

Sage Intelligence Reporting

Report Writing

Methodology

Report Writing Methodology.....	3
CONNECTOR.....	4
CONTAINERS:.....	4
REPORT MANAGER	6
EXCEL TEMPLATE FILE	10
PIVOT TABLES	10
PIVOT CHARTS	12
Using Slicers and Sparkline’s in Excel Templates	16

Report Writing Methodology

Always bear in mind the following key words before, during and on completion of the development of a report:

- Automation
- Flexibility
- Performance
- Ease of use for future support/maintenance

PRE-REQUISITES

Prior to developing a template it is recommend that a comprehensive requirement document be completed in addition to the following:

- Completion of Connector, Advanced & Basic Intelligence Training.
- Completion of the Excel on Steroids training.
- Adequate sample data.
- SQL knowledge is recommended for Connector tool usage.
- A sufficient Data Dictionary of the database being reported off.
- ODBC drivers are installed if required for the application in question.
- Examples of the desired output and layout required.

CONNECTOR



Note: "Graphical Joins", existing "SQL views" and "SQL Joins" are the preferred methods for creating containers and that wherever possible these should be used. Stored procedures and SQL query containers provide less flexibility for end users creating new or modifying existing reports off these containers. For example: parameterization of reports built off "SQL Query" and "Stored Procedure" containers can only be achieved using pass through variables which is an advanced feature of Sage Intelligence.

CONTAINERS:

When developing a container the following criteria should be borne in mind:

1. Type of join to be used (e.g. SQL join, graphical join etc.).
2. SQL expressions should be used as often as possible as these have a faster run time as Excel expressions and can also be used in aggregates.
3. Name containers as per the report name or similar to. This is for ease of use in recognising the contents of the container.
4. Where possible use SQL expressions to create calculated fields as opposed to using Excel formulas at the front end within Excel.
5. Provide meaningful descriptions on containers.
6. Include expressions with meaningful names in the container that may be useful to the client for future use even if they are not required on the report in question.

JOINS:

1. Use Graphical Joins unless you require specific clauses within the join, as this creates a visual entity relationship diagram.
2. Views and Stored Procedures should only be used if designing a report from a database where such Views or Stored Procedures are standard within the database and not client specific. An exception

would be if the need arises for an extremely complex report where the end result cannot be achieved in any other way.

3. For performance it is best to use index fields. However, where joins are required on non-indexed fields it is recommended that the developer index the required fields.

EXPRESSIONS:

1. Use data fields and SQL expressions as much as possible.
2. Pass through variables are a powerful feature but can make containers more complex to use. Only use them when there is no better way to achieve the same result.
3. Excel Formulas should be a last resort if the end result cannot be achieved through SQL expressions. Excel formulas have a negative effect on file size and performance.
4. Provide meaningful names for expressions and data fields, as often database fields are presented in a cryptic manner.

LOOKUPS:

1. Ensure that lookups are working properly on all expressions in the container and not just the expressions intended to be used in the report as other users may create reports off the container.
2. Where the functionality of the database allows for the use of the SQL distinct keyword, ensure that lookups display a distinct list of values.
3. For containers that access large data sets it is recommended that the SQL lookup type for all expressions be changed to "Direct from Container" or "SQL Statement Defined". This will optimise lookups and minimise the load on the database server.
4. Where necessary, set the lookup type to a Customised SQL Statement and customise the statement in order to achieve the preferred end result. The lookup of the expression does not always have to come from the table listed in the expression source; it can be set to obtain the value from a different table.

REPORT MANAGER



- **Note:**
- Before creating any reports ensure all expressions in the container are added to a report which is then run out. This is done to ensure that all expressions are run out successfully.
- Do not use macros. Sage support does not support the distribution of Report Sets with Reports that have calls to macros within them. All macro calls will be removed from reports during Metadata Compilation; this is as a result of macros referencing specific ranges. Should a user customise or alter a report that uses macros then this could cause macros to fail. Macros make reports impossible to support without advanced VBA skills.

1. The type of report being developed needs to be determined, e.g. a standalone report, union report etc.
2. It is recommended to document the reasoning for the use of a specific method in the developing of a report for future reference e.g. Union Report used because X and Y table could not be joined in a container.
3. Keep the number of reports per container to a minimum. It is recommended to duplicate a container if it is to be used for more than one report. This eliminates room for error when editing an expression. If numerous reports are using the same container and the same expression, changing the expression for the purpose of one report could result in a negative impact on the other reports.
4. Assign a unique report code to each report:
 - Report codes are added to reports in the report Manager on the Advanced properties screen of the report (i.e. select Show Advanced, then scroll down)
 - The report code should be made up as follows:
The abbreviation of the application and application version, the database type, the system class followed by a number, then the report version e.g. for an Accpac SQL version 6 General Ledger report the code would be similar to this AE-SQL-GL01-2-0

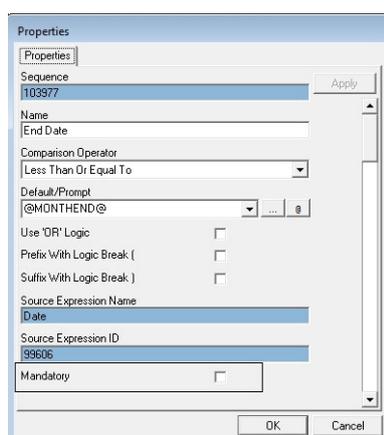
PARAMETERS

1. Rename parameter fields on the parameters properties screen where necessary to provide more meaningful options to the user at runtime. Set “Parameters on Second Sheet” on. Having the parameter values within the workbook can be extremely useful in report headings.
2. Where possible, insert system variables in the report parameter defaults e.g. @MONTHEND@ to eliminate possible errors at run time.

Name	Comparison Operator	Comparator	Use Or Logic	Logic Expression
Start Date	Greater Than Or Equal To	@MONTHSTART@	No	AND Start Date Greater...
End Date	Less Than Or Equal To	@MONTHEND@	No	AND End Date Less Th...

Figure 1: Report Parameter Defaults

3. Set each parameter field’s “Mandatory” tick box on



The screenshot shows the 'Properties' dialog box for a parameter field. The 'Name' is 'End Date', the 'Comparison Operator' is 'Less Than Or Equal To', and the 'Default/Prompt' is '@MONTHEND@'. The 'Mandatory' checkbox is located at the bottom of the dialog and is currently unchecked.

Figure 2: Selecting the mandatory parameter option

UNION REPORTS



Note: Union Sub reports run in Reverse Order (LIFO) within a union report.

1. Place all parameters on the first union sub report which runs. Should filtering be required on subsequent sub reports, Pass Through Variables can be used to pass these values to filters in the sub reports. This will eliminate parameter pop ups later in the run process.

- When adding a sub report to a union report, the reports are all set to run out on Sheet 1. The sub report run order can be amended by changing the “Output Sheet Number” on each union sub report.

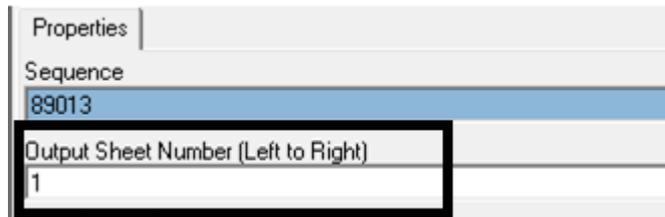


Figure 3: Setting the sub report run order



Always bear in mind that if the “Parameters on Second Sheet” has been selected, no sub reports should be assigned to Sheet 2 of the report

- When running out the report as seen in the graphic below, the Dashboard Sub report will be on sheet 1, the parameters on Sheet 2, the Dashboard Financials Grp Sub report on sheet 3, etc.

Properties		Union Sub Reports	
Name		Output Sheet	
→ Acc Group Code 3-0 (AE-SQL)		5	
→ Dashboard Financials Acc Sub 3-0 (AE-SQL)		4	
→ Dashboard Financials Grp Sub 3-0 (AE-SQL)		3	
→ Dashboard Sales Sub 3-0 (AE-SQL)		1	

Figure 4: Report output order

UNION SUB REPORTS

- Set “Report Hidden” on for all Sub reports, to eliminate confusion between the report and the sub reports making up that report. By doing this, the Sub report will not appear on the menu, but still be accessible via the tools menu.
- Sub Report properties have an option to “Abort Union Report if No Data”. This option should be checked on for any sub reports that must return data for a report output to make sense. Standard reports by default give a “No data” message when there is no data but since a union report pulls data from more than one report (e.g. a dashboard) you may still want to see the report even though some data is not there. There are however union reports that you would not want to run out if 1 or more of the Sub Reports don’t return data. An example would be a

Financial report that is a Union report. In these scenarios you should check this option for the Sub Reports. If there is no data for the sub report(s) then the report aborts.

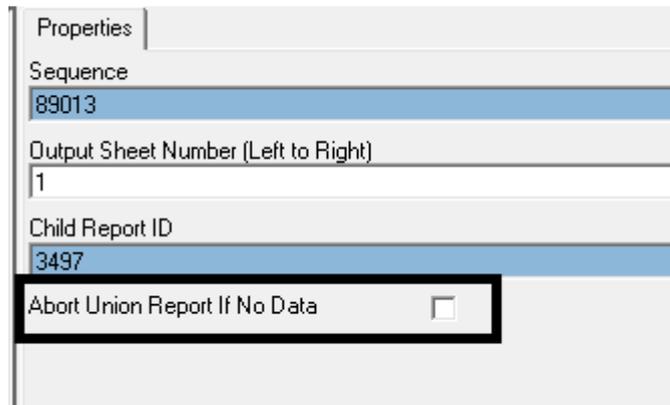


Figure 5: Abort Union Report option

3. “Make Available in Menus” needs to be selected on the relevant reports. This option must not be selected for drill-down and sub reports. This will ensure that the report is available through the Menu for use in application menus.
4. It is recommended that all Sub Reports to have the word “Sub” at the end of the report name to differentiate them from standalone reports that are intended to be run. Sub reports are not intended to be run as standalone.

EXCEL TEMPLATE FILE

When developing a template, it is recommended to do so using Excel 2003 and to link the template back as an .xlt file. This will ensure that the template is compatible on all versions of Excel. Any Report Designer templates created need to be developed using Excel 2007 and linked back as an .xltx file.

PIVOT TABLES

When using a PivotTable in the template file take the following into account:

1. PivotTable Limitations – ensure that you do not display too many row fields that could cause the pivot table limitation to be reached. Use report filter/page fields where necessary to further summarise data.
2. When pivoting data from a source data sheet where the data was rendered by Sage Intelligence, always use the named range *Sheetname!RawData*. This will ensure that the full range of data extracted is always available for use within the pivot table. When adding new columns to a report, these new columns of data will then automatically be included in the pivot table range.

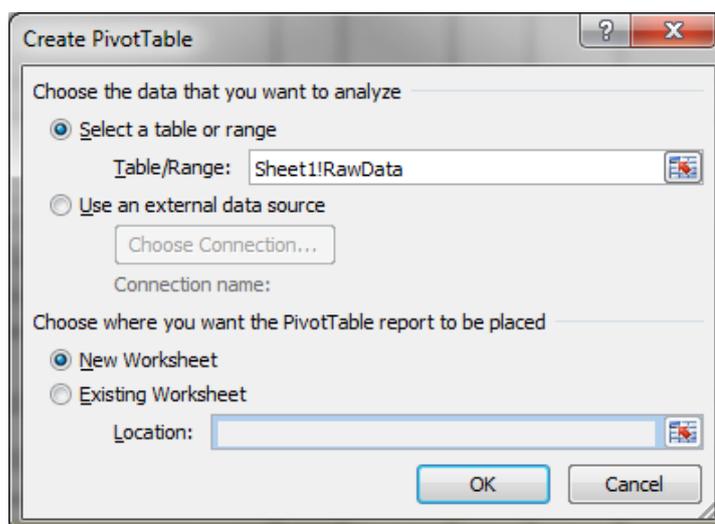
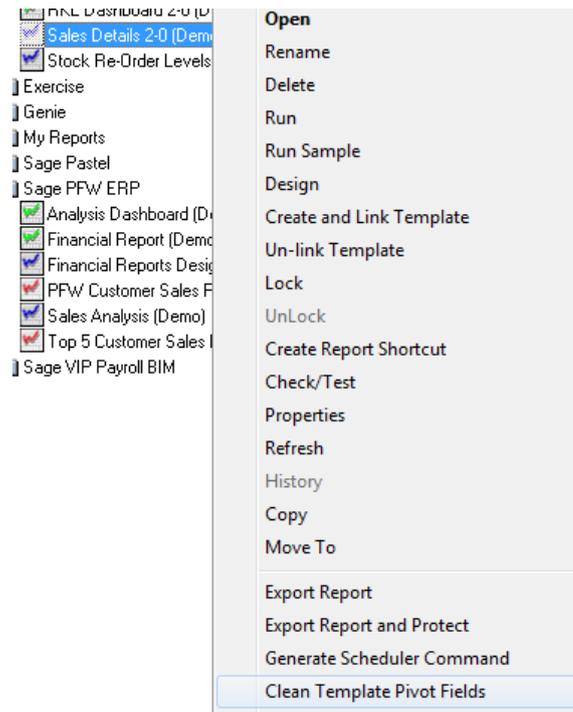


Figure 6: Selecting PivotTable Data Range

3. Turn off the feature “Save data with table layout”. This is a pivot table option which affects the file size when turned on, resulting in unnecessarily large template files.
4. Use the feature “Clean Template Pivot Fields” in the Report Manager to clear out pivot table items before exporting the report for delivery. This will ensure that the user of the report does not view items within the pivot table that were used in the development of the report.



5. Suggested PivotTable settings:

- Set the following footers on each page:
 - At the bottom left, set the date and time
 - At the bottom right, set page numbering
- Set print titles to have row headings repeated at the top of each page.
- Set print titles to have columns repeated at the left of each page where necessary.
- Preview the report and adjust the scaling where necessary, but to a minimum of 75%.
- Reset margins where necessary.

6. Formatting should be uniform on all reports to ensure the end product has a consistent feel.

7. When formatting a pivot, the use of bright colours should be avoided. It should also be taken into account that colour schemes may vary on different systems.

8. It is recommended that the “Quick Pivot” & “Format Pivot” tool be used to format all pivot tables. This can be found under the “Add-Ins tab in excel.

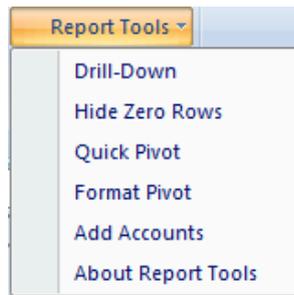


Figure 7: PivotTable format menu

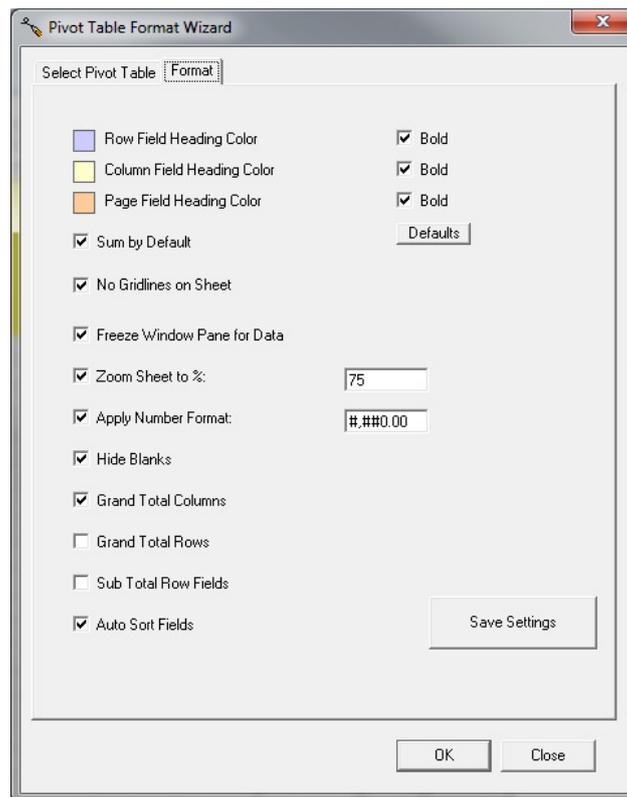


Figure 8: Standardizing the format of PivotTables

PIVOT CHARTS

When creating pivot charts it is recommended that formatting be set to “automatic”. This will ensure that a pivot chart automatically reformats when used in different Excel versions.

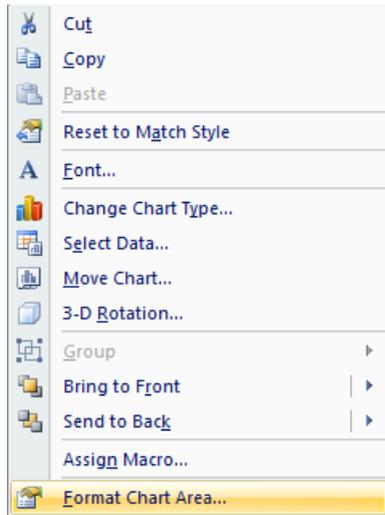


Figure 9: Pivot Chart format menu

All areas of a pivot chart can be formatted. To format the various sections of a pivot chart:

- Vertical and Horizontal Axis – right click on the axis labels and select Format Axis

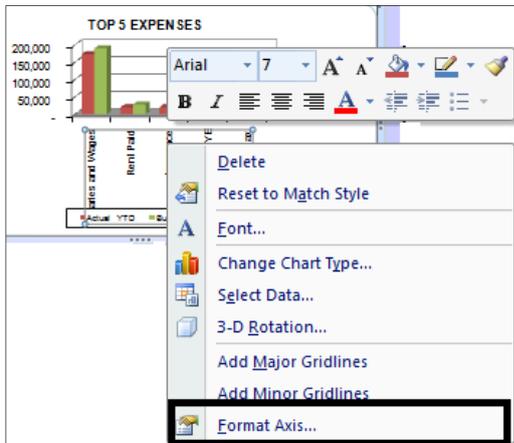


Figure 101: Format Axis

- Right click on the vertical wall of the pivot chart to select Gridlines and the Vertical Axis

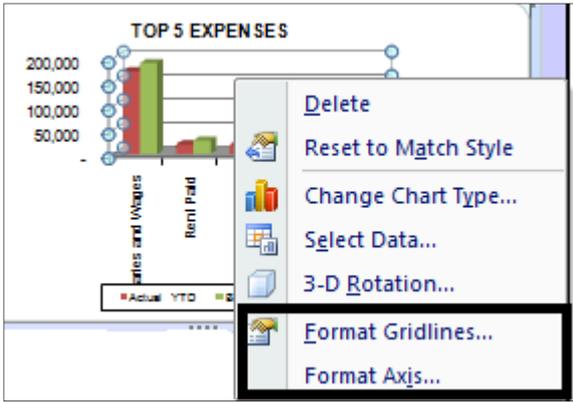


Figure 112: Format Axis & Gridlines

- Click inside the pivot chart to select the back wall

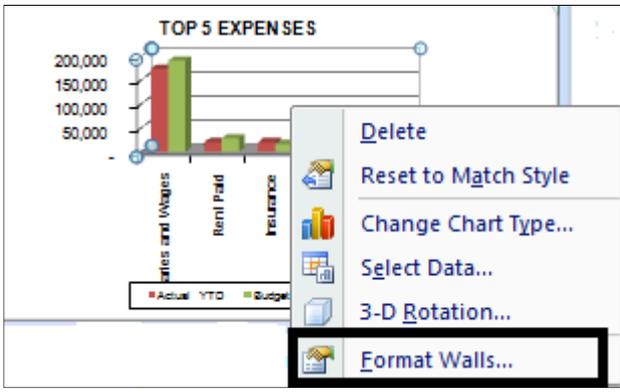


Figure 123: Format Walls

- Right click on the series to format the data series

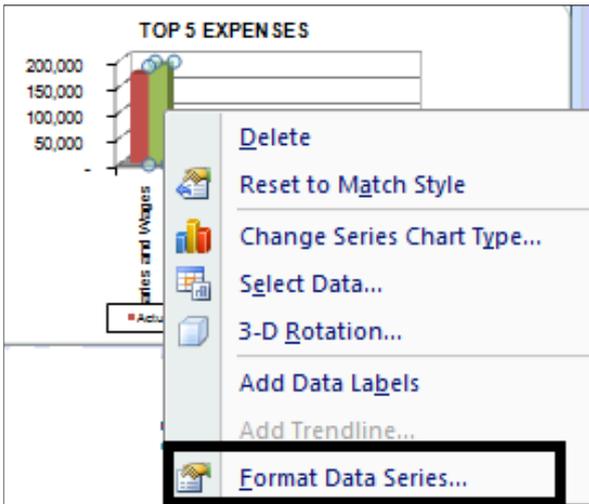


Figure 134: Format Data Series

- Right click in the legend to format

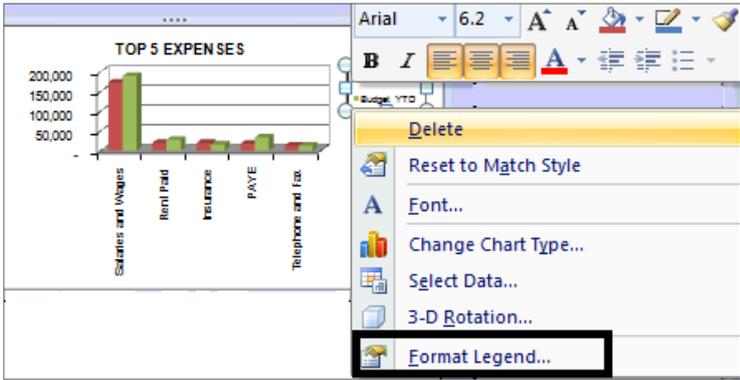


Figure 145: Format Legend

- After selecting the relevant format menu, the following should be set to Automatic

➤ Fill

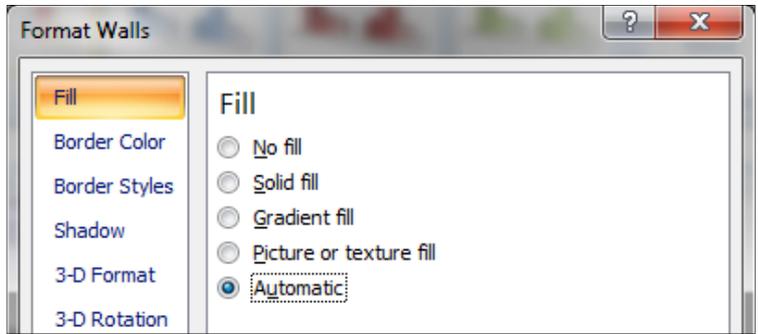


Figure 156: Set Fill to Automatic

➤ Border colour

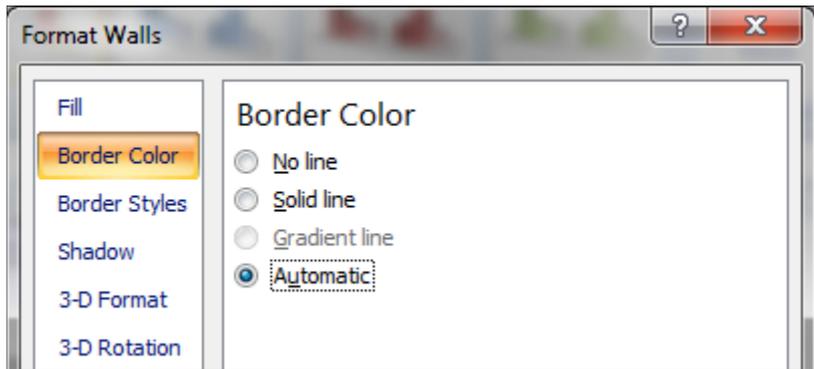


Figure 166: Set Border Color to Automatic

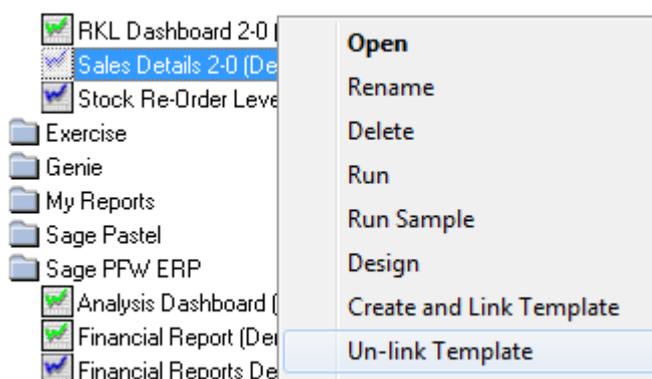
EXCEL FORMATTING

1. Set report headings at the top of each worksheet that is used as a final end result. Include parameter values within the report headings where necessary.
2. Set “Freeze panes” to enable easy scrolling around the worksheet without losing view of report headings etc.
3. Hide the display of zero values on the worksheets.
4. Hide the display of gridlines on the worksheets.
5. Adjust the zoom display if necessary but to a minimum of 75%.
6. Ensure the correct worksheets in the workbook have been hidden.
7. The template is to be saved on the correct worksheet with the cursor position in the first blank cell.

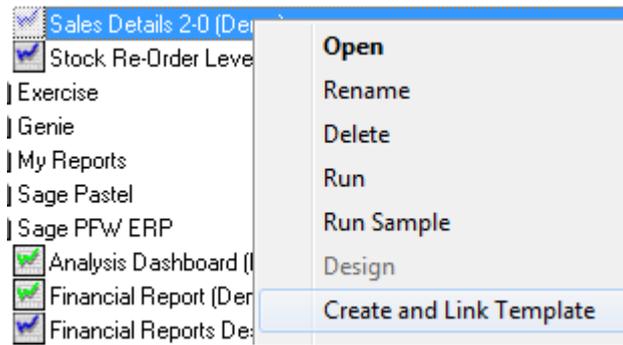
USING SLICERS AND SPARKLINE'S IN EXCEL TEMPLATES

When developing a template, it is recommended to do so using Excel 2003 and to link the template back as an .xlt file. This will ensure that the template is compatible on all versions of Excel. However if you want to use Sparkline's or slicers in your reports you need to make your templates Excel 2010 compatible.

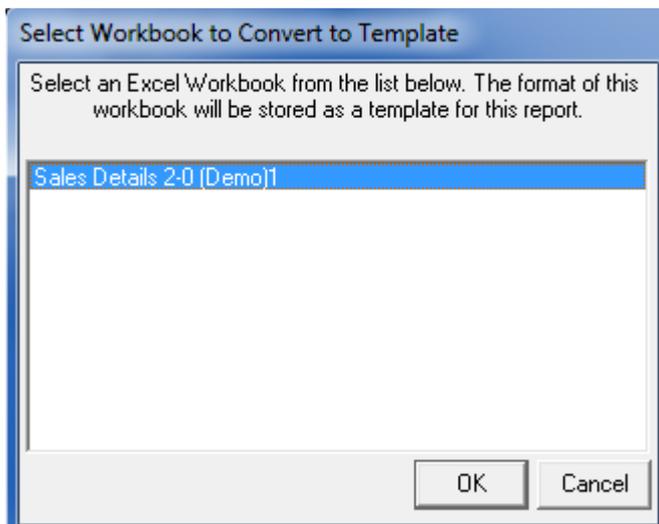
1. Run out your standard report into Excel from the Report Manager
2. Go back your Report Manager and right select on the report you have run.
3. Select Un-link template



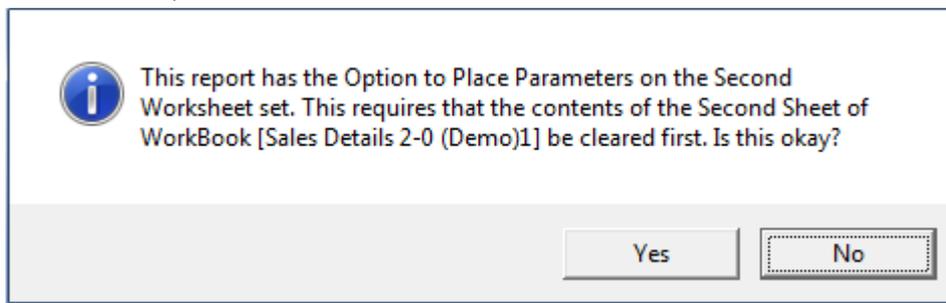
4. The Excel Template is no longer linked to the report.
5. Right select on the Report again and select Create & Link Template



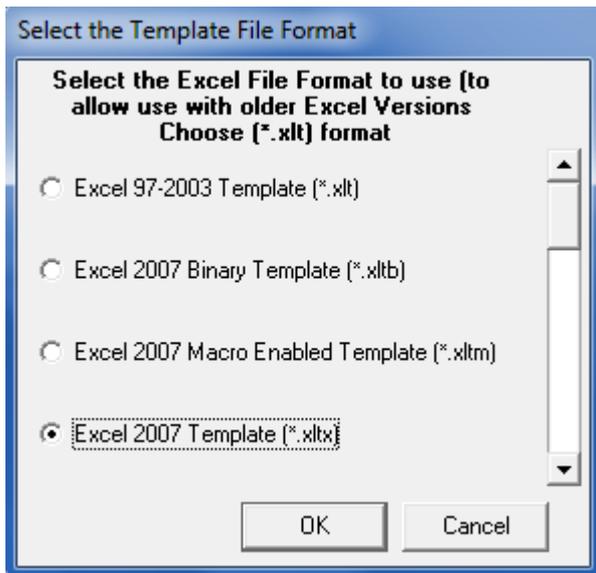
1. The select a workbook to Convert Template screen will open, select the workbook from the list. Select OK



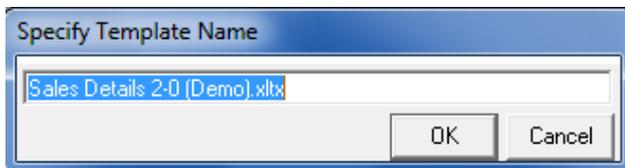
- 2.
3. Select Yes, to place Parameters on the Second Sheet



- 4.
5. Select the Template File Format to use. This time select Excel 2007 (.xltx) this will ensure the template is compatible with Excel 2007 and Excel 2010. Allowing you now to use slicers and Sparkline's in your reports.



- 6.
7. Specify the Template name, you will notice it is now a (.xltx)



- 8.
9. The Excel Template will be linked to the report as an Excel 2007 Template (*.xltx)

