



Leveraging Report Variables for More Robust SAP BusinessObjects Web Intelligence Reporting

David G. Rathbun

Originally presented at BI2012 Conference

Abstract



- This session explores the role of report variables in SAP BusinessObjects Web Intelligence 4.0 and offers undocumented tips and tricks to exploit them for more creative and efficient ad hoc reporting and data analysis. Walk through system demos to understand what's required to merge data providers in SAP BusinessObjects Web Intelligence, including tips to leverage the ForgeMerge() function to fix unbalanced data providers in a full client document. Obtain best practices for using SAP BusinessObjects Web Intelligence report functions and other context operations to enable calculations at different levels of granularity. Explore workarounds for displaying UserResponse() values on separate rows in a table, and see how this improves the user experience. Get techniques to optimize prompt handling, including insight into whether and how to create a prompt syntax that substitutes “today's date” for a prompt default value.

About Dave



- Dedicated to BusinessObjects solutions since 1995
 - Consultant and trainer for fifteen years
 - Currently BI Solutions Architect for PepsiCo
 - Note: Content is my own and does not reflect my employer
- 17 consecutive years presenting at major BI conferences
 - United States, Europe, Australia
- Charter member of BOB
 - <http://busobj.forumtopics.com>
- I Blog! Dave's Adventures in Business Intelligence
 - <http://www.dagira.com>
- SAP Mentor for 2009 – 2012



Demonstration Platform



- Demonstration universes
 - eFashion
 - Island Resorts Marketing
 - Prestige Motors
- Software configuration
 - BusinessObjects Enterprise XI 3.1
 - Oracle 10g
- BusinessObjects toolset
 - Web Intelligence Rich Client
 - Universe Designer



Demonstration slides will be highlighted with this icon

What We'll Cover ...



- Introduction to Variables
- Merging Data Providers
- Calculation Context
- UserResponse()
- Prompt Handling
- Wrap-up

Where Can I Do Calculations?



- ETL process
- Universe objects
- Report structure items
 - Variables
 - Formulas
 - Constants

ETL Process

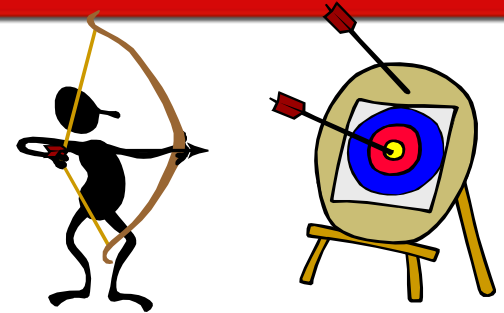


- Advantages
 - Build on core data
 - Can use procedural language
 - Ensure consistency across all data access paths
 - Calculate and store once, retrieve many times
- Disadvantages
 - Change management
 - Complexity
 - Impact analysis
- If a calculation is performed frequently and has high overhead consider pushing it to the ETL

Universe Objects



- Advantages
 - Build once – Use many times
 - Use full range of database functions
 - Ensure consistency from report to report
 - Updates automatically propagate
- Disadvantages
 - Limited to information from one universe
 - Maintenance required by universe designer
 - Some aggregation issues can be tricky
 - See the Database Delegated feature from XI 3.0 for more ideas
 - Some functionality might be missing from the database
 - If-Then-Else logic



Report Structure Items



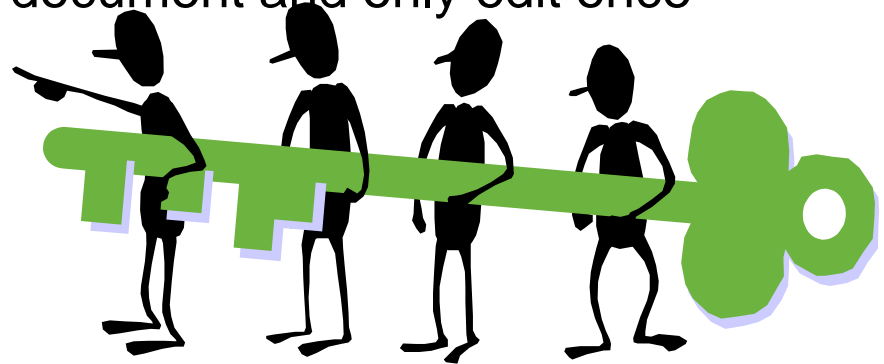
- Advantages
 - Available on all platforms
 - Independent of SQL restrictions
 - Calculations based on document data
- Disadvantages
 - Stored in a single document
 - Require some level of technical expertise
 - Volume of data could impact performance



Why Variables?



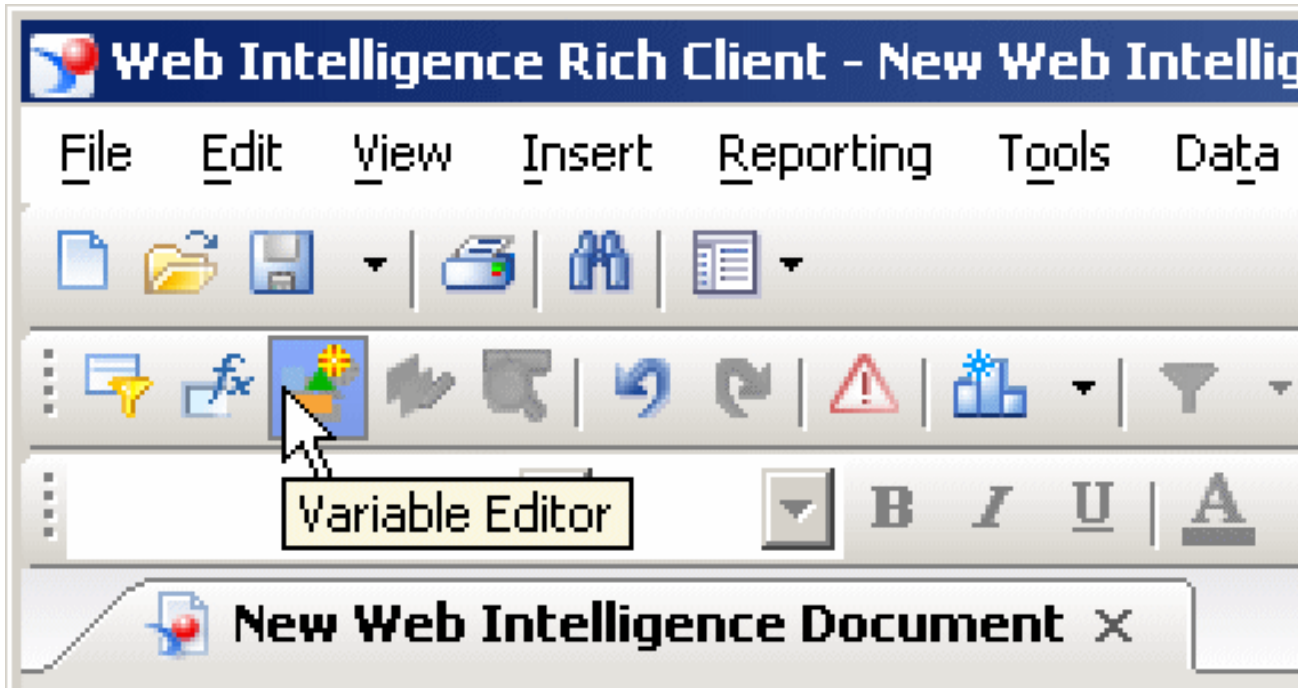
- Convenience
 - Use with all product features
- Clarity
 - A name defines a purpose
- Complexity
 - Complex items can be built in steps
- Consistency
 - Reuse variables throughout the document and only edit once



Building Variables



- Click the Variable Editor



- Alternative: **Data + Variables** from menu

What We'll Cover ...



- Introduction to Variables
- Merging Data Providers
- Calculation Context
- UserResponse()
- Prompt Handling
- Wrap-up

Merging Unbalanced Data Providers



- Web Intelligence allows multiple data providers
 - Common dimension objects are merged on the report
 - Report can contain common data from multiple sources
- What are unbalanced data providers and why are they a problem?

What Are Unbalanced Data Providers?

- Dimensions are the key values for a data provider
- Just as keys are joined between database tables, dimensions are linked (merged) between data providers
- When the number of dimensions from two data providers are not equal, they are unbalanced

Country	Resort	Revenue
France	French Riviera	\$835,420.00
US	Bahamas Beach	\$971,444.00
US	Hawaiian Club	\$1,479,660.00

Resort	Number of guests
Bahamas Beach	565
French Riviera	446
Hawaiian Club	540

- In this example
 - Resort is common to both data providers and merged
 - Country is unbalanced because it has no match

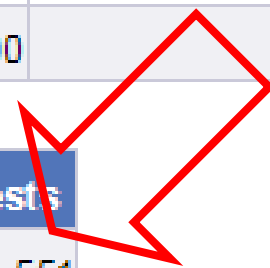
Unbalanced Data Provider Impact



- Unlinked dimensions cause problems in a report block
 - Measures are out of context for those dimension values
 - Total values are displayed for each dimension value

Country	Resort	Revenue	Number of guests
France	French Riviera	\$835,420.00	446
US	Bahamas Beach	\$971,444.00	565
US	Hawaiian Club	\$1,479,660.00	540

Country	Revenue	Number of guests
France	\$835,420.00	1,551
US	\$2,451,104.00	1,551



🖥️ Demonstration 1 – Create common block from unbalanced data providers



Using ForceMerge()

- The ForceMerge() function can fix unbalanced data issues
 - Requires Web Intelligence to consider all merged dimensions rather than just those participating in the block

=ForceMerge ([Number of guests])

Country	Revenue	Guests
France	\$835,420.00	446
US	\$2,451,104.00	1,105

 Demonstration 2 – Using ForceMerge() to fix unbalanced data providers

ForceMerge() Does Not Fix Everything

- Consider the following two unbalanced data providers

Year	Quarter	Number of guests
FY2005	Q3	36
FY2005	Q4	130
FY2006	Q1	132
FY2006	Q2	139
FY2006	Q3	116
FY2006	Q4	135

Quarter	Number of guests
Q1	394
Q2	410
Q3	357
Q4	390

- Quarters roll up to years ...
- ... but the data from block #2 has already been summarized by quarter
- Required data is simply not available

 Demonstration 3 – Unfixable unbalanced data providers

Merging Unbalanced Data Providers



- Unbalanced data providers occur when
 - Two or more data providers exist in a document
 - Common dimensions are linked
 - Extra dimensions are present on one (or both) sides
- Impact of unbalanced data providers
 - Measures project to the higher context and repeat for each dimension value
- Fixing unbalanced data providers
 - ForceMerge() or MultiCube() can fix some issues
 - Data must be able to “project up” a hierarchy
 - Data cannot be “broken down” to a lower level of detail

What We'll Cover ...



- Introduction to Variables
- Merging Data Providers
- Calculation Context
- UserResponse()
- Prompt Handling
- Wrap-up

What Is Calculation Context?



- Web Intelligence uses a cube architecture behind the scenes
 - Dimensions define the scope or context of the cube
 - Measures project to the selected context
- Calculations can set a different context
 - Dimensions can be explicitly included or excluded
 - Different measure values can be obtained
- Generally this is done for specific reasons
 - Percentage of total
 - Subtotals
 - Maximum over a range

Calculation Context Scenario



- We need to identify the best month for all years
 - Select the maximum monthly value across all years available

=Max ([Revenue] In ([Year] ; [Month]))

In Report

- Why does this work?

Year	Month	Revenue	Best Month
FY2008	AUG	\$108,176.00	\$186,611.00
FY2008	JUL	\$69,850.00	\$186,611.00
FY2008	JUN	\$30,378.00	\$186,611.00
FY2008	MAY	\$186,611.00	\$186,611.00
FY2008	APR	\$79,620.00	\$186,611.00
FY2008	MAR	\$43,638.00	\$186,611.00
FY2008	FEB	\$180,118.00	\$186,611.00
FY2008	JAN	\$66,115.00	\$186,611.00
FY2007	DEC	\$28,418.00	\$186,611.00
FY2007	NOV	\$167,252.00	\$186,611.00

What Makes This Work?



- A calculation can have two contexts
 - Input
 - Output
- Running calculations may include a third type of context
 - Reset
- Input context determines what is to be considered during the calculation
- Output context determines how far the results are projected

Let's Dissect the Context



- First, the **input** context tells us to calculate the maximum based on Year and Month
 - 3 years each with 12 months generates 36 **input** values
- Next, the **output** context In Report tells us to take the biggest or Max() of all 36 values

**=Max ([Revenue] In ([Year] ; [Month])
In Report)**

- What if the input context had been In ([Year]) by itself?
- In ([Month]) by itself?

Other Input Context Options — In Year

- An input context of [Year] would have provided only four possible input selections
 - The total revenue would have been projected to the Year level
 - The maximum of those four would have been output at the report level

Year	Revenue
FY2005	\$357,352.00
FY2006	\$1,077,118.00
FY2007	\$1,087,548.00
FY2008	\$764,506.00

Other Input Context Options — In Month



- An input context of [Month] would have provided only 12 possible input selections
 - Values would have included May for all 3 years
 - The maximum of those 12 values would have been output at the report level

Month	Revenue
JAN	\$201,615.00
FEB	\$544,128.00
MAR	\$111,095.00
APR	\$226,020.00
MAY	\$555,628.00
JUN	\$99,374.00
JUL	\$183,330.00
AUG	\$469,930.00
SEP	\$83,615.00
OCT	\$199,775.00
NOV	\$522,674.00
DEC	\$89,340.00

- Note: Dimensions used in an input context do not have to appear in the block

Calculation Context Options



- These slides barely scratch the surface of calculation context!
- There are multiple context operators
 - In Specifically include a dimension (or list)
 - ForEach Use only the specified dimensions
 - ForAll Ignore the specified dimension list
- There are multiple context key words
 - Body Lowest context level (by row/column)
 - Block By section break
 - Report Overall report

Calculation Context Recap



- Web Intelligence uses a cube architecture behind the scenes
 - Dimensions define the scope or context of the cube
 - Measures project to the selected context
- Calculations can set a different context
 - Dimensions can be explicitly included or excluded
 - Different measure values can be obtained
- Generally this is done for specific reasons
 - Percentage of total
 - Subtotals
 - Maximum over a range

What We'll Cover ...



- Introduction to Variables
- Merging Data Providers
- Calculation Context
- `UserResponse()`
- Prompt Handling
- Wrap-up

Capturing and Displaying User Prompt Selections

- Many reports will have prompts to make them more flexible
- To preserve the context of the query, prompt selections should be shown on the report
- Web Intelligence provides the `UserResponse()` function to perform this task

`UserResponse()` Returns the response to a prompt

Syntax `UserResponse([dp;]prompt_string[;Index])`

UserResponse in Action



Prompts

Select or type the prompt values for each prompt below.

✓ Enter value(s) for Manufacturer: **Ferrari;Rolls Royce;Jaguar;Lotus**

Type a value

Refresh values

Manufacturer

Caterham

Subaru

Aston Martin

Enter value(s) for Manufacturer:

Ferrari

Rolls Royce

Jaguar

Lotus

Ferrari;Rolls Royce;Jaguar;Lotus

Manufacturer	Model Name
Ferrari	F335
Ferrari	F355
Ferrari	F456

What About a Different Format?



- By default UserResponse() returns a single string
- With some additional string functions the format can be changed
- Replace() substitutes one string value for another
- Char() returns a specific character from the ASCII set
 - Char(13) returns a carriage return
 - Char(10) returns a line feed

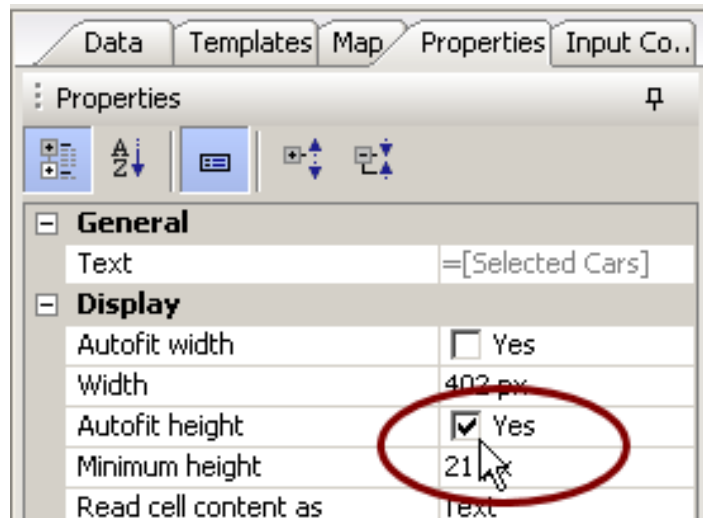
```
=Replace (UserResponse (  
"Enter value(s) for Manufacturer:");  
";" ;Char (13)+Char (10) )
```

 Demonstration 5 – Processing results of UserResponse() function

UserResponse Results



- By replacing the embedded ; with some printer control characters the selected car names now appear on separate lines
 - Be sure to tick the Autofit Height checkbox to ensure all values are visible



Ferrari
Rolls Royce
Jaguar
Lotus

Manufacturer	Model Name
Ferrari	F335
Ferrari	F355
Ferrari	F456

UserResponse Wrap-Up



- UserResponse() is used to capture a user selection or entry for a prompt
- Various other functions can process that value if needed
- What sort of options can be configured on the prompt itself?

What We'll Cover ...

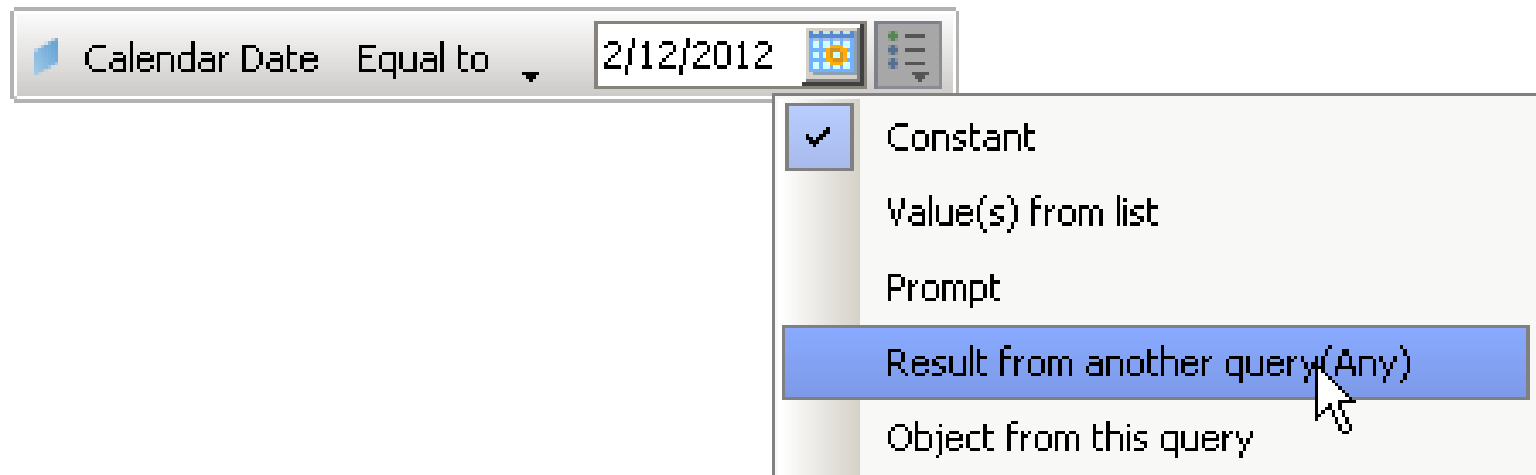


- Introduction to Variables
- Merging Data Providers
- Calculation Context
- UserResponse()
- Prompt Handling
- Wrap-up

Query Condition Choices



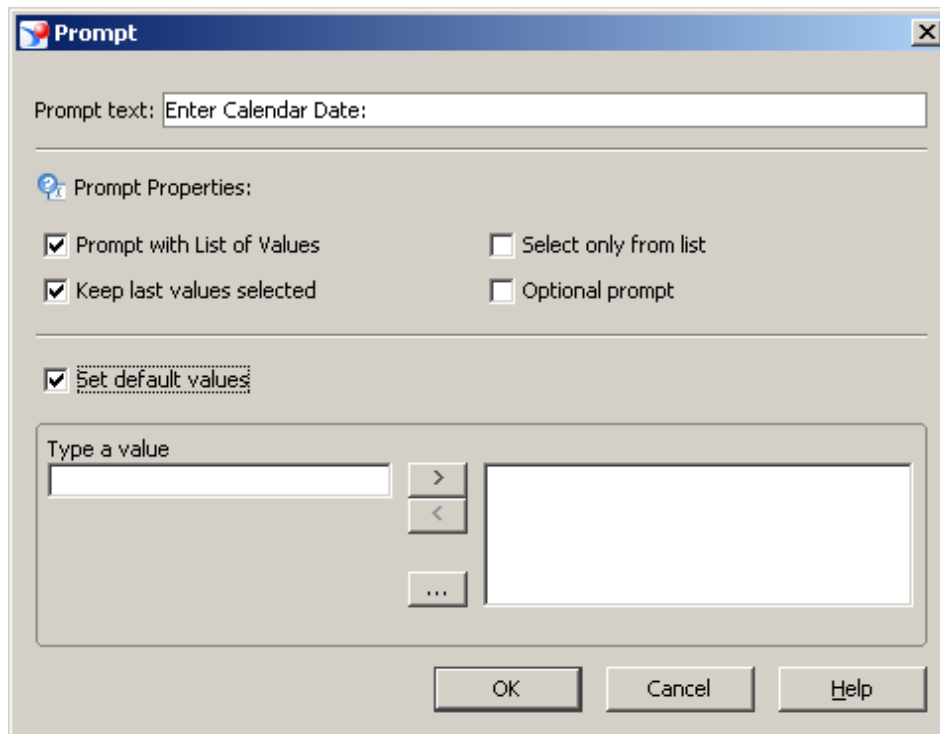
- Prompts can be defined in the report or in the universe
 - Report prompts have more options
 - Universe prompts can be reused for consistency
- What are the “extra” report prompt choices?



Prompt Options



- Once a condition is defined as a prompt these options can be set



- Notice what's not there? There is no option for a calculated prompt default value ...

Enabling “Today” As Default Value for Date Prompts



- There is no option to calculate a default prompt value
- What I want:
 - Use “Today” as a prompt value
 - Change the value to some other date if needed
- What I get:
 - Select a “magic date” value to substitute for today’s date
 - A standard date prompt which accepts any date input and preserves the calendar interface
- How it’s done:
 - Create a special prompt object in the universe

 Demonstration 6 – Building and using a “today” prompt object

Today Prompt Object



- Pick a date value to become the “magic” date
 - Typically use 1/1/1900
- Check for that input value in the date prompt logic and replace it with the today date if found

```
CAL.CALENDAR_DATE =  
case to_date(@Prompt('Enter date','D',,mono,free))  
when to_date('01/01/1900','MM/DD/YYYY')  
then trunc(sysdate)  
else to_date(@Prompt('Enter date','D',,mono,free))  
end
```

Calculated Default Prompt Option



- This really isn't what I want but it works
- Date prompt preserves calendar widget
- Reports can be scheduled with 1/1/1900 and always run for today
- Some times we can't get exactly what we want but we can get what we need

What We'll Cover ...



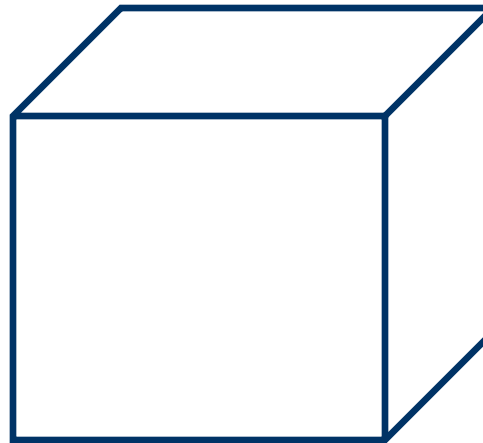
- Introduction to Variables
- Merging Data Providers
- Calculation Context
- UserResponse()
- Prompt Handling
- Wrap-up

Wrapping It Up



- Variables extend the power of SAP BusinessObjects
- Be creative, play with your functions!
- Don't be afraid to ...

Think



Additional Resources



- Many of the techniques addressed today have been covered in blog posts on my business intelligence blog
 - Dave's Adventures in Business Intelligence Blog
 - www.dagira.com
- There are several online communities that also offer help
 - BOB (BusinessObjects Board)
 - <http://busobj.forumtopics.com>
 - SAP Community Network Forums
 - <http://forums.sdn.sap.com/index.jspa>

7 Key Points to Take Home



- It can be more efficient to process data during ETL or as universe objects
- However, report variables have certain advantages over either of those solutions
- There is a wide variety of formula functions, have a look around!
- Calculation context concepts – once mastered – provide specific control over formula results
- ForceMerge() can fix some unbalanced data provider issues that occur when merging multiple data providers
- Prompt choices are captured with UserResponse() which can then be further processed
- While we don't have calculated default values for prompts, a solution can still be delivered ... with concessions from the users

Disclaimer



SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Wellesley Information Services is neither owned nor controlled by SAP.

Any opinions, comments, statements, or technical details provided in this presentation are my own personal opinions and may or may not reflect the opinions or policies of my employer or clients.