IBM Planning Analytics:

Caveats on the use of

TM1 Dimension Element/Member Alias Names

in Rules & Feeders

Andreas Kugelmeier
Executive Consultant, FOPM
Planning Analytics Architect
IBM Data and AI Expert Labs
Mobile Phone: +1-215-384-7302
Email: kugelmeier@us.ibm.com

Notices & Disclaimers

Document Version History

Date	Version	Author	Description
08/04/2015	0.9	Andreas Kugelmeier	Draft Version
12/02/2015	1.0	Andreas Kugelmeier	First Version

Table of Contents

1. Summary

The use of TM1 Dimension Element Alias Names in Rules should be avoided: Calculations that reference an Alias (such as 'Current Month' for example) will have changing elements as the basis for the calculation. While the calculation statement (the rule) will not change, and underlying data may not have changed, the data being referenced will. Such Alias-based calculation rules are dependent on the alias mapping as defined in the attributes cube - hence a dependency is needed to perform invalidation (or stale data can persist after version submission.) Therefore, dependencies to the attribute cubes are established once such alias-based rules are encountered. The establishment of those dependencies can cause a temporary lock: In normal operations, cube dependencies are established when data which crosses cube boundaries (such as data that is derived by a rule that references an external cube) is retrieved. To create the dependency information, the server must lock the cubes while the dependency is established, potentially maintaining the lock during a long view calculation. Since this is a 'write' lock, other users are prevented from accessing the cubes.

2. Sample Scenarios

Calculations that reference an Alias such as 'Current Month' will have changing months as the basis for the calculation. While the calculation statement (the rule) will not change, and underlying data may not have changed, the data being referenced will. <u>The calculations are dependent on the alias mapping as defined in the attributes cube - hence a dependency is needed to perform invalidation</u> (or stale data can persist after version submission.) Therefore, dependencies to the attribute cubes are established once such alias-based rules are encountered.

The establishment of those dependencies can cause a temporary lock: In normal operations, cube dependencies are established when data which crosses cube boundaries (such as data that is derived by a rule that references an external cube) is retrieved. To create the dependency information, the server must lock the cubes while the dependency is established, potentially maintaining the lock during a long view calculation. Since this is a 'write' lock, other users are prevented from accessing the cubes. The AddCubeDependency function allows the dependency to be established within a Turbointegrator process, preventing later lock contention as no new dependency need be established.

See http://www-01.ibm.com/support/docview.wss?uid=swg27024296 & http://www-01.ibm.com/support/docview.wss?uid=swg27024191

Note that the dependencies-lock does not apply to feeder statements as feeders will only be evaluated during cube-start-up.

Beyond the possible locking due to the establishment of dependencies there are other caveats to take into consideration. Let's say we use a 'Current Month' alias like:

	Current Month Control (Alias)
Jan	Jan
Feb	Feb
Mar	Mar
Apr	Apr
May	Current month
Jun	Jun
Jul	Jul
Aug	Aug

In a cube, we have two rules tied to 'Current Month':

```
['a','current month']=1;
['jan']=['current month'];
```

> (1) the value for current month is to be = 1 and
> (2) Jan is to be equal to the 'current month'.

These rules result in the following data in our sample cube:

alphabet	month							
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug
a	1	0	0	0	1	0	0	0
b	5	2	3	4	5	6	7	8
c	5	2	3	4	5	6	7	8

Because May is currently our 'Current Month', month May gets 1 as expected, and Jan reads from May as expected. Now, if we reset (May, Current Month Control) to May, then set (Jun, Current Month Control) to "Current Month" like in

	Current Month Control (Alias)
Jan	Jan
Feb	Feb
Mar	Mar
Apr	Apr
May	May
Jun	Current Month
Jul	Jul

TM1 does **not** react to the change:

alphabet \ month	Jan	Feb	Mar	Apr	May	Jun	Jul	A
a	1	0	0	0	1	0	0	
b	5	2	3	4	5	6	7	
c	5	2	3	4	5	6	7	

TM1 will not react unless I edit the rule, or perform dimension maintenance on 'month' or 'alphabet:

alphabet \ month	Jan	Feb	Mar	Apr	May	Jun	Jul	
a	1	0	0	0	0	1	0	
b	6	2	3	4	5	6	7	
c	6	2	3	4	5	6	7	

Notes
- changing the value of the alias via the "Edit Attributes" dialog or via direct entry against the }ElementAttributes_month cube (CellPutS, Spread, etc.) will not cause TM1 to identify the change. Only dimension maintenance or editing the TM1 rule will trigger the new calc logic
- the same issue applies to feeders using Alias names: if/once the alias name changes, the feeder will not update. Only re-processing feeders will cause a re-evaluation of the feeder statements/logic.

3. Sample Objects

}ElementAttributes_month.dim   }ElementAttributes_month.cub   sample2.cub   month.dim

sample2.RUX   alphabet.dim