ABSTRACT
The degree of customization required for different kinds of reports and analyses used in presentations, documents, and spreadsheets varies from one organization or department to another. Often the results needed can be achieved by using a point-and-click tool. If that is not possible, then a coded approach is required. The amount of syntax required increases depending on the SAS® procedure that you choose. However, to achieve complete flexibility, a new tool from SAS called the Report Writing Interface (RWI) is required. RWI, part of ODS, is new in SAS® 9.4. Using DATA step statements, report structures, currently not achievable with other coded method can be generated. In addition, RWI enables you to incorporate style customization anywhere. The syntax uses a common dot notation that programmers from multiple languages are familiar with. This seminar examines capabilities, examples, advantages and disadvantages of this latest reporting methodology.

INTRODUCTION
This presentation documents features and abilities of a new data step dot notation syntax called the Report Writing Interface or RWI for short. RWI can create reporting structures and incorporate customization features not available with any SAS existing reporting procedures.

SIMPLE REPORTING STRUCTURES

Fixed row and column cell/entity location. 3 cells per row. Uniform report structure

### COMPLEX REPORTING STRUCTURES

CAPABILITIES OF EXISTING REPORTING TOOLS

PROC PRINT

Display 4. Output generated with simple syntax from proc print.

The syntax required for print is simple, and straight forward. Limitations include sub-totals which cannot be customized. No summary analysis. Column width which cannot be adjusted.

PROC TABULATE

Display 5. Multi-level summary output created with Proc Tabulate.

If I need to create a summary report tabulate is my first choice. Multiple category and analysis variables can be included in the report structure. Virtually all cells in the report can be customized. Sub-Totals can be in any desired text.

PROC REPORT

Comparison Across Columns

Flexibility Across Rows.
Display 6. Detail or Summary output with Proc Report. Customizations illustrated here not possible with other reporting procs.

Proc report, because of compute blocks, can incorporate a limited amount of data step syntax. Comparisons across rows and columns, not possible with tabulate, can be accomplished here. Reports can be detail or summary reports. Sub-totals are completely customizable with multi-line executive summaries. General structure of the report, however, is fixed row and column like tabulate and print.

DATA STEP REPORTING

![Display 7. Data Step Reporting with file and put statements. Oldest reporting method in the SAS system.](image)

This method uses data step syntax like RWI, so report content can be placed anywhere independent of a fixed location. This is the oldest reporting method. The big limitation is the restriction to the ODS Listing destination only. Style customizations are not possible.

PURPOSE OF RWI

- Gives you the ability to create any structure, and apply any customization regardless of location.
- This reminds me of graphic annotations for reports.
- But! The code required is more voluminous and potentially complex.

RWI SYNTAX NOTATION AND FORM

EXPLANATION OF DOT NOTATION

```sas
data _null_;
declare odsout htest();
hetest.table_start();
if _n_ = 1 then do;
  htest.row_start(type:"Header");
  htest.format_cell(text:"SASHHELP.CARS",column_span:10);
  htest.row_end();
/*partial code */
run;
```

RWI syntax is a standard dot notation language. The first part of the two-level name is the name of the object **htest**. The second part **table_start** is the name of the method or action you want to apply to the
report. Methods have properties, which define the state or identity of the object. With the `format_cell` method the header will have a column span of ten cells.

**NESTED FORM OF THE SYNTAX**

```plaintext
rep2.table_start();
    rep2.head_start();
        rep2.row_start(type:"Header");
            rep2.format_cell(text:"Summary Analysis of &statename ");
        rep2.row_end();
    rep2.head_end();
rep2.table_end();
```

For many opening RWI object and method declarations, matching closing syntax is required.

**RWI SYNTAX TIPS**

**DETERMINATION OF COLUMNS REQUIRED:**

Display 8. To successfully align report structures, a determination of the maximum number of columns in some row is required.

To successfully manipulate RWI solutions, a determination must be made of the row that has the greatest number of cells.

**COLOR NOTATION VERSES EQUAL SIGNS:**

```plaintext
rep3.format_cell(data:Verbage3 , just:'L' , column_span:2 , inline_attr:"color=darkbrown");
```

Logically a name value pair assignment is made with an equal sign in many programming language notations. RWI use a colon for an assignment of a property. This takes some time to get use too.

**CONSTRUCTING SYNTAX OUTSIDE METHOD ASSIGNMENTS:**

```plaintext
text='----------'|put(grandtotpop,comma14. );
Colorvalue='color='||put(profit,traffic.); ☺
rep2.format_text(data:text, column_span:3, style_attr:"Color=traffic.") ☒
```
Constructing text programmatically and making an assignment in data step code separate from property assigns is often required to get the code to resolve correctly.

**CREATION AND RESLOVING MACRO VARIABLES**

```latex
else %let fmt=eurox10.;
htest.format_cell(data:put(msrp,&fmt, inline_attr:"color=green");
else call symputx('fmt','eurox10.);
htest.format_cell(data:put(msrp, symget(fmt)), inline_attr:"color=green");
```

Macro triggers % and & are not data step syntax, and will resolve prior to data step execution. This creates a timing issue which prevents dynamic construction of text strings. With data step macro routines CALL SYMPUTX or the SYMGET function must be used.

**EXTENDED CAPABILITIES OF RWI**

**HEADING STRUCTURES:**

Display 9. Normally data in the body of a report is aligned to the heading structure at the top with RWI the two are separate and independent.

Notice the circled area associated with Model, Acura and the Suggested Price value of $36,945 both have content under this area. Normally Acura would be the only value under model, but in this analysis the suggested price value starts in Model field. The body content and heading structure are independently defined in different areas of data step code.

**FOOTER STRUCTURES:**

Display 10. Dedicated footer methods in RWI allow any kind of complex or simple potentially multi-line footer structure to be placed at the bottom of the report.

Data step code of virtually any complexity and form can be used to create any desired footer section. The user is only limited by their creativity.
COLUMN FLEXIBILITY

Display 21. Column locations, widths, borders can be custom and completely varied from one row to the next.

Columns of data in a report body are not fixed. Notice in display 11 the green revenue information varies from one row to the next. RWI is the only location where this is possible (Sample syntax shown below.)

ROW FLEXIBILITY

Display 32. The thickness of a row from one to the next can be varied. Possibly a couple of words are required in one row, and three or four sentences for the next.

Notice the red row_span:2 property. This can be varied for any row/column combination.
FORMATTING CAPABILITIES

Display 43. Formatting can be varied not only across rows but potentially across columns as well.

Formatted values can be programatically changed from one row to the next, and frankly the syntax is easier, and application more consistent than with `proc report` using compute blocks embedded with call define.

```sas
if mod(I,2)=0 and I in (1,3) then call symputx('fmt','dollar8.');
if mod(I,2)=0 and I in (5) then call symputx('fmt','yen8.1');
else call symputx('fmt','eurox10.1');
htest.format_cell(data:put(msrp,symget(fmt)),inline_attr:"color=green fontweight=bold fontsize=6");
```

STYLE ENHANCEMENTS

Display 54. The appearance of text across any row or column combination can be completely ,.

Custom fonts, sizes, color, weight, style.

An extensive set of customizations can be applied using the Style Attribute List website. Independent of row or column locations. One with a row or column cell can be completely different than another.

Very Useful style list website.

[http://go.documentation.sas.com/?docsetId=odsug&docsetTarget=n0otdo2g12obp3n0zmnghcn7p4vu.htm&docsetVersion=9.4&locale=en](http://go.documentation.sas.com/?docsetId=odsug&docsetTarget=n0otdo2g12obp3n0zmnghcn7p4vu.htm&docsetVersion=9.4&locale=en)

MULTI-TASKING with RWI

Display 65. This basic report was generated entirely with RWI and Data step syntax. Merging of tables, using hash object to sort in memory, summarization and custom reporting were all handled in one step.

An equivalent solution employing other methods would require multiple processing steps. In each , inbound and outbound I/O segments would be expended. If a user were processing a large table with millions of records, only having one step as opposed to 5 or 6, would dramatically reduce the runtime of the overall process.
/**Declare a Hash Object **/
if _n_=1 then do;
   declare hash final(ordered:'descending',multidata:'Y');
   final.definekey('totpop','state','county');
   final.definedata('statename','countynm','zip','totpop');
   final.definedone();
   declare hiter f('final');
...
/**Later in the solution the work with Merging **/....
   merge work.cities end=d1
      work.zippy(keep=state county city zip countynm);
/** Then Summarizing with Hash Object **/
do until(rc ne 0);
   track+1;
   if track=1 then rc=f.first();
   grandtotpop+totpop;
   rc=f.next();
end;
/**Then Reporting **/
repl.format_cell(data:make, inline_attr:"color=yellow row_span:2");
repl.format_cell(data:put(msrp,euro8.), inline_attr:"color=green
tweight=Bold , row_span:2");

RWI AND MACRO

Display 16. The data step code using RWI can be complex and significant in terms of code required.
If the routine is converted into a SAS macro, simple left to right parameters make structure modification and content changes much easier.

**ANIMATIONS**

Display 77. Animation and Static images can be incorporated into RWI output. (This will be demoed during the presentation)

The presenter has developed routines, which will animate any Unicode character, or text string. Graphic animations embedded in an RWI report will also be displayed.

**GRAPHICS AND IMAGES**

Display 88. Multi-panel displays can be created with RWI incorporating SAS and other kinds of output. (Sample syntax shown below)

```sas
rep3.format_cell(inline_attr:"preimage='c:\public\Global Forum RWI\image1.jpg'"); rep3.format_cell(inline_attr:"preimage='c:\public\Global Forum RWI\strep25.png'" ); rep3.format_cell(inline_attr:"preimage='c:\public\Global Forum RWI\inside_the_box.png'" );
```

RWI and data step flexibility allows these elements to be incorporated independent of location.
Display 99. The previous analysis has been modified to include a multi-line report associated with a single image entity.

No reporting tool other than RWI can associate a single entity with a multi-cell entity.

**VIDEO AND AUDIO**

Display 20. If a user has SAS9.4M4 or above, video can be added to multi-panel displays using RWI and ODS layout gridded in combination. (This will be demoed during the presentation)

Video can only be utilized with RWI. Proc report, tabulate, and print can reference a URL with video content, but navigation away from the original website is required. In RWI the video displays within the gridded layout

**VIDEO AND AUDIO REQUIREMENTS:**

- HTML5 or later.
- SAS9.4M4 or later.
- For organized structures must use layout gridded RWI form.

**ODS DESTINATION FUNCTIONALITY:**

- RWI syntax can be used with ODS Mark Up (HTML4 and 5, MSOFFICE2K, EXCEL EXCELXP, etc.)

- RWI syntax can also be utilized with Printer destinations. PDF, PCL and PRINTER.
LIMITATION ENCOUNTERED

PROC DS2 AND RWI
- Ultimately at part of this presentation, I wanted to be able to multi-task using RWI and DS2 capabilities.
- Scenarios could then be created, where remote database data in a multi-threaded environment could be quickly summarized and then used with any report structure.
- I could not get RWI notation to be recognized from within proc ds2 syntax per SAS94M5

ODS Layout and RWI
- Layout methods available to RWI allow multiple pieces of output to be combined into a single display.
- But unlike ODS Layout, RWI does not allow the embedding of procedure syntax in the overall notation.
- Output from proc tabulate, and split cannot be easily combined with report structures from RWI.
- The overall notation is much easier to work with strictly from ODS Layout.

LAYOUT GRIDDED AND RWI:
- Video and Audio are displayed sequentially top to bottom with basic RWI syntax.
- To get this kind of media in a structured tabular form with elements side to side and top to bottom, Layout Gridded using RWI notation must be used.
- ODS Layout outside the RWI realm cannot display video and audio.

PROC DS2 AND RWI:
- Ultimately at part of this presentation, I wanted to be able to multi-task using RWI and DS2 capabilities.
- Scenarios could then be created, where remote database data in a multi-threaded environment could be quickly summarized and then used with any report structure.
- I could not get RWI notation to be recognized from within proc ds2 syntax per SAS94M5.

CONCLUSION

Standard uniform reporting structures can be generated with common SAS procedures like print, tabulate, and report. If a non-standard structure is required then a Report Writing Interface (RWI) solution will have to be implemented. Further, overall processing step requirements can be significantly reduced with RWI, dramatically improving the efficiency of the overall program. Style enhancements can also be applied to any cell. With this tool, a proficient/creative data step programmer has unlimited capabilities. RWI does have drawbacks, the volume and complexity of the code required increases with this methodology. To justify the additional development time required for a complex report structure, the form should be extensively utilized by the organization.
REFERENCES
GLOBAL FORUM PAPER SAS5762-2016 BY CYNTHIA ZENDER

USEFUL DOCUMENTATION

- Report Writing Interface: http://go.documentation.sas.com/?docsetId=odsadvug&docsetTarget=n0ys4rvjs0i2ukn0znadr0qtu2qs.htm&docsetVersion=9.4&locale=en

- Style Attribute List http://go.documentation.sas.com/?docsetId=odsug&docsetTarget=n0otdo2g12obp3n0zmnhcn7p4vu.htm&docsetVersion=9.4&locale=en

- A copy of all the programs are available upon request Ted.Durie@sas.com