ALL ABOUT YOU!

8TH ANNUAL PERSPECTIVE USERS' CONFERENCE
MARCH 1- 2 , 2016

PERSPECTIVE:
powered by **RESOLVER**

Report Building Considerations

- While the time to build a report can seem considerable, keep in mind that reports are designed to be templates. You build it once and execute it any time you need to run it.
- Execute your report a couple of times during the building process. The Preview cannot show true, real data, as the AE query must be run in order to populate the data into the report. This is not done until the report is executed.

Naming Objects & Elements

Report Objects (text, pictures, etc.) and Report Container Elements (tables, crosstabs, and charts) will always be named generically (e.g. Text, Picture, Chart, Table, etc.). When you have more than one of the same object or element within your report, re-name them for easier identification.

- **To rename text or pictures:** Click the **Objects tab**, single-click on the object twice (not double-click) then type the new name.
- **To rename tables, charts, or crosstabs**: Click the **Report Structure** tab, single-click on the element twice (not double-click), then type the new name.

Filters

The data returned by your Analysis Expert query or queries may already be filtered by criteria, but you can further filter this data within the report itself.

For example, your query pulls one year's worth of data, but you want a table, crosstab, or chart to display data only for January.

1. Go to the **Report Structure** tab then select the element (table, crosstab, or chart).
2. In the **Properties** window, go to the **Filter(s)** field and select formula.
3. Enter the appropriate formula (e.g. Month(<Reported Date/Time field>)=1) for all Incidents where the reported month equals 1 (January)).

Copy & Paste

Your report may require multiple tables, crosstabs, or charts that will have the same or similar look, layout, and content. Build the element, copy and paste it, then edit the pasted version to update the properties.

For example, your query pulls one year's worth of data and you require two tables displaying data from different months.

1. Build table 1 with the appropriate layout, content, formatting, etc. then add a filter (as described above).
2. From the **Report Structure** tab toolbar, copy then paste the table.
3. In the **Properties** window, edit the filter on the pasted table for the next month
4. Rename the pasted table (as described above), as the two tables will have the same name.

Copy and paste functionality also exists within Tables. When in the **Table Contents** window, all **Header**, **Data**, **Footer**, **Group Header**, and **Group Footer** lines may be copied and pasted if will share similar properties with others.

Multi-Column Reports

You can create reports with more than one column so tables, crosstabs, or charts appear side by side.

1. Click the **Report Structure tab** then single-click the report container.
2. In the **Column Count** field, add the desired number of columns.
3. Click the **Report Structure** tab then single-click on each element in the report container.
4. From the **Column Count** field for each element, select:
   a. **1 column** for the element to appear in the full width of the Report Container; OR
   b. **X columns** for the element to appear in just one of the columns.

Functions

Functions are used within the Report Designer to perform calculations, convert data, and more. Functions consist of the **Function Name** followed by a data field selection between the brackets (e.g. **Function(**<Data Field>**)**). For certain outputs you may need to embed functions within each other (e.g. **Function1(Function2(**<Data Field>**))**).

1. **Now()** - Produces the current date (e.g. **Now()** results in **2/3/2016**) To include the time as well, select the appropriate format from the **Date Format** tab. For example, to print the current date/time (i.e. 2/3/2016 10:00:00): **Date$(Now(),**"%02d.%02m.%04y, %02H:%02i:%02s"

   Date$() converts the resulting date/time value to a string to be printed. *"%02d.%02m.%04y, %02H:%02i:%02s"* is the format for the date/time.
   TIP: You can edit the format as needed, for example:

   - *"%02d.%02m.%04y %02H:%02i %PM"* = day/month/year 00:00 PM (31/01/2016 01:30 PM)
   - *"%02m. %02d. %04y"* = month/day/year (01/31/2016)
   - *"%D, %M %02d, %04y"* = day name, month name, day #/year (Sunday, January 31, 2016)
   - *"%02m. %04y"* = month/year (01/2016)
   - *"%M %04y"* = month year (January 2016)

2. **AddYears(), AddMonths(), AddWeeks(), AddDays(), AddHours(), AddMinutes(), AddSeconds()** - Add or subtract (using negative numbers) years, months, hours, etc. from the date value. For example, to subtract 2 hours from the current time of 10am (i.e. **2/3/2016 8:00:00**): **Date$(Addhours(Now(),-2), "%d.%m.%y, %H:%02i:%02s").**
TIP: When printing, be aware that the time zone will be based on the server time where Perspective is installed. This may differ from your time zone, in the case of Hosted customers, or those with users in different time zones.

3. **Year(), Year$(), Quarter(), Month(), Month$(), Dow(), Dow$()**

   - **Year(), Quarter(), Month(), Dow()** returns the **Numerical** value of the date (e.g. **Month(**<Reported Date/Time>**)** – sample result: **3**).
   - **Year$(), Month$(), Dow$()** returns the **Name** of the date value (e.g. **Month$(**<Reported Date/Time>**)** – sample result: **March**).

   TIP: For crosstabs and charts specifically, using the numerical and string functions together can be useful. For example:

   - *Use the **Month()** or **Dow()** as a Row/Column grouping (crosstabs) or Coordinate Value (charts)*
   - *Use the **Month$()** or **Dow$()** as the Displayed Contents (crosstabs) or Coordinate Label (charts)*

   This combination uses the numerical value to define (and sort) the data, but prints the name. Using the string functions as the row/column groupings or coordinate values would still work, but data would be sorted alphabetically and would therefore appear out of order.

4. **Count(), Sum(), Avg()** - Calculates a value (e.g. **Count(**<Incident Number>**)** - sample result: **2128**).

5. **Page$(), TotalPages$()** - Calculates and prints the current page and total number of pages in a report respectively. For example, to print a page count in your report, add a text box and enter **"Page "+Page$()+" of "+TotalPages$()** – sample result: **Page 1 of 10**

   *TIP: Shadow pages will **not** be counted. Shadow pages are an option you'll see with crosstabs. If a crosstab table is too large to fit in the report container space, the overlapping data can be set to print on shadow pages.*

6. SetVar(), GetVar()

SetVar() is used to store a value for later use, and consists of a **Name** for the value and the **Value** itself. This can be important as the designer does not save or store data once it has been printed. For example, to store a count of incidents for 2015: **SetVar("2015",** **Count(**<Incident Number field>**))**

GetVar() is used to recall a stored value, and uses the **Name** to recall the value. This can be used in more complex aggregate calculations such as comparing one year to another or for use in other objects (text boxes, gauges, or data graphics, etc.) For example, to recall the count of incidents for 2015: **GetVar("2015")**

7. Cond() - Conditions can be useful for a variety of purposes, such as substituting data, highlighting data, filters, etc. Conditions require an Argument, a result if True, and a result if False.
For example, to prevent blank cells from appearing in a Crosstab because a user didn't enter the Category value in the Class rollup**:** In the **Displayed Contents** field of the "Category" cell of the crosstab, enter: **Cond(IsNullOrEmpty(**<Class Category field>**),** "Not Specified", <Class Category field>**)**

- *IsNullOrEmpty()* - Checks to see if the Category field is null or empty (no data)
- *"Not Specified"* - The text I want displayed if the Category is null or empty (True)
- *<Class Category field>* - Displays the Category value if the field is **not** null or empty (False)