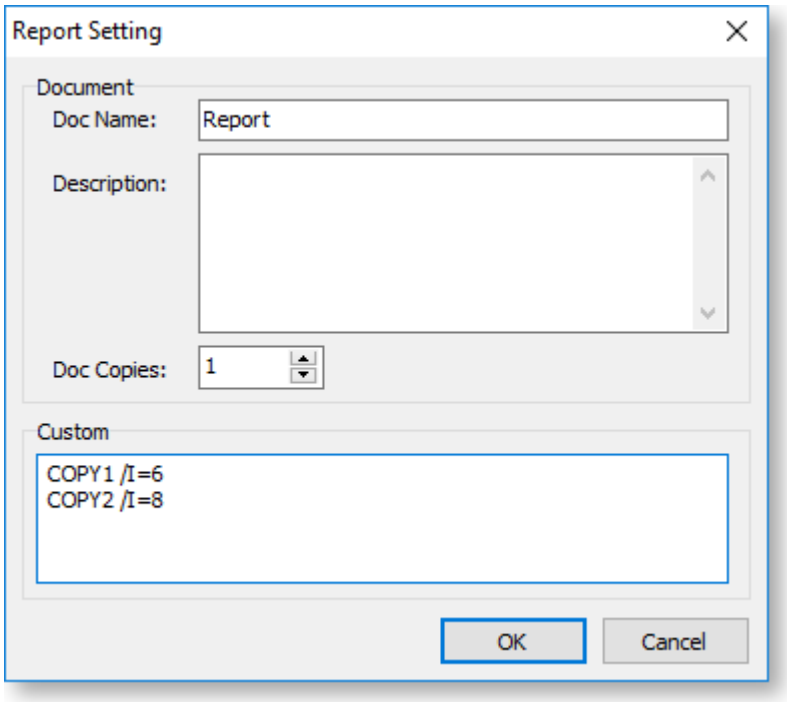
When you add a shared printer in Windows it is added with the name of "'Printer Share Name' on 'Computer it is shared on'" for example "HP8000 on Server2". This means that when you try to add the printer again so you can use another tray, it won't let you because the printer already exists. To get around this you need to add the printer alias and share it again *on the computer that it is already shared on*.

When you have added the printer to the printer list on the print server, change the printing preferences so that the Paper Source uses the tray required. This will be different between the makes of printers, but clients connecting to this shared alias will inherit this change. You can then add the new printer alias on client's PCs. Once the printers is set up you will now have to set your invoice to print two copies to the different trays.

The first step is to set the print options in Exo Business Config:



Then Change the Invoice Form to print two copies to different locations by selecting **Settings** from the Design tab's Tools menu when you have the **Invoice.clf** open.



Once this is all done when you now print an invoice normally, two copies will be printed to the two locations that were set up.

How do I set the number of copies to print from a database field?

Try the following code:

```
procedure ReportBeforePrint
begin
 Report.PrinterSetup.Copies:= <value>;
end;

{The line above could be any one of these:
 Report.PrinterSetup.Copies := 3;
 Report.PrinterSetup.Copies := Dr_accs['invfileno'];
 Report.PrinterSetup.Copies := plparams['CopiesValue'];
}
```

The only conditions in which the above piece of code would fail are:

- If there is a profile that controls the number of copies of a specific form.
- If the form is not previewed and is called through a custom button that directly sends the report to the printer.

Report Layout Tips

How do I dynamically resize a report?

Some reports need to resize dynamically, e.g. "Top X" dashboard reports need to resize so when you choose "Top 10" it only shows 10, and when you select "Top 50" it expands to show all 50. Without changing the page size dynamically, you'd need to have a fixed page that was as long as the longest that your report could possibly be; otherwise your data would flow onto second and subsequent pages (since you can't change pages in the Dashboard view, the data would be hidden from the user). Since having very long pages is not an ideal solution, page resizing is currently used on all the standard Dashboard reports.

In order to set the page size, you need to change a Printer Settings property of the report. In our case, we can set the page width and margins at design-time, and all we need to worry about at runtime is the page height. Here is an example:

```
Report.PrinterSetup.PaperHeight := nNewPageHeight;
```

Dynamic page height often depends on the number of records you have retrieved from the database, so the page resizing needs to occur after the detail bands have all been calculated. This becomes complicated, because resizing the page potentially alters the entire flow of all the sections of the report, it needs to happen before the final report has started generating. So it needs to be done in a four-step process:

1. Expand the report to be arbitrarily high in the Page Setup (say, 4000 pixels).
2. Allow the report to run through once - this should put all the Detail records on one page. (If we don't try to get them all on one page, often the records on the second and subsequent pages won't be calculated at all, since they are generated only when you move to page two.)
3. Count the number of detail lines and calculate how long the page should be - don't forget to take into account the top and bottom margins, all visible sections, all group headers and footers (one of each for each group), and all detail lines (one for each record in the primary data source). This is complicated by the fact that there's no way to calculate the actual printed height of sections that have dynamically adjusted height; you can only get the value of the height that was set in the design canvas. Try to avoid dynamic height sections (and stretching memo fields, etc.) for this reason.
4. Resize the page with the newly calculated height and refresh the report with the RefreshPage() procedure. This will redraw the report and re-query the database.

Here's the final piece of code that is used in many dashboard reports (not including the calculations are done to determine the page height):

```
Report.PrinterSetup.PaperHeight := nPageHeight
+ Report.PrinterSetup.MarginTop
+ Report.PrinterSetup.MarginBottom + 50;
```

You can find this in the "ResizePage" custom program of each report.

How do I force the Summary band to cover the whole of the last page of the report?

1. Enlarge the Summary section to be the size of your page.
2. Set the Header and Footer bands to be not visible if the page count has been reached:

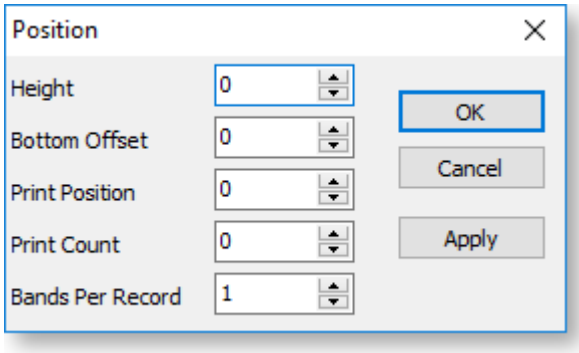
```
procedure ReportOnStartPage;
begin
    if Report.PageNo = Report.PageCount then
    begin
        Header.visible := false;
        Footer.visible := false;
    end
    else
    begin
        Header.visible := true;
        Footer.visible := true;
    end;
end;
```

3. When the Summary band prints, set the page limit property to the current page:

```
procedure SummaryBeforeGenerate;
begin
    Report.PageLimit := Report.PageNo;
end;
```

How do I force the Group Footer to appear at the end of each page?

Right-click on your Group Footer band and choose **Position**. Set the **PrintPosition** property so that the band aligns itself to the bottom of the page - the more you increase the value, the further down the page the band moves.



Note: Remember that each group will show on a separate page. The default nature of the group footer is to appear at the end of each group. If the "Group Footer" is positioned at the bottom of the page, then each group will end at the bottom of a page. Therefore each group will occupy at least 1 full page.

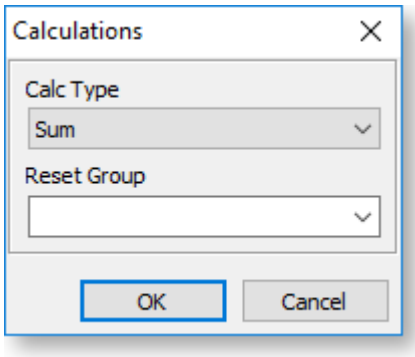
How do I skip a group when a condition is met?

Add code to the group's OnGetBreakValue event to see if the condition is met and if so cancel new page and set the Group Header, Group Footer and Detail bands to not visible.

```
procedure Group1OnGetBreakValue(var BreakValue: String);
begin
    if DR_TRANS['sumAmount'] = 0 then
    begin
        Group1.NewPage := false;
        GroupHeaderBand1.Visible := false;
        Detail.Visible := false;
        GroupFooterBand1.Visible := false;
    end
    else
    begin
        Group1.NewPage := true;
        GroupHeaderBand1.Visible := true;
        Detail.Visible := true;
        GroupFooterBand1.Visible := true;
    end;
end;
```

How do I count the number of records in each group at the end of the group?

Place a DBCalc component in the Group Footer band. Select the field for which you want to do the count from the Edit toolbar. Right-click on the component and select **Calculations** to open the Calculations window:



Select "Count" for the **Calc Type**. Placing this component in the Group Footer band will show the count of each group in the report. You can do the same to perform Sum and Average calculations.

> **Note:** You can select a **Reset Group** on the Calculations window to set when the calculation is reset to zero.

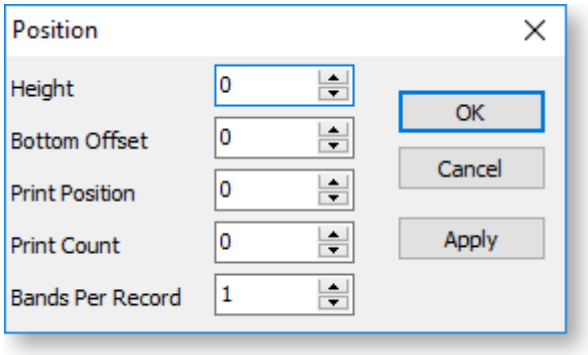
How do I get a record count in the header rather than in the footer?

Set up a DBCalc component to calculate the record count and place it in the Header or Group Header band. Right-click on the component and enable the **LookAhead** option - this will pre-calculate the lines before displaying the value.

> **Note:** This could delay the preview of complex or large reports while the value calculates.

How do I set the number of Detail lines per page?

Right-click on the Detail band and select **Position**. In the **Print Count** field, type the number of records you want to be displayed per page in your report.



How do I draw column lines in the Detail band?

1. Adjust the width of the detail band as required.
2. Add Line components to the Detail band everywhere you want a column line - for vertical lines, choose "Left" or "Right" from the dropdown in the Edit toolbar.
3. Resize each line manually to reach from the top of the to the bottom, or right-click on each line and select **Parent Height**.

To show an end line in each page of the report, add a horizontal Line component at the very top of the Footer band.

How do I set a watermark image or text in the background?

1. Add a background layer by right-clicking on **Design Layers** in the Report Tree and selecting **New Page Layer**. A tab for the new layer will appear at the bottom of the Report Tree - click this tab to edit the layer.
2. Place an Image component on the background layer. Right-click on the component and select **Picture** to choose the image file to display.
3. Right-click on the Image component and enable the **Stretch** and **Maintain Aspect Ratio** options, then re-size it as required. The picture will extend and spread along the whole page uniformly as the height increases.
4. Return to the foreground layer. Right-click on the other components on this layer and ensure that **Transparent** is selected for all of them - this removes any foreground colour on the fields, allowing the background image to be seen through them.

Note: It is a good idea to lighten the background image using an image editing program, otherwise it may make the report content hard to read.

Useful Code Snippets

How do I execute a stored procedure to pre-populate a table after parameters are entered?

The best place for this is in the ReportOnInitializeParameters() event handler. This is called after the plParams pipeline is populated, but before the main pipeline is opened. This allows you to use values from plParams in the event handler, and allows for parameters to be refreshed. For example:

```
procedure ReportOnInitializeParameters(var aCancel: Boolean);
begin
 aCancel := false;
 ExecuteSQL('EXEC testSPRepProc ' + plParams['noLinesValue']);
end;
```

How do I create a barcode on a clarity form as a concatenation of two fields?

Add an "Expression" field on the Calcs tab of the Query Designer for the primary data source. Enter the following expression, or change it to match your particular needs:

```
stock_items.stockcode + ' ' + stock_items.description
```

You may need to add some other control characters to the string, depending on the barcode type - check your barcode documentation.

This expression joins the stockcode and description fields together from the STOCK_ITEMS table. Change the name of this expression to something meaningful (such as "ConcBarcodeField"). You can then load this concatenated field into the Barcode component.

How do I make the Footer appear only on the last page of the report?

Type the following code in the "BeforePrint" Event of the Footer band:

```
Footer.Visible := Report.SecondPass and (Report.Page = Report.PageCount);
```

How can I strip blank spaces from a database field on a report?

Type the following piece of code in the OnGetText event handler of a DBtext component:

```
var
newtext : string;
x1 : integer;

begin
     newtext:='';
     for x1:=1 to length(text) do
          begin
               newtext:= newtext + trim(copy(text,x1,1));
          end;
     text := newtext;
end;
```

How do you set a TotalVariable component to be blank when zero?

Go to the Calc tab and select the TotalVariable component in the tree view (make sure the tree is set to Events mode). Enter the following code for the component's OnCalc event:

```
if Value = 0 then TotalVariable1.Visible := false;
```

How do I cancel a report if a runtime parameter is not specified?

Enter code like the following for the report's BeforeOpenDataPipelines event handler:

```
if (plparams.fieldobjects['PeriodValue'].isnull) then
begin
     ShowMessage('Please select a period');
     Report.Cancel;
end;
```

How do I display an RTF file?

Add a Rich Text component to the report and use the following piece of code to check for the existence of the file and display it.

```
if fileexists('c:\RichTextFile.rtf') = true then
     RichText1.LoadFromFile('c:\ RichTextFile.rtf');
```

How do I set a data source's range (record count)?

All the "Top X"-style dashboard reports allow you to filter on a number of records, or a range count. This is achieved by setting some properties on the data pipeline object before the data is queried or refreshed. It will not filter or limit a data set that has already been retrieved (unless you specifically force the data to be refreshed, which will re-query the database). In the OnBeforeOpenDataPipelines event (or other suitable place), you can put lines like the following:

```
Master.RangeBegin := rbFirstRecord;
Master.RangeEnd := reCount;
Master.RangeEndCount := nCount;
```

Where *Master* is the data source name and *nCount* is a positive integer specifying the maximum number of records to be retrieved. *rbFirstRecord* is a special internal variable telling the database where to start, *reCount* is a similar variable telling it where to finish, and *nCount* is the actual record count. If you'd like to set it back to returning all records, enter:

```
Master.RangeEndCount := reCount;
```

How do I enter custom colour values?

You can define any colour you like in Exo Clarity. You may have used the colour tools on the Design tab to come up with new colours, but you can also specify them in code if you know the Red/Green/Blue (RGB) values of the colour. The RGB function takes three integer parameters, one for each colour, from 0-255 indicating the intensity of that colour. So RGB(0,0,0) is black (no light of any colour) and RGB(255,255,255) is white (full intensity of red, green and blue).

You can use a custom colour in the following way:

```
Label1.Font.Color := RGB(200, 0, 0); { a relatively bright red colour }
Label2.Font.Color := RGB(100, 200, 100); { a pale green colour }
Label3.Font.Color := RGB(0, 0, 80); { a deep blue colour }
```

How do I show or hide a chart series?

You can set the series' **Visible** property to show or hide it as required:

```
Series.Visible := false; { hides this series }
```

How do I insert or clear data from a chart series?

You can manually insert data by getting the series you are looking for with the **SeriesByTitle** property, then calling the series' **Add** procedure with the label and value, e.g.

```
var Series : TChartseries;

begin
    Series := DPTeeChart4.SeriesByTitle('Series1'); Series.Add('First
item', 125.4); Series.Add('Second item', 131.8);
end;
```

To remove all items, call the Clear procedure:

```
Series.Clear;
```

> **Note:** Have a look at some of the "Top X"-style reports to see this in action. Check the subreport, looking at the DetailAfterGenerate event and the ReportOnStartFirstPass event.

How do I use code to send an email?

You can export a report to PDF and send it via email manually when running the report; to send emails programmatically, you need to set up the EmailSettings object, then call SendMail, e.g.

```
begin
   Report.EmailSettings.Recipients.Text := DR_ACCS['EMAIL'];
   Report.EmailSettings.FromAddress := 'myaddress@example.com';
   Report.EmailSettings.Subject := 'Here is your invoice';
   Report.EmailSettings.Body.Text := 'Please find your invoice
attached.';
   Report.EmailSettings.FileName := 'Invoice - ' + DR_ACCS['NAME'] +
'.pdf';
   Report.EmailSettings.PreviewInEmailClient := false;
   Report.SendMail;
end;
```

If the report has groups and you have ticked the **Email new file** option on the Groups window, Exo Clarity will export and send a separate file for each group. By adding code to the group's OnGetEmailSettings event, you can specify different email settings for each group.

Setting the PreviewInEmailClient property to false means that your default email program does not pop up a window preview the email before it sends (this can cause security warnings, however). Specifying a Body property means the email will be sent as plain text - you can specify a BodyRtf or BodyHtml property instead for an RTF or HTML email.

Index