

{REMOVED} SPC Charts Developer How-To Guide for SQL Query, Power BI Template, and Excel Tool

Version 1.1.1



Change log

Version	Date	Author	Comment
1.0	14/12/2021	{REMOVED}	Development of guide
1.1	15/03/2022	{REMOVED}	Updated to SQL v2.3, Power BI v1.2, and Excel v1.4 <ul style="list-style-type: none"> • Added T and G charts <ul style="list-style-type: none"> • Updated metric format guidance • Updated low mean warning value guidance • Updated value guidance • Updated ghost flag guidance • Updated target guidance • Renamed Power BI table and pages • Added Power BI page • Updated hard-coded Excel references • Updated guidance on setting baselines • Updated guidance on resetting Power BI dropdowns • Added Power BI summary icon guidance • Added Power BI chart axes guidance
1.1.1	08/04/2022	{REMOVED}	Updated to Excel v1.4.1 <ul style="list-style-type: none"> • Updated guidance on connecting to SQL Server in Excel



Contents

Change log.....	2
Contents.....	3
Introduction	4
Intended audience	4
Versions.....	4
Contact.....	4
SQL query	5
SQL Server version	5
Structure	5
Settings.....	6
Data requirements	7
Adding data	9
Executing the query	10
Power BI template	11
Connecting to SQL Server	11
Pages	11
Customising pages	12
Additional columns	13
Customising charts.....	13
Summary icons.....	13
Chart axes.....	13
Excel tool.....	14
Connecting to SQL Server	14
Additional columns	15
Making changes	15



Introduction

The {REMOVED} team of {REMOVED} have developed and released a suite of tools to allow individuals and organisations to create statistical process control (SPC) charts.

Intended audience

This guide explains the process for using the SQL query and the accompanying Power BI template and Excel tool.

This guide is for the developer who is using these tools to either build a report or create ad hoc SPC charts. It assumes a certain level of knowledge of the tools, along with access to the necessary programs with the necessary admin rights. It also assumes an understanding of SPC charts.

This guide is not for the end user. There is a worksheet, *Instructions*, within the Excel tool that includes simple instructions. There is no corresponding page in the Power BI template.

Versions

Before we begin, check you are using the latest versions of the files and this guide, which can be downloaded from the {REMOVED}. If you are not able to self-join the workspace, contact us on the email below.

The versions used within this guide are:

Program	Filename	Version
SQL	{REMOVED} SPC Charts SQL Query	v2.3
Power BI	{REMOVED} SPC Charts Power BI Template for SQL Query	v1.2
Excel	{REMOVED} SPC Charts Excel Tool for SQL Query	v1.4.1

If you are using old versions of the files, check the FutureNHS workspace for previous versions of this guide.

The link above can also be found at the start of each file. It is advised you use this link to regularly check for updates. Even if you do not plan to switch to a newer version, the change log at the start of each file may notify you of a correction that you need to replicate in your version.

Contact

For queries and feedback on either this guide or any of the tools, please contact the MDC team by email at {REMOVED} and quote the name and version number.

We will begin with the SQL query.

SQL query

There are two stages to creating an SPC chart:

1. Take the data and calculate the SPC elements, such as the process limits, common cause or special cause variation, and the summary icons.
2. Take this expanded dataset and plot it as an SPC chart.

The SQL query handles the first stage. The Power BI template or Excel tool are then used for the second stage. It is also possible to use the SQL query with another business intelligence tool if you wish to develop your own solution.

The current version of the SQL query supports XmR, T, and G charts.

SQL Server version

Before we change the SQL query, we need to check that we are using a compatible version of SQL Server. If you are unsure, execute the following query:

```
SELECT @@VERSION
```

The SQL query has been tested on SQL Server 2012. It will not work on an older version. If you are using an older version, check the {REMOVED} for alternatives. If you create your own version, you can share it there too or contact the MDC team.

Structure

The SQL query is over 2000 lines long, which is why it has been split into multiple steps and sub-steps, each beginning with guidance on whether to make any changes, and if so, what.

The seven steps and their sub-steps are:

Step 1	<i>Setup</i>		
Step 2	<i>Settings: Special cause</i>		<i>Process limits</i>
	<i>Settings: Other</i>		<i>Special cause: single point</i>
Step 3	<i>Metric data</i>		<i>Special cause: shift</i>
	<i>Raw data</i>		<i>Special cause: trend</i>
	<i>Baseline data</i>		<i>Special cause: two-to-three sigma</i>
	<i>Target data</i>		<i>Special cause combined</i>
Step 4	<i>Hierarchy</i>		<i>Icons</i>
	<i>Partitions</i>		<i>Row count</i>
	<i>Baselines</i>		Step 5 <i>Warnings</i>
	<i>Targets</i>		Step 6 <i>Output</i>
			Step 7 <i>Clear-up</i>

The change log is found at the start of the SQL query.

At the bottom is an index, which details all the columns used. It includes the data type, whether it is returned, what step or steps it is created or updated in, and how its value is derived.

The SQL query comes pre-populated with sample data, which you need to replace with your data in Step 3, but first, we might want to change some settings.

Settings

In Step 2, there are sub-steps, containing a total of 12 parameters that can be changed, which all affect how the SQL query executes.

Step 2a contains three parameters that affect how the process limits and special cause rules are calculated, which also affects the summary icons. Step 2b contains the other nine parameters.

There is a description above each parameter that explains what it does and what the valid options it can be changed to are.

If you do not wish to change any of these settings, you can skip to adding your data in Step 3.

The parameters are:

Parameter	Purpose	Options
@ExcludeMovingRangeOutliers	Recalculates the process limits after removing any high points on the mR chart.	On/Off
@SettingSpecialCauseShiftPoints	The number of non-ghosted points necessary to trigger the special cause rules of a shift and trend, respectively.	Any
@SettingSpecialCauseTrendPoints		
@SettingGlobalWarnings	Whether warnings are returned.	On/Off
@SettingMinimumPoints	The minimum number of non-ghosted points needed for an SPC chart. Otherwise, a run chart is returned, and a warning is or is not returned.	Any
@SettingMinimumPointsWarning		On/Off
@SettingMinimumPointsPartition	The minimum and maximum number of non-ghosted points needed for each step of an SPC chart. Otherwise, a warning is returned.	Any
@SettingMaximumPointsPartition		Any
@SettingMaximumPoints	The maximum number of points that can fit on a chart. Otherwise, a warning is returned.	Any
@SettingPointConflictWarning	Whether warnings are returned when special cause rules for both improvement and concern are triggered. Either for every point or just when the variation icon is affected.	On/Off
@SettingVariationIconConflictWarning		On/Off
@SettingGroupHierarchyIndentSpaces	The number of spaces used for each level of the group hierarchy.	Any

Now that the settings have been changed, or kept as their defaults, we need to know what data to add.

Data requirements

Step 3 is split into four sub-steps, each of which follows the same structure. A temporary table is created, which is then populated, originally with the sample data included.

Before adding your data, remove the INSERT INTO lines that add the sample data in these four sub-steps. You may prefer to comment these out rather than removing entirely to keep as reference.

Steps 3c and 3d add baselines and targets, respectively. Targets can only be added for XmR charts. If you do not have any baselines or targets to set, you can leave these tables empty, but you must still remove the sample data. **Do not remove either of the CREATE TABLE sections** as the temporary table still needs to exist, even if they are not populated.

Guidance of what the columns in each temporary table is for and what values are accepted is found at the top of each sub-step.

The columns in #MetricData in Step 3a are:

Column	Purpose
MetricOrder	Determines the orders the metrics appear, automatically by IDENTITY(1, 1), which is the same order they are inserted into the temporary table. It can be removed, and the column inserted manually.
MetricID	Distinguishes one metric from another so must be unique. Intended to be a short in length and only shown in the Power BI template and Excel tool where space is minimal.
MetricName	The descriptive name for MetricID used predominately in the Power BI template and Excel tool.
ChartType	Determines which SPC chart type to use: <ul style="list-style-type: none"> • XmR for most cases • T for rare events with only the dates of events provided • G for rare events with dates provided and the number of opportunities since the last event.
MetricImprovement	Determines the direction of improvement and, therefore, the colours used for special cause variation and variation icon: <ul style="list-style-type: none"> • Up or Down uses blue and orange for improvement and concern, respectively. • Neither uses purple for both.
MetricConflictRule	Determines which takes prominence when a point triggers special cause rules for both improvement and concern: <ul style="list-style-type: none"> • Improvement or Concern (when blue and orange). • NULL (when purple)
MetricFormat	Determines what format the value is displayed in within the charts and tables: <ul style="list-style-type: none"> • General uses whole or decimal numbers • Percentage uses percentage in whole or decimal numbers

	<ul style="list-style-type: none"> Time must be entered as the number of 24-hour days (can be greater than 1). It displays as time in the Excel tool and summary tables in the Power BI template but remains in decimal format within the chart in the Power BI template. <p>Set as General for T and G charts.</p>
LowMeanWarningValue	Determines whether a warning is returned when an SPC chart has a mean at any point under the value set, which can be used as an indicator that an XmR chart may not be the right chart type. It must be set as a number or NULL. Set as NULL for T and G charts.
ChartTitle	Determines what title is shown, along with the GroupName. If there is no title, set as NULL, the Power BI template uses MetricName instead, but the Excel tool only shows the GroupName.
HorizontalAxisTitle	Determines what titles are used for the horizontal and vertical axes, respectively. If there is no title, set as NULL. The titles are shown in Excel tool, but only as summary data on the <i>SPC Chart</i> page in the Power BI template.
VerticalAxisTitle	

The columns in #RawData in Step 3b are:

Column	Purpose
MetricID	This must match #MetricData
Group	Distinguishes one chart from another for the same metric and is descriptive as it is shown in the Power BI template and Excel tool.
GroupParent	Determines the hierarchy for Group: <ul style="list-style-type: none"> The parent Group that this is a child of. NULL for the top of the hierarchy or to not set one.
Date	The date for the point. If the data are not daily, use the day at the start of the week/month/etc.
Value	The value for the point in decimal format. For example: <ul style="list-style-type: none"> 90% as 0.9. 6pm as 0.75. Set as NULL for T charts.
RecalculateLimitsFlag	Whether to recalculate the process limits: <ul style="list-style-type: none"> 1 to recalculate. 0 to not.
GhostFlag	Whether to ghost the point, which has the effect of it not being included into the calculations of SPC elements, but still be shown on the chart: <ul style="list-style-type: none"> 1 to ghost. 0 to not. Set as NULL for T and G charts.
Annotation	Determines what annotation to show. If there is no annotation, set as NULL. The annotations are shown in the Excel tool, but only as a circled marker and a tooltip in the Power BI template, and only with a tooltip in XmR charts.

The columns in #BaselineData in Step 3c are:

Column	Purpose
BaselineOrder	Determines which baseline to keep – the first – if multiple baselines are added to an SPC chart, automatically by IDENTITY(1, 1). It can be removed, and the column inserted manually.
MetricID	This must match #MetricData and #GroupData.
Group	This must match #GroupData or be set as NULL to be added to all groups for that metric.
Date	At least one of these columns must be added to determine the date the baseline is set up to or the number of non-ghosted points. This is the last point in the baseline, similar to recalculating the next point. If both are added, and they do not match, the date is used. If only one is added, set the other column as NULL. If there is no baseline, do not add any data.
PointsExcludeGhosting	

The columns in #TargetData in Step 3d are:

Column	Purpose
TargetOrder	Determines which target to keep – the first – if multiple baselines are added to an SPC chart, automatically by IDENTITY(1, 1). It can be removed, and the column inserted manually.
MetricID	This must match #MetricData and #GroupData
Group	This must match #GroupData or be set as NULL to be added to all groups for that metric.
Target	The target value for the point in decimal format. For example: <ul style="list-style-type: none"> • 90% as 0.9. • 6pm as 0.75. If there is no target, do not add any data.
StartDate	If a target has changed during the date range on the chart or it has started or stopped within the date range, these columns can be added. The dates are inclusive, so can be used to cover multiple points, or can be inserted for each point by setting StartDate and EndDate equal to Date. If the target remains the same over the entire date range, set as NULL.
EndDate	

Additional columns can be added to #MetricData and #RawData. These need be added to Step 4b, to be included in the temporary tables, and Step 6, to be returned with the output table. The first column of Step 6 must remain RowID. We will look later at how to add these to the Power BI template and Excel tool.

Now that we know what data to add, we can add it.

Adding data

The data can be added in a few ways.

One method is to replicate that used with the sample data, which is to use INSERT INTO statements. If the data are not already in SQL server, this may be the best method to use.

There are four worksheets within the Excel tool, *SQL #MetricData Inserts*, *SQL #RawData Inserts*,

SQL #BaselineData Inserts, and *SQL #TargetData Inserts*, that can be used to generate these statements.

Other methods include using a query to link to a table, or tables, within SQL Server that contain the data, or using a stored procedure.

The syntax necessary for these can be found commented out at the bottom of Step 3b. If used for a different step, update the name of the temporary table in the `INSERT INTO` line.

Now that we have added the data, it is time to execute the SQL query.

Executing the query

When the SQL query is executed, a message is printed after each step or sub-step with a time stamp. This allows for progress to be monitored mid-execution.

If the SQL query executes successfully, two tables are returned, a warning table and an output table, which are selected in Steps 5 and 6, respectively. However, if the `@SettingGlobalWarnings` parameter has been turned off, Step 5 is skipped and only the output table is returned.

The warning table contains a range of checks, which include data quality checks, checks for invalid inputs, as well as SPC use as determined by the settings chosen. These checks are listed at the top of Steps 2 and 3.

The output table contains the data that was inserted into the SQL query, along with all the other columns necessary to populate the charts in the Power BI template and Excel tool.

If the SQL query is executed multiple times and changes made in Step 3 are not flowing through, execute the lines in Step 1 or Step 7 to clear the temporary tables.

What is done with this output table, and whether the SQL query needs modifying further, depends on whether the Power BI template or Excel tool is used to display the charts.

We will begin with the Power BI template.

Power BI template

The Power BI template is used to create a report, by modifying and building upon the pages included, to then publish, or for ad hoc charts.

The Power BI template comes preloaded with the sample data returned by the SQL query. The data source needs to be updated before it can be refreshed, and your data loaded.

Connecting to SQL Server

There are two methods of connecting to the SQL query, through a table or stored procedure. The former loads in static data very quickly, while the latter loads in live data but more slowly.

If connecting through a table, update Step 6 of the SQL query by uncommenting the INTO line and replacing DatabaseName, SchemaName, and TableName with the destination to store the output table in SQL Server. Then execute the query to store the table. If this query is going to be run multiple times, you may wish to add a line beforehand to drop the table first if it exists.

If connecting through a stored procedure, store the SQL query as a stored procedure and prevent the first table, the warning table, from being returned by turning off the @SettingGlobalWarnings parameter or storing it in a table in SQL Server.

Update the following parameters (*Home > Transform data > Edit parameters*):

Parameter	Change to
<i>SQLConnectionMethod</i>	<i>Table or Stored procedure.</i>
<i>SQLServerName</i>	Match the destination of the stored output table or stored procedure.
<i>SQLDatabaseName</i>	
<i>SQLSchemaName</i>	
<i>SQLTableOrStoredProcedureName</i>	

Then apply changes, which may require permission to run a native database query. This query simply selects all the data from the table or executes the stored procedure.

Once the data has refreshed, some of the pages in the Power BI template will need updating.

Pages

The pages included in the Power BI template and the changes to make are:

Page	Change
<i>Title</i>	Requires no change. Can be deleted, but it is advised you hide instead to keep the link to check for updates and the change log, which will be helpful if technical support is needed.
<i>Icon Summary Table</i>	Updates automatically, although you may wish to resize the <i>Metric</i> and <i>Group</i> slicers and columns in the summary table.
<i>Icon Summary Matrix</i>	
<i>Icon Matrix</i>	
<i>SPC Chart</i>	Update the <i>Group</i> dropdown first by temporarily disabling single and deselecting <i>TOTAL</i> and then select from the <i>Metric</i> dropdown before selecting from the <i>Group</i> dropdown.
<i>SPC Chart (w/ mR)</i>	

4 SPC Charts	These pages will all appear blank (unless the same <i>MetricID</i> and <i>Group</i> columns are used as in the sample data). Update which charts appear by modifying the <i>ChartFilters</i> table (<i>Home > Transform data</i>). Select the <i>ChartFilters</i> query in the <i>Queries</i> page and then click the cog next to first applied step, <i>Source</i> , in the <i>Query Settings</i> pane. Update this table as detailed below.
4 SPC Charts (w/ mR)	
7 SPC Charts + Icon Table	
9 SPC Charts	
Icon Descriptions	Requires no change, but the text can be replaced directly.
Icon Tooltip	Requires no change and remains hidden. These pages may look a mess but when viewed as tooltips, only data for a single chart are used.
Chart Tooltip	

If you do not need pages with multiple charts, you can hide them and skip the next section.charts.

Customising pages

As mentioned above, the *ChartFilters* query needs to be updated for the charts to appear on the pages with multiple charts, but first, go to a page and open the *Selection* page (*View > Selection*).

You will see a group called *SPCChart* for each chart, with a number suffix from 1 to 4, 7, or 9. The visuals contained within the group depends on the page.

Note that the charts on the *7 SPC Charts + Icon Table* and *9 SPC Charts* pages are smaller than the rest used throughout the Power BI template.

If you wish to change the number of charts that appear, it is advised you duplicate the page first. To remove a chart, delete a group from the *Selection* page. To add a chart, copy a group, rename it with a new number suffix, and then update the *ChartNumber* filter on each visual within the group. This may require the *ChartFilters* query to be updated first.

In addition to each visual in the group being filtered on *ChartNumber*, each of these pages have a page filter on *PageID*. The *SPC Chart* and *SPC Chart (w/ mR)* pages also have this page filter.

The columns to update in the *ChartFilters* query are:

Column	Purpose
<i>PageID</i>	Filters the <i>...SPC Chart...</i> pages. The same <i>PageID</i> can be used on multiple pages. The <i>SPC Chart</i> and <i>SPC Chart (w/ mR)</i> pages are filtered on <i>PageID</i> of <i>All</i> , which has all other columns blank to return all charts.
<i>ChartNumber</i>	Filters each of the visuals in a group on the multiple chart pages. For additional charts to appear on the <i>SPC Chart</i> and <i>SPC Chart (w/ mR)</i> pages or in an icon table, leave blank.
<i>MetricID</i>	Filter to a single or multiple charts. If both are added, a single chart is returned; if neither are added, all charts are returned; and if only one is added, all charts with the same <i>MetricID</i> or <i>GroupName</i> are returned. For the individual charts on the multiple chart pages, a single chart must be returned, or a slicer or filter added.
<i>GroupName</i>	

Additional columns

If additional columns are added to the output table in Step 6 of the SQL query, they will not automatically appear in the data model of the Power BI template.

Open *Power Query Editor (Home > Transform data)* and select the *MetricData* query or *ChartData* query in the *Queries* page. Then select the *Selected Columns* applied step in the *Query Settings* pane and amend the line in the formula bar to include the additional columns

Now that we have created a report, we might want to customise the charts.

Customising charts

All the charts can be customised at the same time by changing the following parameters:

Parameter	Purpose
<i>SettingChartTitleFormat</i>	Determines whether the group name is included in the chart title and how. If there is no chart title, <i>MetricName</i> is used.
<i>SettingAnnotationMarker</i>	Determines whether all points containing an annotation are circled.
<i>SettingBaseline</i>	Determines whether baselines are shown by a solid line.
<i>SettingOneTwoSigmaLines</i>	Determines whether the one- and two- sigma lines are shown on the X chart.

Additional information on summary icons and chart axes follow, but the report is now ready to be published (*Home > Publish*).

Summary icons

There are two ways in which summary icons are loaded into the report. On the *Icon Descriptions* page, they have been inserted individually, so will always appear. However, elsewhere in the Power BI template, they are loaded from the internet, depending on what chart they refer to. Therefore, an internet connection is required for them to load.

The online icons are stored on GitHub. The image URLs can be found in the *AssuranceIcons* and *VariationIcons* tables. If these cannot load because access to GitHub is denied, the icons can be stored locally, for example on a SharePoint page, and the links updated.

Chart axes

The axes of the main chart are determined by *MinChartAxis* and *MaxChartAxis* in the *ChartData* table. Similarly, the axes of the mR chart is determined by *MinmRChartAxis* and *MaxmRChartAxis* in the *mRChartData* table.

These all determine the range needed to fit the data and then a little more is added. This is done for two reasons. The axis minimum is decreased so that the markers appear fully, for example when at zero; the axis maximum is increased so that there is room for the summary icons.

Excel tool

The Excel tool is used to view individual SPC charts. It comes preloaded with the sample data returned by the SQL query.

The data can be copied directly from the output table returned by Step 6 of the SQL query and pasted into the table on the *Data* worksheet. Otherwise, a connection can be made to SQL Server.

Connecting to SQL Server

There are two methods of connecting to the SQL query, through a table or stored procedure. The former loads in static data very quickly, while the latter loads in live data but more slowly.

If connecting through a table, update Step 6 of the SQL query by uncommenting the INTO line and replacing DatabaseName, SchemaName, and TableName with the destination to store the output table in SQL Server. Then execute the query to store the table. If this query is going to be run multiple times, you may wish to add a line above to drop the table first if it exists.

If connecting through a stored procedure, store the SQL query as a stored procedure and prevent the first table, the warning table, from being returned by turning off the @SettingGlobalWarnings parameter or storing it in a table in SQL Server.

Add a new data source (*Data > Get Data > From Database > From SQL Server Database*), populate the fields, and add one of the following SQL statements under *Advanced options*:

- `SELECT * FROM [SchemaName].[TableName]`
- `EXEC <stored procedure>`

depending on whether connecting through a table or stored procedure.

Load the data in without transforming it, which will create a table on a new worksheet. You may wish to rename this table (*Table Design > Table Name*) and worksheet.

To switch to this new table, references to the existing table, *T_Data*, need to be updated by following these steps:

1. Go to the *Chart* worksheet.
2. Unprotect the worksheet (*Review > Unprotect*).
3. Use *Find & Replace* (*Home > Find & Select > Replace...*) to find *T_Data* and replace all instances with the name of the new table.
4. Open *Name Manager* (*Formulas > Name Manager*).
5. Replace all instances of *T_Data* with the name of the new table for *AssuranceIcon* and *VariationIcon*.
6. Add headings (*View > Headings*).

7. Unhide all columns.
8. For both PivotTables in *Columns AN* and beyond, update the data source to the new table (click on PivotTable > *PivotTable Analyze* > *Change Data Source*)
9. Hide *Columns P* and beyond.
10. Remove headings (*View* > *Headings*).
11. Open Visual Basic (*Alt + F11*).
12. Replace all instances (4) of *T_Data* with the name of the new table in the *ChartType* line in the *M_Updateformat* sub in *Module 1*.
13. Click the *Update Metric* button.

The *Data* worksheet can then be deleted.

Additional columns

If additional columns are added to the output table in Step 6 of the SQL query, they will be included in the table on the *Data* worksheet automatically whether copied and pasted over or loaded through a SQL Server connection.

Making changes

The Excel tool used hard-coded references throughout. These are listed below so that they can be updated if the workbook is changed:

Worksheet	Cells	Referenced
<i>Title</i>	All	<i>Workbook_Open</i> sub
<i>Chart</i>	All	<i>Workbook_Open</i> sub
	<i>B7:H36</i>	Conditional formatting
	<i>B37</i>	<i>M_SummaryRowFirstPage</i> sub <i>M_SummaryRowLastPage</i> sub <i>M_SummaryRowNextPage</i> sub <i>M_SummaryRowNextRow</i> sub <i>M_SummaryRowPreviousPage</i> sub <i>M_SummaryRowPreviousRow</i> sub <i>M_UpdateIconColours</i> sub <i>M_UpdateMetric</i> sub <i>Worksheet_Change</i> sub
	<i>C2 (R2C3)</i>	<i>M_UpdateMetric</i> sub
	<i>C4</i>	<i>AssuranceIcon</i> named range <i>M_UpdateFormat</i> sub <i>M_UpdateMetric</i> sub <i>VariationIcon</i> named range
<i>C38</i>	<i>M_SummaryRowLastPage</i> sub <i>M_SummaryRowNextPage</i> sub <i>M_SummaryRowNextRow</i> sub	

	<i>M_UpdateIconColours</i> sub <i>Worksheet_Change</i> sub
<i>E7:G36</i>	<i>M_UpdateFormat</i> sub
<i>G2</i>	<i>M_ExportTopChart</i> sub <i>M_ExportFullChart</i> sub <i>M_UpdateFormat</i> sub <i>M_UpdateMetric</i> sub
<i>H7:H36 (R7–36C8)</i>	<i>M_CreateIcons</i> sub <i>VariationIconSummaryRow7–36</i> named range
<i>I7:I36 (R7–36C9)</i>	<i>M_CreateIcons</i> sub <i>AssuranceIconSummaryRow7–36</i> named range
<i>N2</i>	<i>AssuranceIcon</i> named range <i>M_UpdateFormat</i> sub <i>M_UpdateMetric</i> sub <i>VariationIcon</i> named range
<i>N3</i>	<i>M_ResetFormat</i> sub
<i>N5</i>	<i>M_ResetFormat</i> sub
<i>N6</i>	Conditional formatting <i>M_ResetFormat</i> sub
<i>N9</i>	<i>M_ResetFormat</i> sub
<i>N15</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N16</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N17</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N19</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N20:N22</i>	<i>M_UpdateFormat</i> sub
<i>N20</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N21</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N22</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N23</i>	Conditional formatting <i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N25</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N26:N27</i>	Conditional formatting
<i>N26</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N27</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N29</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
<i>N30</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub

	<i>N32</i>	<i>M_ResetFormat</i> sub <i>M_UpdateFormat</i> sub
	<i>R3</i>	<i>M_UpdateMetric</i> sub <i>Workbook_Open</i> sub
	<i>AN3:AN1048576</i>	<i>M_SummaryRowLastPage</i> sub <i>M_SummaryRowNextPage</i> sub <i>M_SummaryRowNextRow</i> sub <i>M_UpdateIconColours</i> sub
	<i>AO3:AO1048576</i>	<i>MetricID</i> named range
	<i>AP3:AP1048576</i>	<i>DD_Metric</i> named range <i>M_UpdateMetric</i> sub
	<i>AR5:AR1048576</i>	<i>M_SummaryRowLastPage</i> sub <i>M_SummaryRowNextPage</i> sub <i>M_SummaryRowNextRow</i> sub <i>M_UpdateIconColours</i> sub
	<i>AS5:AS1048576</i>	<i>DD_Group</i> named range
	<i>AS5</i>	<i>M_UpdateMetric</i> sub
<i>SPC Icons</i>	<i>B2</i>	<i>VariationIconConcernHigh</i> named range
	<i>C2</i>	<i>VariationIconConcernLow</i> named range
	<i>D2</i>	<i>VariationIconImprovementHigh</i> named range
	<i>E2</i>	<i>VariationIconImprovementLow</i> named range
	<i>F2</i>	<i>VariationIconNeitherHigh</i> named range
	<i>G2</i>	<i>VariationIconNeitherLow</i> named range
	<i>H2</i>	<i>VariationIconCommonCause</i> named range
	<i>I2</i>	<i>VariationIconEmpty</i> named range
	<i>K2</i>	<i>AssuranceIconPass</i> named range
	<i>L2</i>	<i>AssuranceIconHitOrMiss</i> named range
	<i>M2</i>	<i>AssuranceIconFail</i> named range
	<i>N2</i>	<i>AssuranceIconEmpty</i> named range