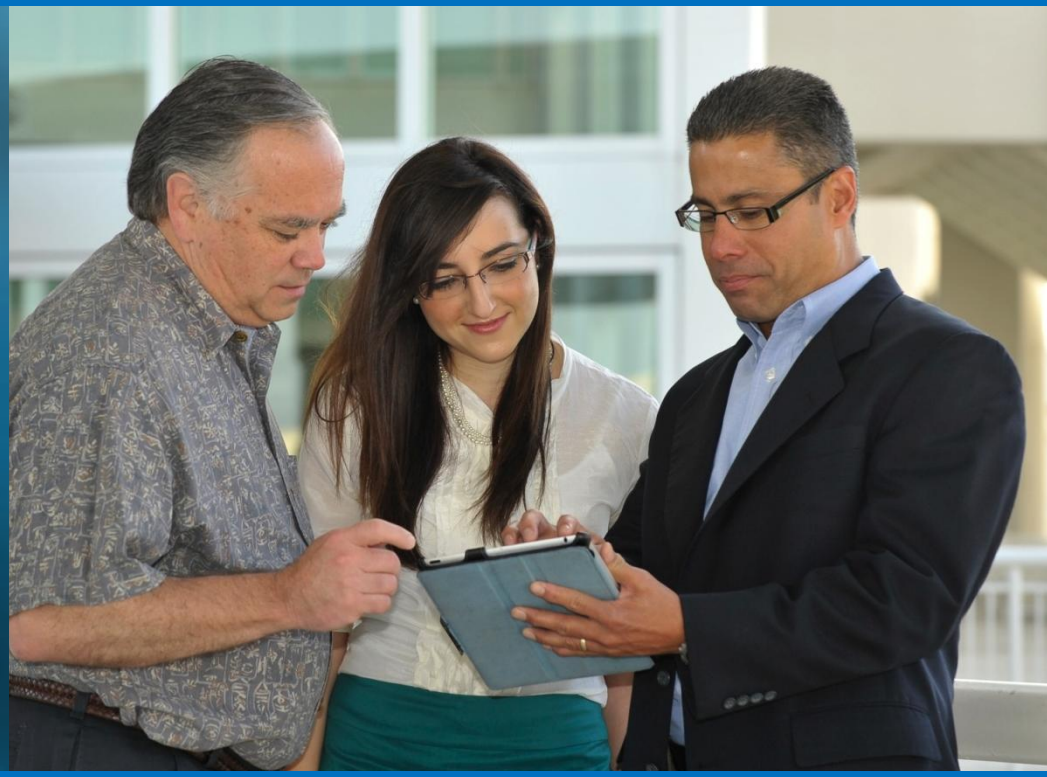
ASUG SAP BusinessObjects
USER CONFERENCE

September 9–11, 2013
Anaheim, California

Business Intelligence
for a PASSIONATE
COMMUNITY



Advanced Web Intelligence Techniques for Aspiring Jedi Knights

Alan Mayer – Solid Ground Technologies

Agenda

- **Introduction**

- Query Techniques
- Report Techniques
- Performance Considerations
- Testing
- Conclusion

Introduction

Alan Mayer

- Co-founded Integra Solutions in 1993
  - Used BusinessObjects since 1992 (Version 2.2)
  - Wrote the first BusinessObjects training manuals
  - Over 75 Fortune 1000 customers before company was sold in 2007
- Presented at every national conference
- Founded Solid Ground Technologies in 2009
  - Different company – same principles
  - Specializing in BusinessObjects consulting and training

Agenda

- Introduction
- **Query Techniques**
- Report Techniques
- Performance Considerations
- Testing
- Conclusion

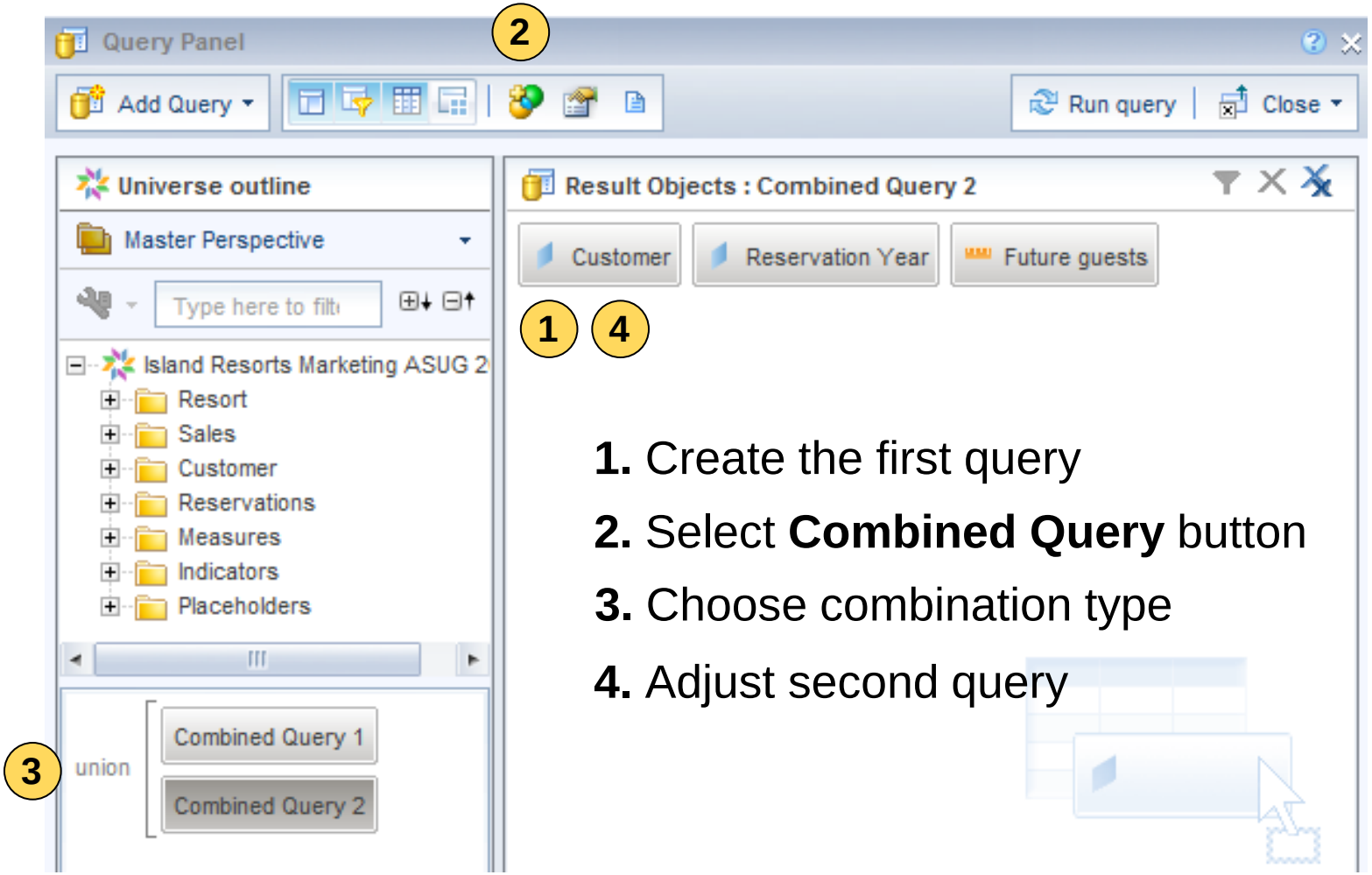
Query Techniques

- Several topics to discuss:
  - Combined queries (UNION / INTERSECT / MINUS)
  - Subqueries
  - Data providers with multi-SELECTs
  - Row & time restrictions

Combined Queries

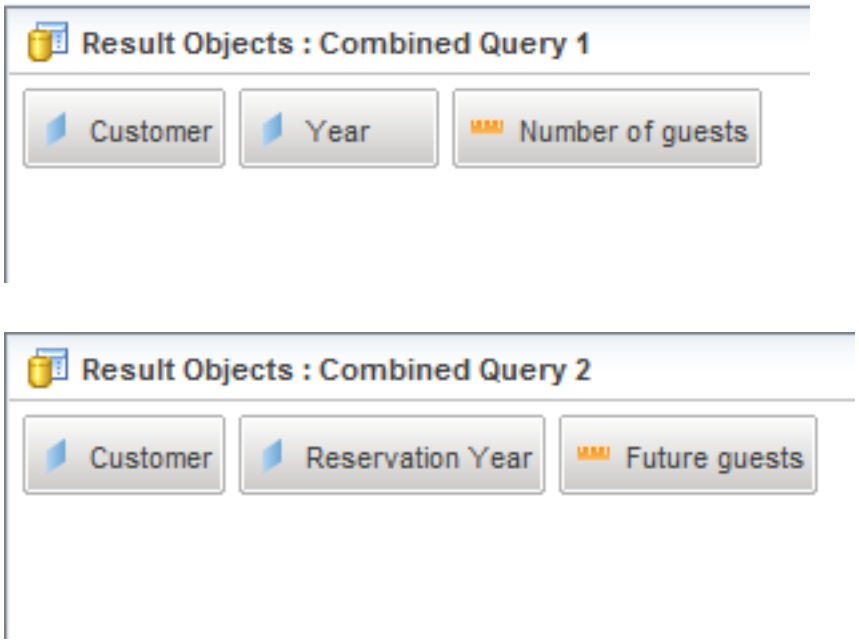
- UNION, INTERSECT and MINUS queries allowed
  - A few rules must be followed:
    - Same number of objects in each query
    - Same data type used in each position
  - Each operator gives you a different result:
    - UNION: Combines rows from two or more queries (SELECTs)
    - MINUS: Returns rows from the first query that aren't in the second
    - INTERSECT: Returns rows that are in both queries
- Can provide faster results if used properly

Creating Combined Queries



Union Queries

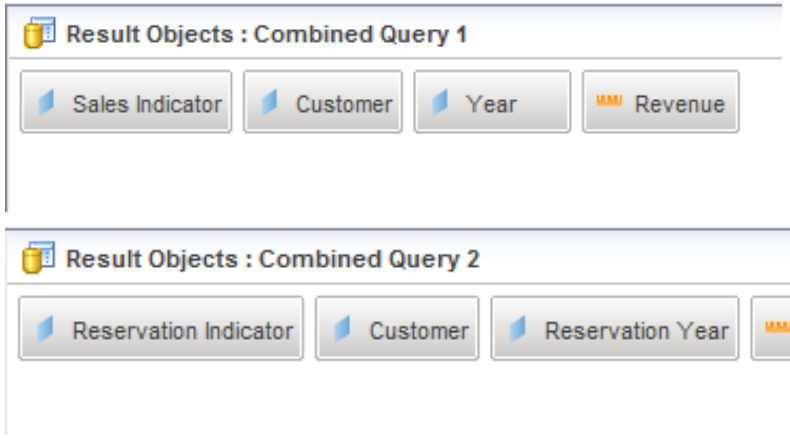
- Results from UNION queries are unclear
  - Can't determine which rows come from a query
  - Column headers from first query only



Customer	Year	Revenue
Arai	FY2004	8,036
Baker	FY2004	128,362
Baker	FY2005	150,666
Baker	FY2006	162,566
Baker	FY2007	8
Brendt	FY2004	8,420
Diemers	FY2004	10,976
Dupont	FY2007	4

Union Queries, cont'd

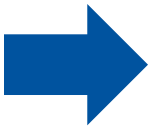
- Indicator objects can help clarify UNIONs
  - Requires a Universe Designer to create the objects
  - Used like descriptive tags



Customer	Sales Indicator	Year	Revenue
Arai	Sales	FY2004	8,036
Baker	Reservation	FY2007	8
Baker	Sales	FY2004	128,362
Baker	Sales	FY2005	150,666
Baker	Sales	FY2006	162,566
Brendt	Sales	FY2004	8,420
Diemers	Sales	FY2004	10,976
Dupont	Reservation	FY2007	4

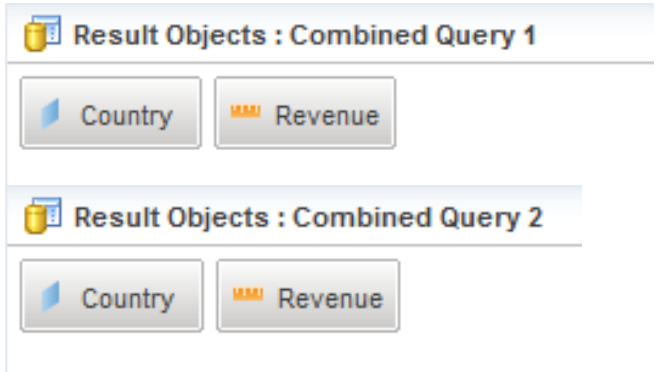
UNION Queries, cont'd

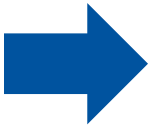
- UNION queries add one more challenge …
  - They eliminate duplicate rows before combining both queries
  - Result: Only DISTINCT rows returned



Single Query

Country	Revenue
France	835,420
US	2,451,104



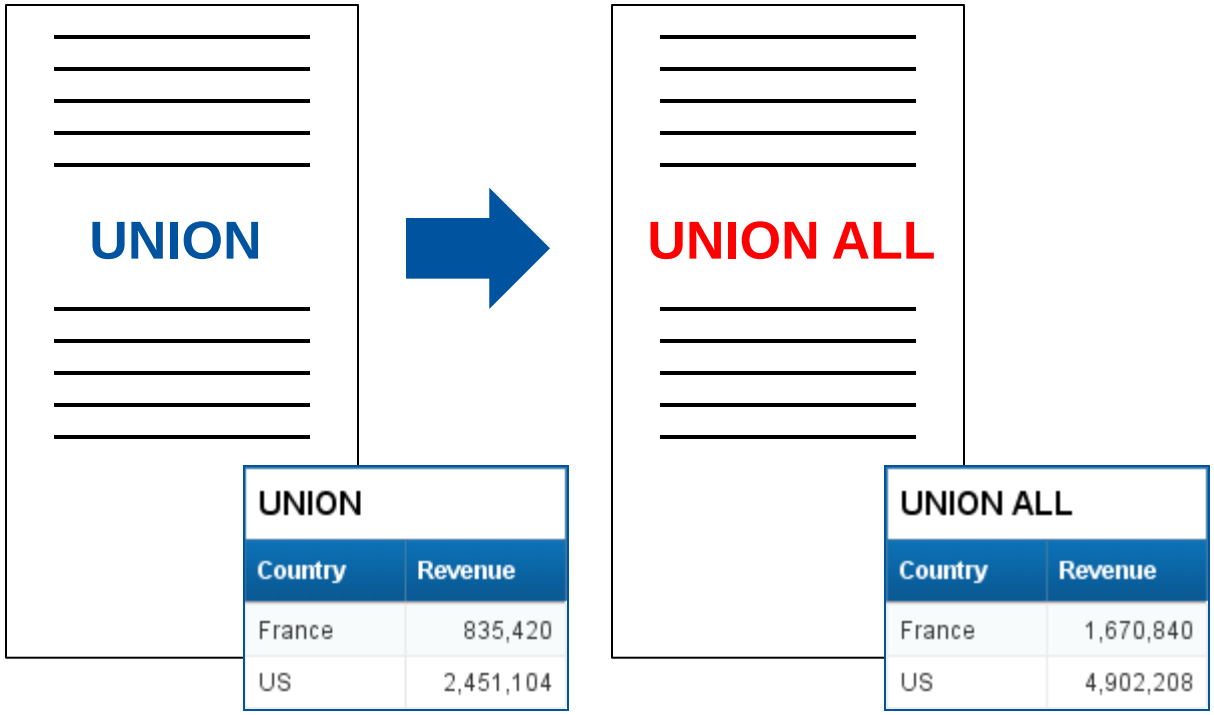


UNION

Country	Revenue
France	835,420
US	2,451,104

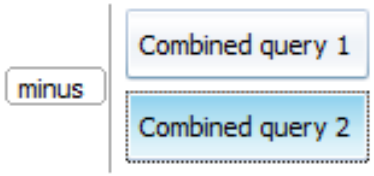
UNION Queries, cont'd

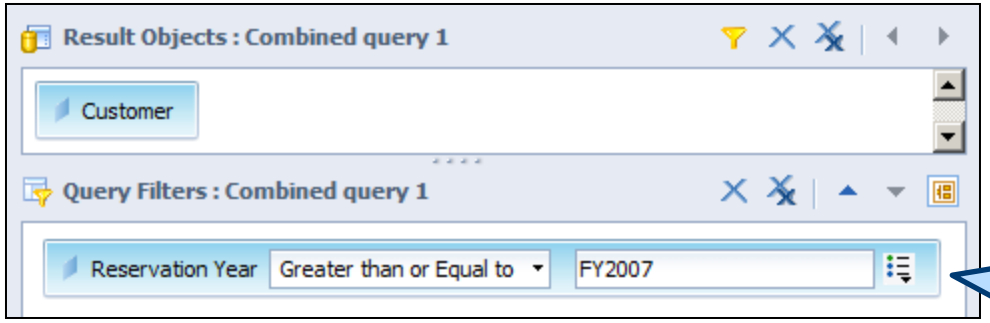
- Solution: Use **UNION ALL**
  - No rows are removed based on uniqueness
  - Requires modification to SQL statement directly
    - Custom SQL script must be used

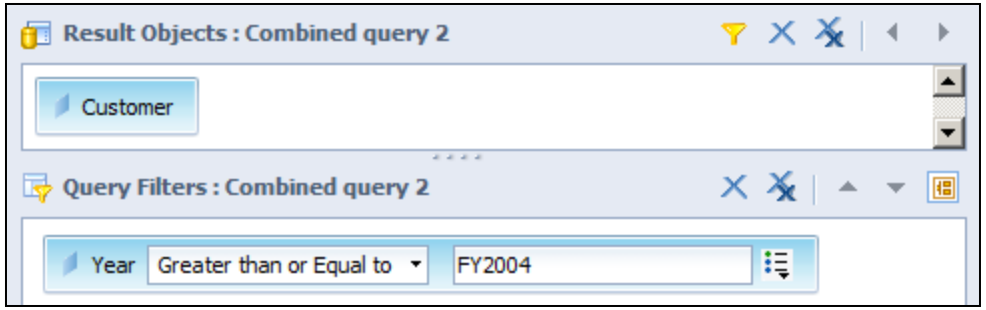


MINUS Queries

- Find New Customers
  - Customers with reservations and no previous stays





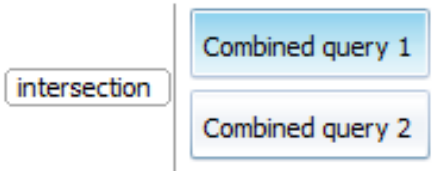


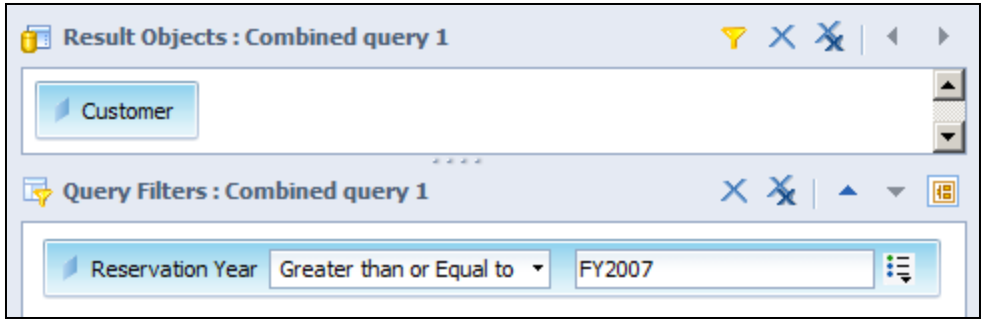
Why create the condition like this?

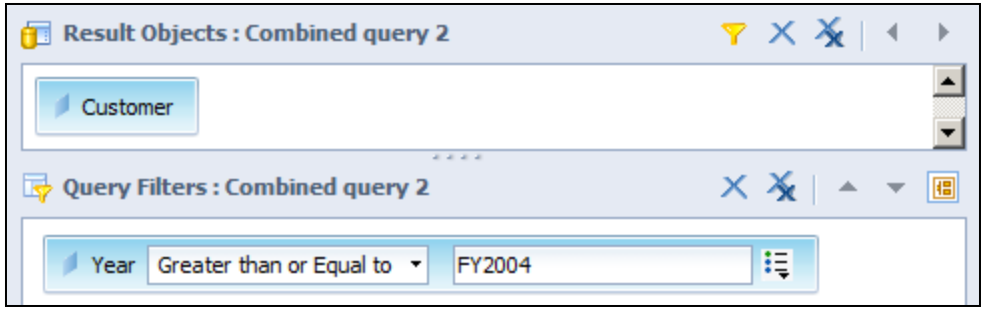
We'll see in a minute

Intersect Queries

- Find Repeat Customers
  - Customers with reservations and previous stays







Demonstration

Subqueries

- Uses a query to complete a condition
  - Serves as the operand
  - No restrictions on the number of result objects
  - Subquery can only return values for one object
- **ANY** or **ALL** values returned can be used

Why use a subquery?

Sometimes the condition can't be created as a simple comparison.

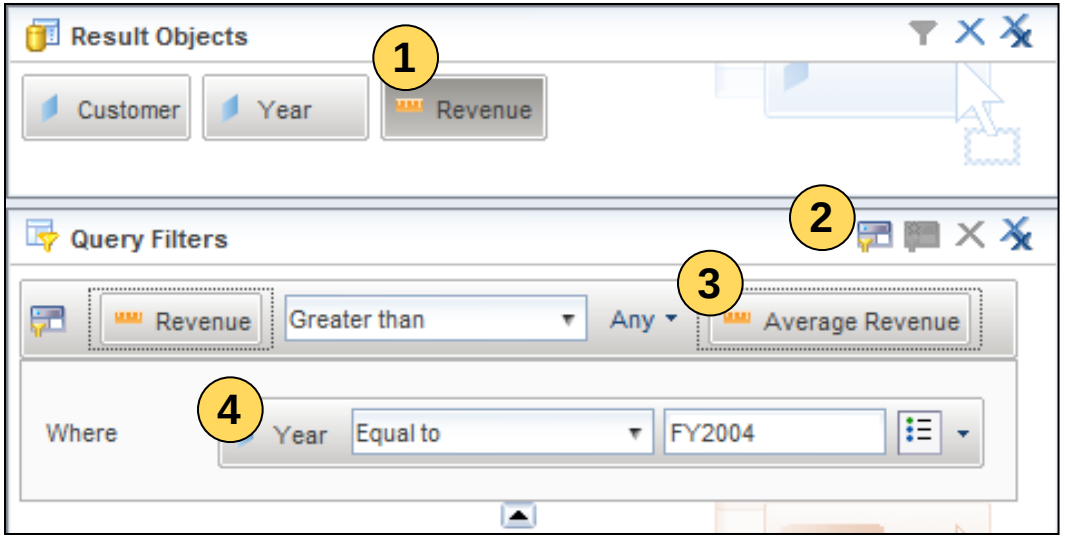
CONDITION

Object – **Which** business term is being compared (Customer, Revenue, …)

Operator – **How** are we comparing (Equal to, In List, …)

Operand – **What** values are we comparing to?

Creating a Subquery



Subqueries can be nested

Meaning … a second subquery can be used as the condition!

1. Select the object to compare

2. Select the **Subquery** button

3. Select an object whose values you'd like to compare (Any / All)

4. Add any conditions on this object you'd like (optional)

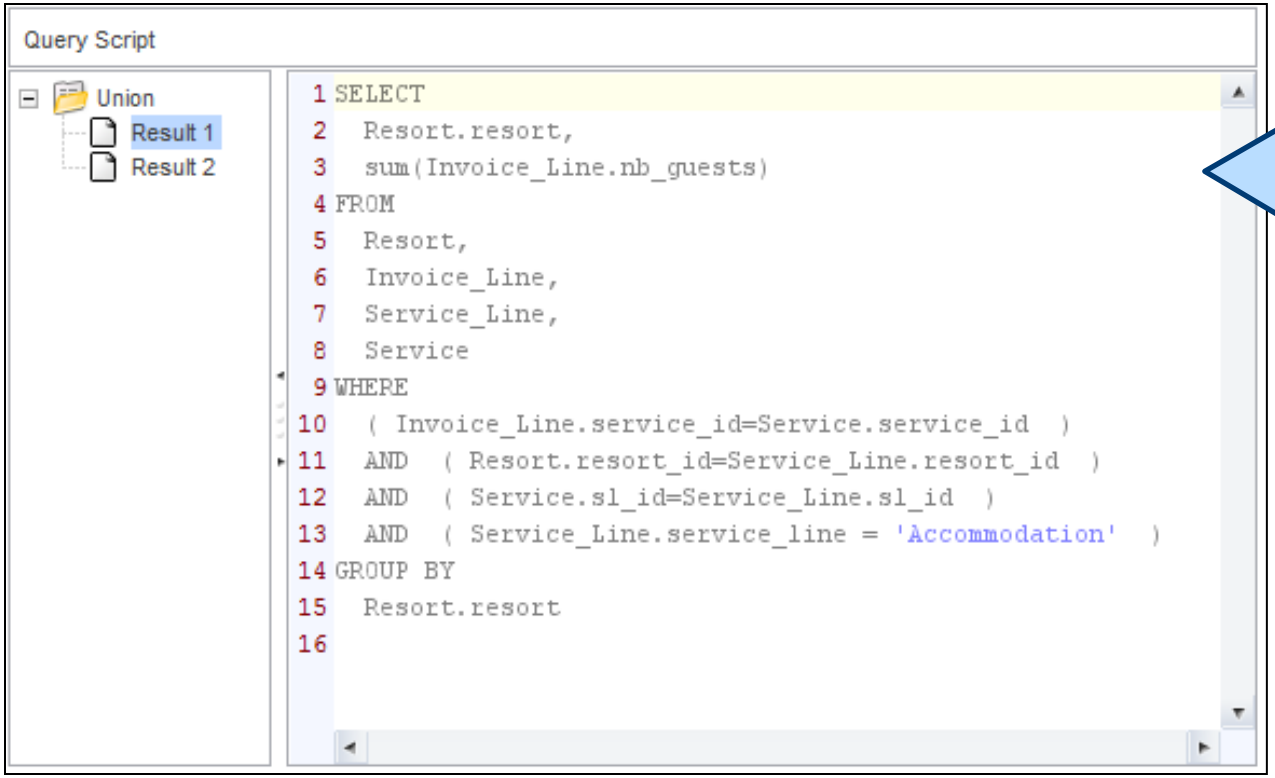
Demonstration

Multiple SELECTS

- Some data providers create multiple SELECT statements
  - Various reasons …
  - Multiple measures used as result objects
  - Universe has two or more contexts defined
- Knowing when this occurs is important
  - May slow down performance
    - Multiple queries to the database
  - Results may not combine correctly
  - Time and row limits are applied differently in these situations

Joined or Unioned SELECTs

- SELECT statements are combined by Webi (not the DB)
  - Combines based on the objects each SELECT has in common
  - Usually the objects are part of the GROUP clause

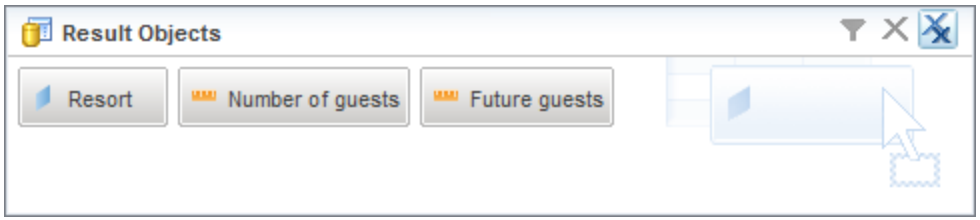


Union is the term used by BI 4.x

XI 3.1 uses the word **Join**

Joined or Unioned Results

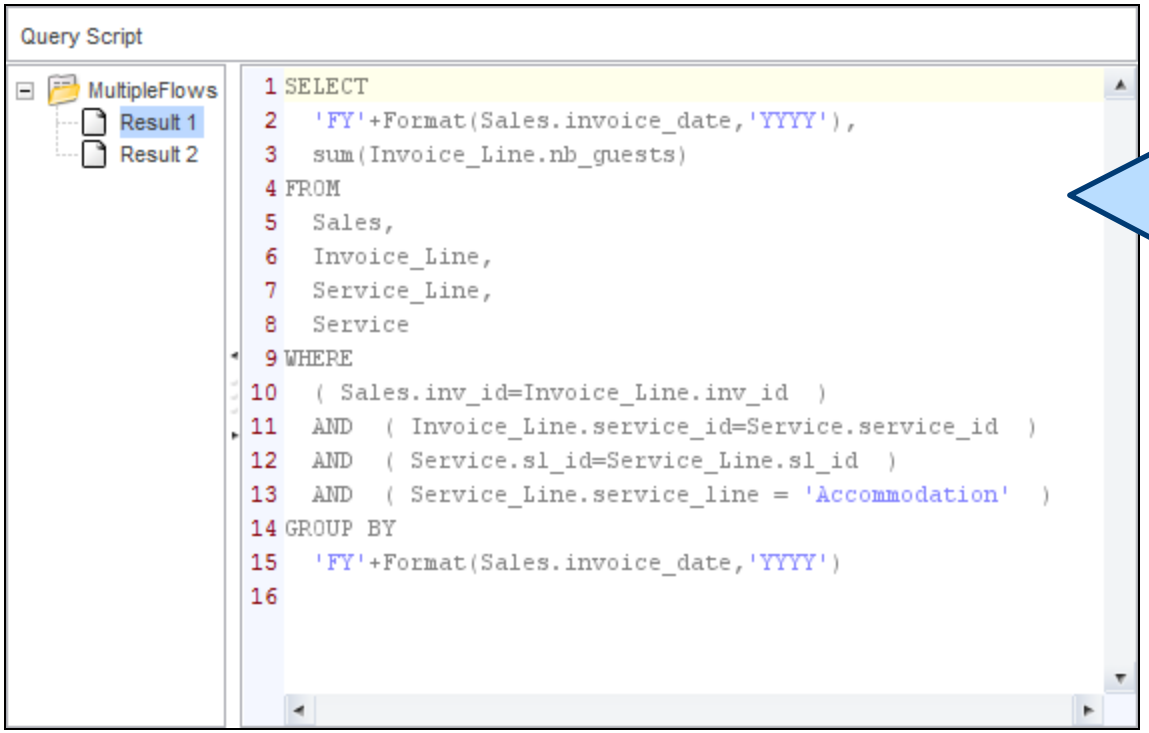
- The results from this combination is usually acceptable
  - Looks like the data came from a single query
  - This technique usually passes report reviews



Resort	Number of guests	Future guests
Bahamas Beach	565	35
French Riviera	446	46
Hawaiian Club	540	21

MultiFlow / Synchronized SELECTs

- SELECT statements are combined by Webi (not the DB)
  - At least one object is not part of both queries
  - This will cause an interesting side effect

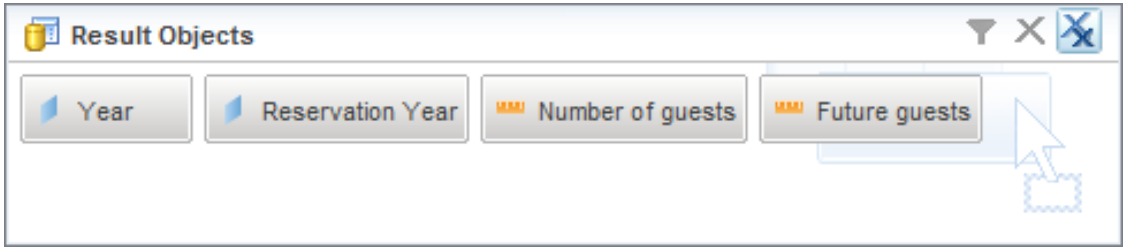


MultipleFlows is the term used by BI 4.x

XI 3.1 uses the word **Synchronization**

MultiFlow Results

- The results may **NOT** be acceptable
  - Cannot combine all results in the same table
  - Two or more tables are used to separate the data



Year	Number of guests
FY2004	518
FY2005	525
FY2006	508

Reservation Year	Future guests
FY2007	65
FY2008	23
FY2009	14

MultiFlow Results – Trying to Fix

- Trying to adjust the results may make matters worse …
  - Some developers try adding objects manually to one table
  - The results are incorrect

Forcing the Flow

Year	Number of guests	Future guests
FY2004	518	102
FY2005	525	102
FY2006	508	102

Reservation Year
FY2007
FY2008
FY2009

Row and Time Limits

- Universe developers can place limits on your queries
  - Many different restrictions can be established
  - Most popular are row and time limits
    - Helps prevent runaway queries
- Multi-SELECT queries have different rules for rows / time
- Not many users OR developers know those rules
  - Or the consequences …

Row Limits for Multi-SELECT Queries

- The row limit is enforced BY SELECT
- 50,000 row limit could actually retrieve 150,000 rows if 3 SELECT statements are combined
- This only applies to multiple SELECT statements combined locally by Webi
  - Does not apply to the Combine Queries technique described earlier
  - UNION, INTESECT, MINUS are safe
  - Combination of results done at the database

More rows than you might expect!

Time Limits for Multi-SELECT Queries

- The time limit is averaged across SELECT statements
- 3 minute time limit could limit each query to a minute if 3 SELECT statements were involved
- As with row limits, this only applies to multiple SELECT statements combined locally by Webi

Less time than you might expect!

Partial Results and Multi-SELECTs

- One nasty bug related to row limits was recently fixed …
  - Partial Results Indicator used results from the LAST SELECT processed
  - What this meant for users:
    - Reports may have been produced that were incomplete
    - No visual indication from the Partial Results Indicator
  - My own testing:
    - Broken since XI 3.1 SP3 Fix Pack 4 (3.3.4)
    - Fixed in X 3.1 SP6 Fix Pack 1 (3.6.1)
  - If you're still on XI 3.1 and using multi-SELECT logic, you may want to check on this.

Demonstration

Agenda

- Introduction
- Query Techniques
- **Report Techniques**
- Performance Considerations
- Testing
- Conclusion

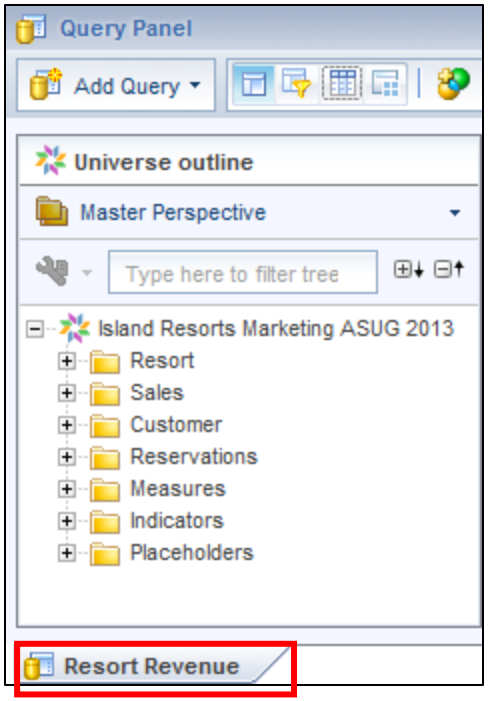
Reporting Techniques

- Topics geared to the report writer:
  - Names for variables, blocks, and cells – and why
  - Documentation – manual and automatic
  - Multi-layered variables
  - Dynamic sorts and breaks
  - Interactive drilldowns

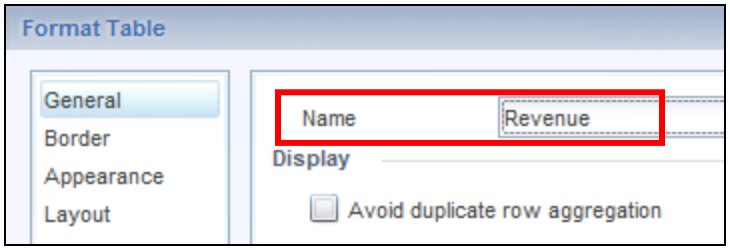
Names are Important

- Everything you create should have a name

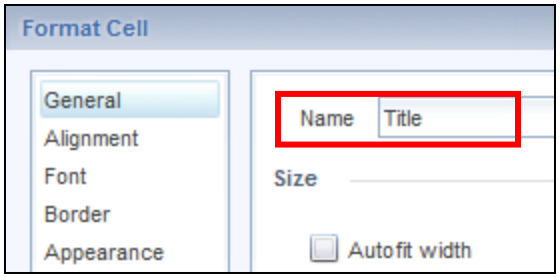
Queries



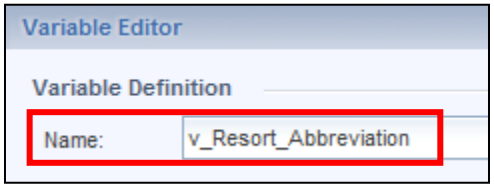
Blocks



Cells

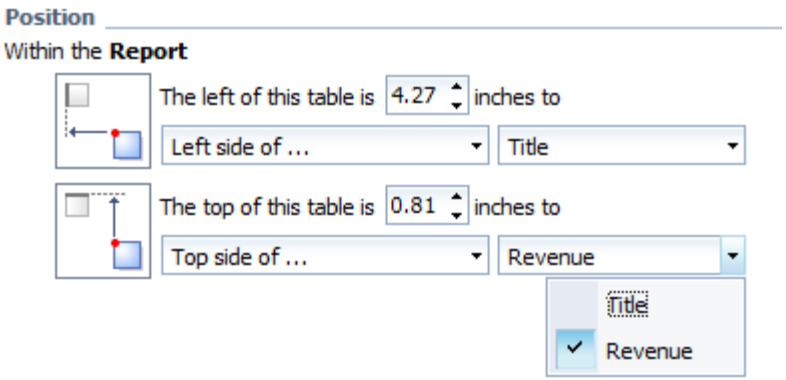


Variables

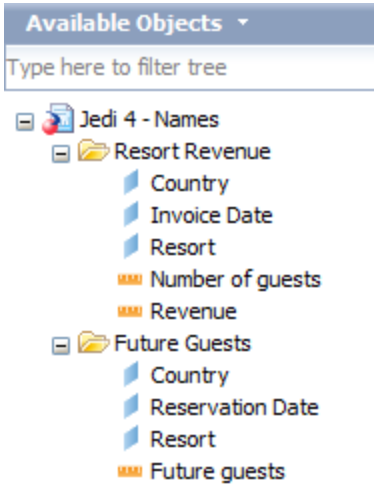


Names are Important

- Placement
  - Helps when positioning one block or cell relative to another



- Sorting
  - Query names allow easier sorting of variables



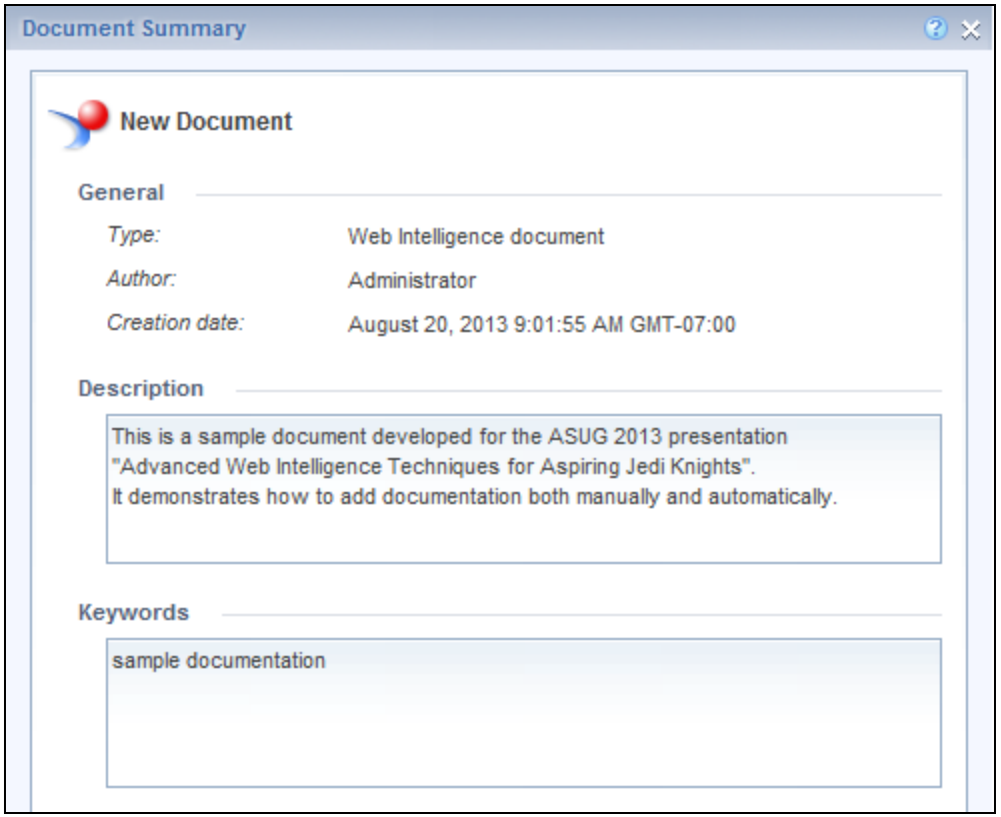
Variable Names

- Consider the following rules for variable names
  - ALWAYS add a prefix to a report variable's name
  - Different prefixes give you more control
    - **v_** generic prefix for any variable name
    - **c_** add to constants
    - **p_** add to prompts
  - Why?
    - Easier maintenance
    - Can manipulate the report with refreshing
  - Examples:
    - v_PaymentDue
    - p_Year
    - c_BucketSize

The remainder of the name is up to you. v_Payment_Due works just as well.

Document Everything

- Add description wherever possible
  - Document description
  - Keywords



Document Everything, cont'd

- Add a report page as an introduction
  - A cell formatted to wrap text contains your descriptions
  - An empty two-column table contains your prompts
  - Position the prompt table relative to the description cell

This is a sample document developed for the ASUG 2013 presentation "Advanced Web Intelligence Techniques for Aspiring Jedi Knights". It demonstrates how to add documentation both manually and automatically.

Prompt	Value
Year	FY2006
Month	Aug

Use the advice just given for prompt variables:

p_Year =UserResponse("Year")
p_Month =UserResponse("Month")

Document Everything, cont'd

- Add a change log
  - Document who modified the report and when
  - An additional 2 to 3 column table on the same page

This is a sample document developed for the ASUG 2013 presentation "Advanced Web Intelligence Techniques for Aspiring Jedi Knights". It demonstrates how to add documentation both manually and automatically.

Prompt	Value
Year	FY2006
Month	Aug

Version	Date	Author	Description
1.0	8-20-2013	Alan Mayer	Initial Creation

Some report writers keep this information in a separate system. Making it part of the report is very convenient.

Document Everything, cont'd

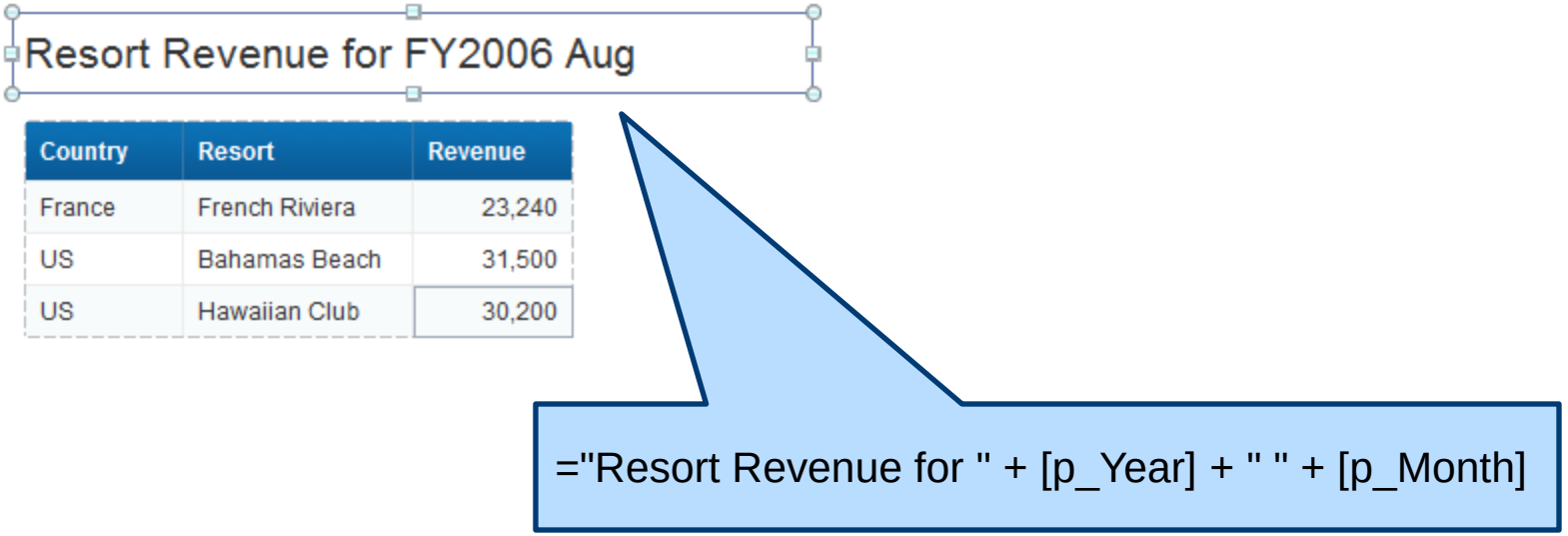
- Add a report page as an introduction
  - A cell formatted to wrap text contains your descriptions
  - An empty two-column table contains your prompts
  - Position the prompt table relative to the description cell

This is a sample document developed for the ASUG 2013 presentation "Advanced Web Intelligence Techniques for Aspiring Jedi Knights". It demonstrates how to add documentation both manually and automatically.

Prompt	Value
Year	FY2006
Month	Aug

Document Everything, cont'd

- Embed prompt values as repeated headers
  - Create a variable that will contain prompt values
  - Repeats on every page
    - Format Cell > Layout > Repeat on every page



Layered Variables

- Concept that works well for complex logic
- Break up the formula into simpler pieces …
- … then use those pieces to build the final variable
- Let's look at an example using sample data
  - Island Resorts Marketing
  - Group invoices based on date ranges
  - Very similar to aging

Layered Variables - Constants

- First, define the constants for the report
  - Base date is defined (normally the current date)
  - “Days available” in each bucket

Variable	Formula
c_BaseDate	=ToDate("2006-01-01"; "yyyy-MM-dd")
c_Bucket1	=30
c_Bucket2	=30
c_Bucket3	=30

These variables could be replaced with prompts for an even more interactive example.

Layered Variables – Layer 1

- Next, define the start and end dates for each bucket
  - Buckets 1 shown below
  - Buckets 2 – 3 and the Remainder bucket defined the same way
  - Note how these formulas use the constants previously defined

Variable	Formula
v_Bucket1_Begin	=[c_BaseDate]
v_Bucket1_End	=RelativeDate([v_Bucket1_Begin];[c_Bucket1] - 1)

The Remainder bucket picks up all invoices after Bucket 3 up through the end of the year.

Layered Variables – Layer 2

- Now define the Indicators
  - Indicators determine what bucket an invoice belongs in
  - The Indicator for Bucket1 is shown below
  - Note how the formula reads almost like a sentence

Variable	Formula
v_Bucket1_Indicator	=If [Invoice Date] Between([v_Bucket1_Begin];[v_Bucket1_End]) Then 1 Else 0

Layered Variables – Layer 3

- Create the Bucket Totals as the last step
  - The formula uses Indicators to make the decision easier

Variable	Formula
v_Bucket1_Amt	=Sum(If [v_Bucket1_Indicator] = 1 Then [Revenue] Else 0)

Layered Variables – Alternatives

- Complicated logic can be pushed back to the universe
  - Best practice where possible
  - Many reports can benefit from the same central definitions
  - Can push back even further to the HANA model or database
- Trading database performance for centralized definitions
  - This may not be acceptable for certain projects

Demonstration

More Dynamic Sorts and Breaks

- Certain report features are hard-coded at design time
- Sorts and breaks fall into this category
  - Sorts and breaks are based on a selected variable
- There **IS** a way to change these features at run time
- Each technique will require a little preparation
  - Additional universe objects are required

Dynamic Objects

- The secret to making prompts dynamic
- Must be added to the universe
- Created from metadata rather than tables
  - Cannot use them alone in a query
- All follow the same IF-Then-Else formula:

```
If @Prompt('Prompt sentence',
           'A',
            {X,Y},,,)
   = X,
   table1.column1,
   table2.column2)
```

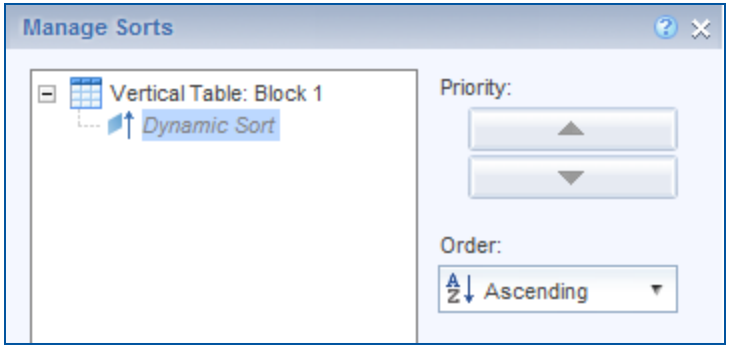
The conditional function used for IF-THEN-ELSE will vary by database.

Sorting Dynamically

- First - create the Dynamic Sort object:

```
IIF(@Prompt('Sort by','A',{'Country','City'},,,)
    = 'Country',
    Country.country,
    City.city)
```

- Second – sort on that object then hide it



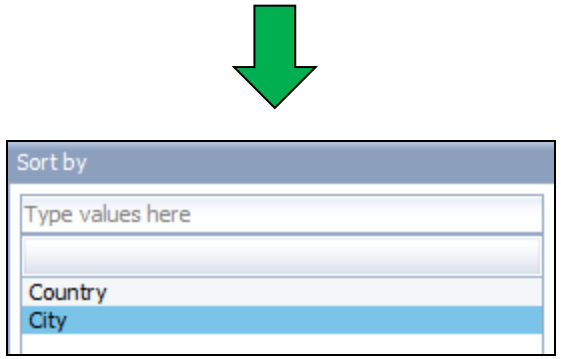
What is sorted can be programmed this way, but the **direction** of the sort cannot.

Sorting Dynamically, cont'd

- The results:

Dynamic Sorts

Country	Region	City	Customer	Revenue
Germany	Bavaria	Augsburg	Titzman	394,996
Germany	Bavaria	Munich	Schiller	388,524
Germany	East Germany	Berlin	Reinman	12,112
Germany	East Germany	Berlin	Schultz	20,330
Germany	East Germany	Dresden	Durnstein	4,400
Germany	East Germany	Magdeburg	Weimar	222,910
Germany	Ruhr	Cologne	Diemers	10,976



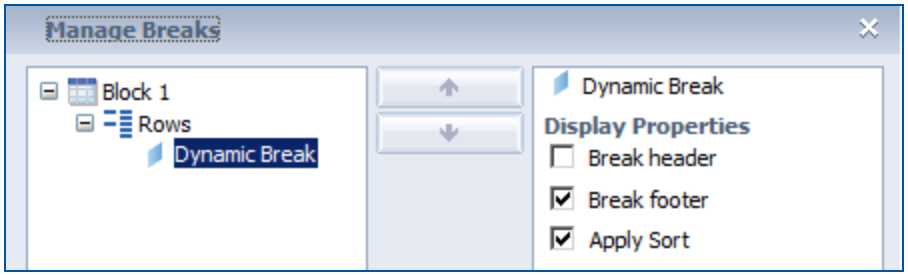
Country	Region	City	Customer	Revenue
Germany	Bavaria	Augsburg	Titzman	394,996
Germany	East Germany	Berlin	Reinman	12,112
Germany	East Germany	Berlin	Schultz	20,330
US	Mid West	Chicago	Baker	441,594
Germany	Ruhr	Cologne	Diemers	10,976
US	South	Dallas	McCarthy	400,899
Germany	East Germany	Dresden	Durnstein	4,400

Breaking Dynamically

- First - create the Dynamic Break object:

```
IIF(@Prompt('Break by','A',{'Country','Region'},,,)
    = 'Region',
    Country.country,
    Region.region)
```

- Second – break on that object then hide it

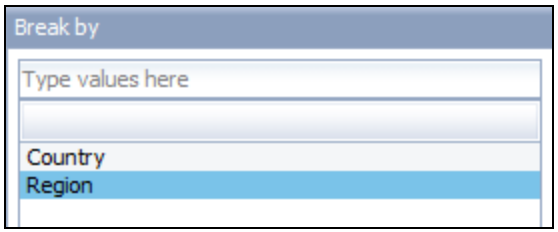


Breaking Dynamically, cont'd

- Dynamic breaks in motion!

Dynamic Break

Dynamic Break	Country	Region	City	Revenue
Germany	Germany	Bavaria	Augsburg	394,996
	Germany	Bavaria	Munich	388,524
	Germany	East Germany	Berlin	32,442
	Germany	East Germany	Dresden	4,400
	Germany	East Germany	Magdeburg	222,910
	Germany	Ruhr	Cologne	10,976
Germany			**Sum:**	**1,054,248**



Dynamic Break	Country	Region	City	Revenue
Bavaria	Germany	Bavaria	Augsburg	394,996
	Germany	Bavaria	Munich	388,524
Bavaria			**Sum:**	**783,520**

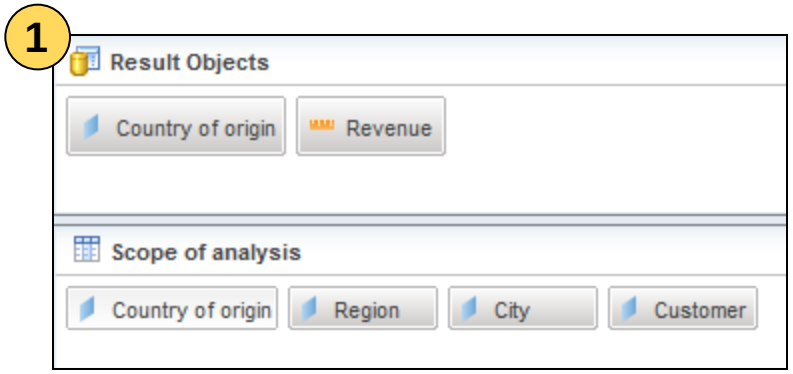
Demonstration

Interactive Drilling

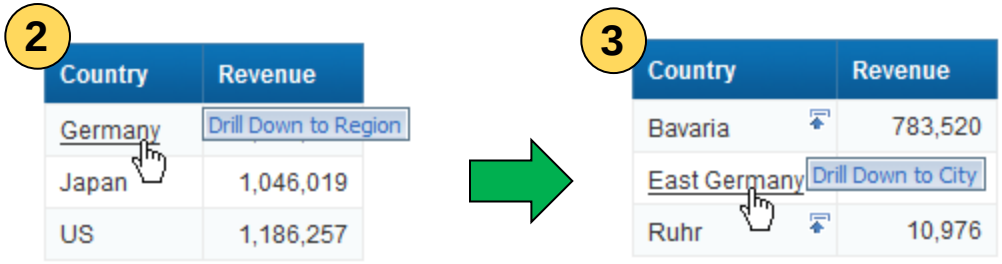
- Drilling allows the user to retrieve additional information
  - For a particular table row or chart element
  - For a selected cell
- Drilling can be added to Webi reports using three methods:
  - Navigation paths (drilling in XI 3.1)
  - Document links
  - Element links (only BI 4.x)
- We'll show examples of each

Using Navigation Paths

- Legacy technique introduced in BusinessObject v5



Navigation Paths

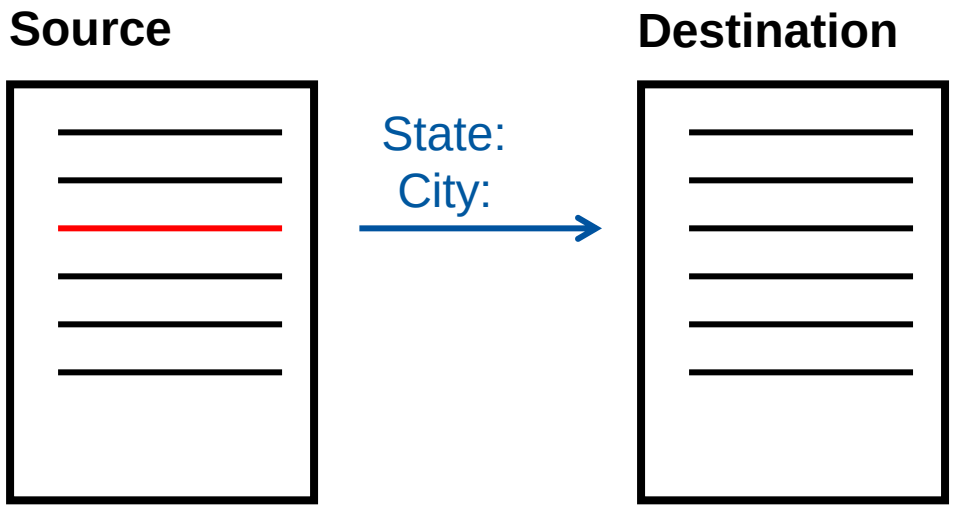


1. Add drill levels using the Scope of Analysis pane
2. Refresh the report and place into Drill mode
3. Click on any hyperlinked dimension to drill to the next level

The dimension clicked on is replaced by the next in the navigation path (hierarchy). This technique does not add any additional detail.

Document Links

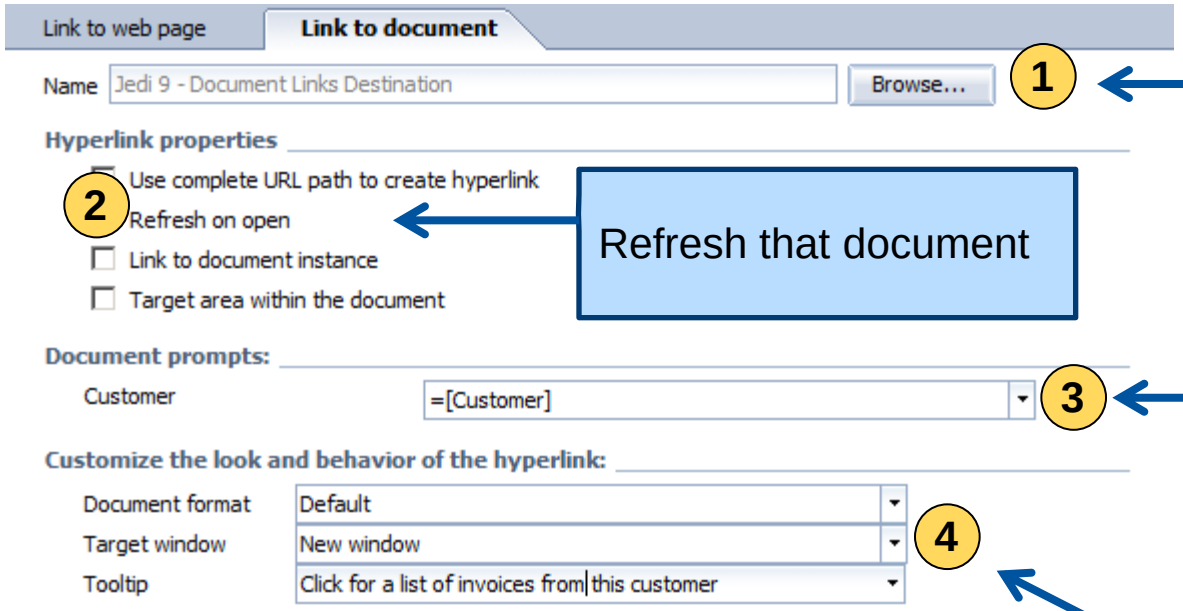
- Documents can be linked to others via prompts
  - Links are created in the source document
  - Those links answer prompts from a second document
  - That document can contain much more detail



The previous drilling technique using navigation paths could not add more detail

Document Links, cont'd

- Link Source to Destination Document
  - Right-click on the selected area
    - Linking > Add Document Link



Locate the document to link to

Refresh that document

Map prompts from the document in Step 1 to objects in this one

Choose a new or existing window to display

Element Links

- Blocks can be linked to others via prompts
  - Same concept as document links
  - Values from the first block control a second block

Source block

		FY2004	FY2005	FY2006
France	French Riviera	295,940	280,310	259,170
US	Bahamas Beach	287,929		
US	Hawaiian Club	479,685		

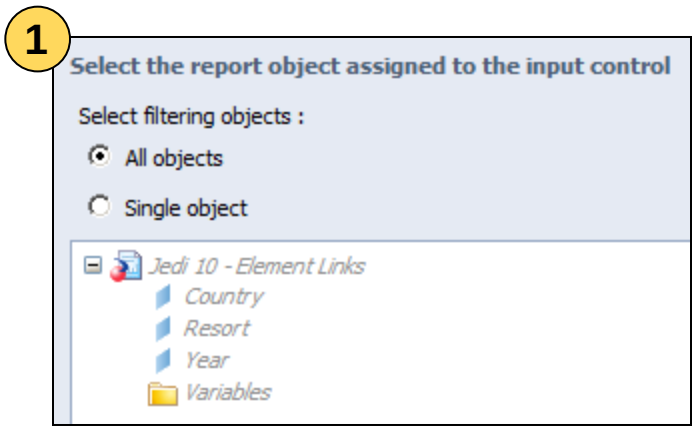
Country=France
Resort=French Riviera
Year=FY2005

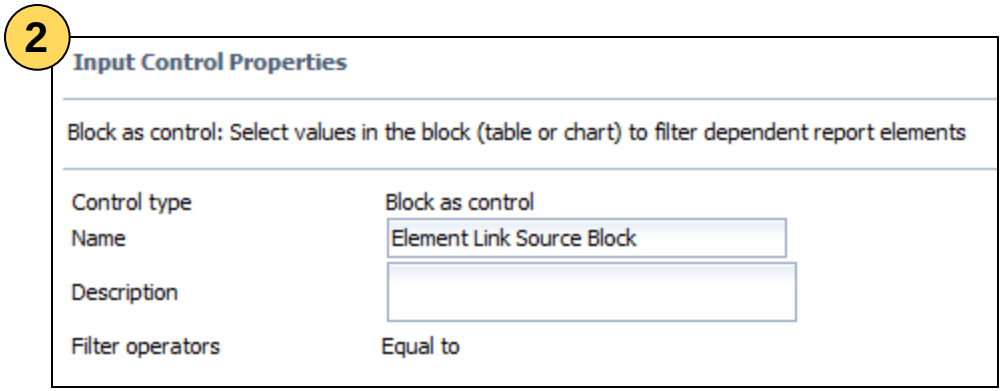
Destination block

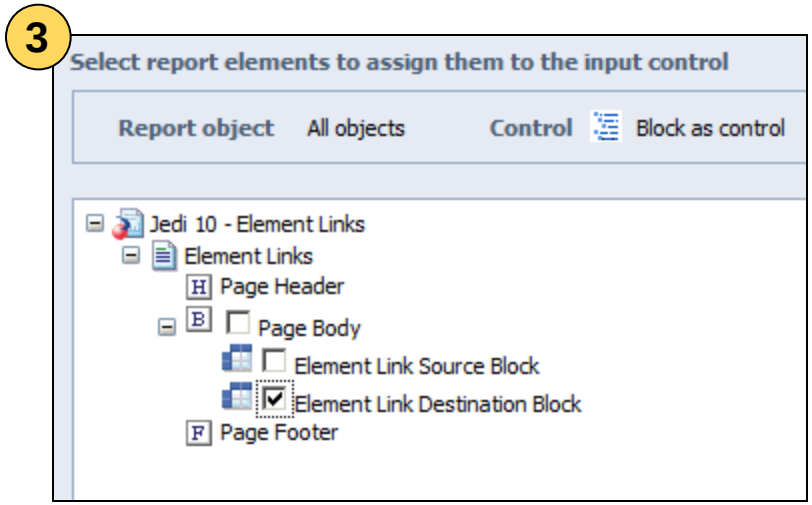
	Q1	Q1	Q1
	Jan	Feb	Mar
Accommodation	10,710	11,520	29,160
Food & Drinks	3,570	2,880	3,840

Element Links, cont'd

- Creating the link







1. Select all dimensions from one block
2. Add a description (optional)
3. Choose the destination block

Demonstration

Agenda

- Introduction
- Query Techniques
- Report Techniques
- **Performance Considerations**
- **Testing**
- Conclusion

Bonus! - Performance

- Numbers to consider
  - Number of unused query objects
  - Number of data providers
  - Number of unused report variables
  - Number of rows returned
- Other items to check
  - Query conditions
  - Report complexity
  - Formula and variable complexity
  - Size of the final document
- Grade your reports
  - Based on rows / time / size

Bonus! - Testing

- Why test?
  - Software version has changed
  - Report modifications
  - Lifecycle promotion (Dev to Test to Prod)
- Easier items to test for:
  - Data
  - Format
  - SQL / SELECT statements
- Harder items
  - Duration
  - Data source validation

Agenda

- Introduction
- Query Techniques
- Report Techniques
- Performance Considerations
- Testing
- **Conclusion**

Key Learnings

- Many advanced techniques are available for Webi reporting
- This presentation covered a selected few
  - Query (Combined, Sub, Multi-SELECT)
  - Report (Documentation, layered variables, drilling)
  - Performance
  - Testing
- Any one of these topics is worthy of their own presentation
- If you liked what you learned …
  - Turn in a review – Session 1214
  - And may the Force be with you!

Questions?

Alan Mayer
Session 1214
Advanced Web Intelligence Techniques for Aspiring Jedi Knights

alan.mayer@solidgrounded.com
214-295-6250 (office)
214-755-5771 (mobile)
214-206-9003 (fax)

Thank you for participating.

Please provide feedback on this session by completing a short survey via the event mobile application.

SESSION CODE: 1214

Learn more year-round at www.asug.com