Deep-Dive to
Analysis Services Security

By Gerhard Brueckl

gerhard@gbrueckl.at

http://blog.gbrueckl.at

SSAS
MAESTRO
by Microsoft

pmOne

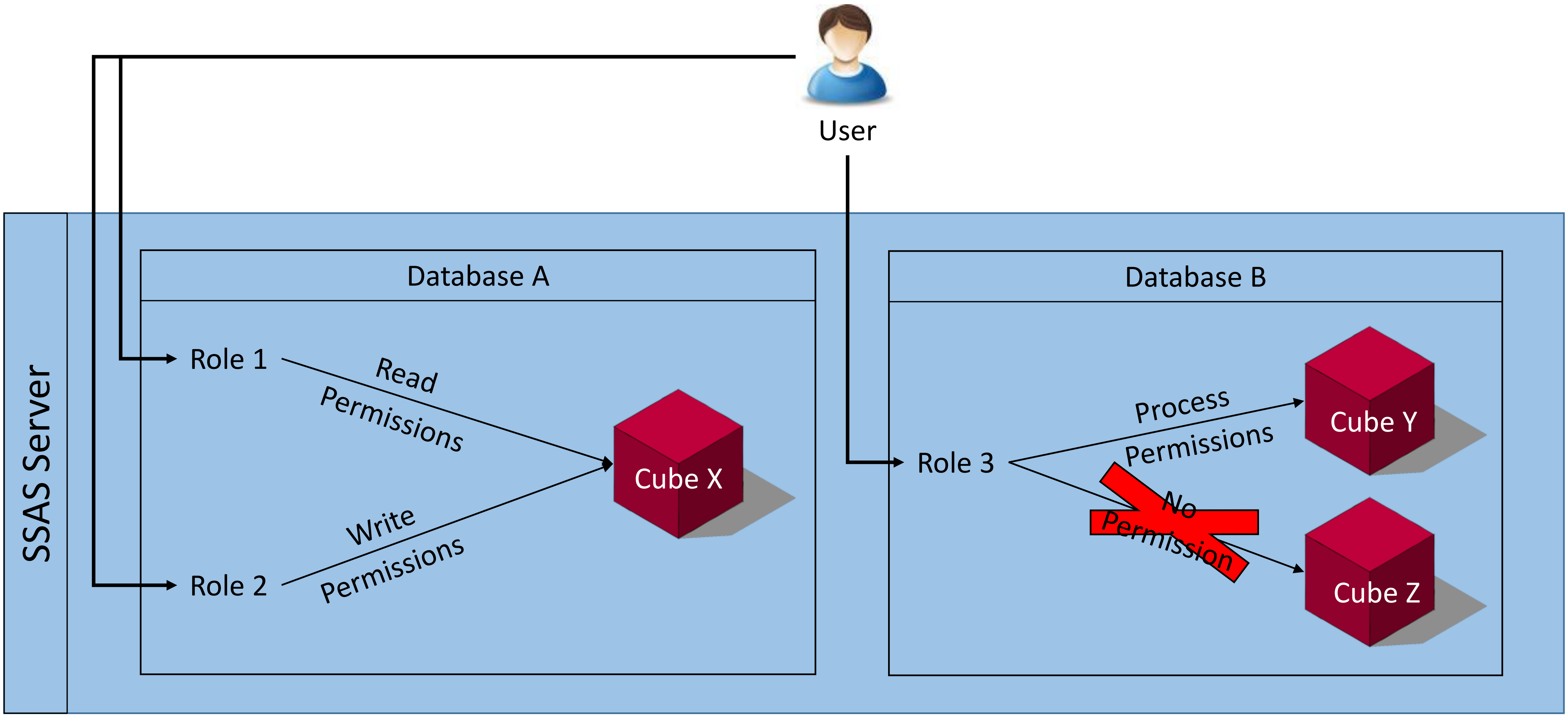
http://www.pmone.com

Agenda

- General

- Multiple Roles / Permissions
- Evaluation
- Caching
- Dynamic Security / Assemblies

General


User
SSAS Server
Database A
Role 1
Read Permissions
Role 2
Write Permissions
Cube X
Database B
Role 3
Process Permissions
Cube Y
No Permission
Cube Z

User, Roles and Permissions

Users and Groups

- Are assigned to Roles
- Active Directory

For Non-Active Directory Environments:

- Using Local Users
- msmdpump.dll via IIS and HTTP

Demo

Access from other Domain

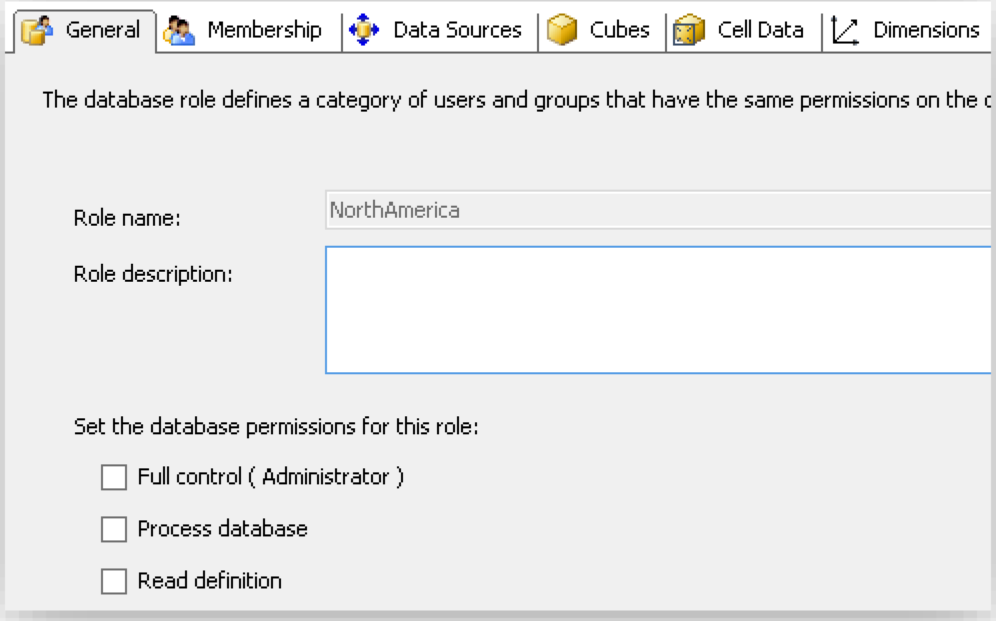
User, Roles and Permissions

Roles

- Belong to Database
- Contains Users/Groups
- Defines Permissions

Special Role

- Server Administrator



The database role defines a category of users and groups that have the same permissions on the c

Role name: NorthAmerica

Role description:

Set the database permissions for this role:

□ Full control (Administrator)

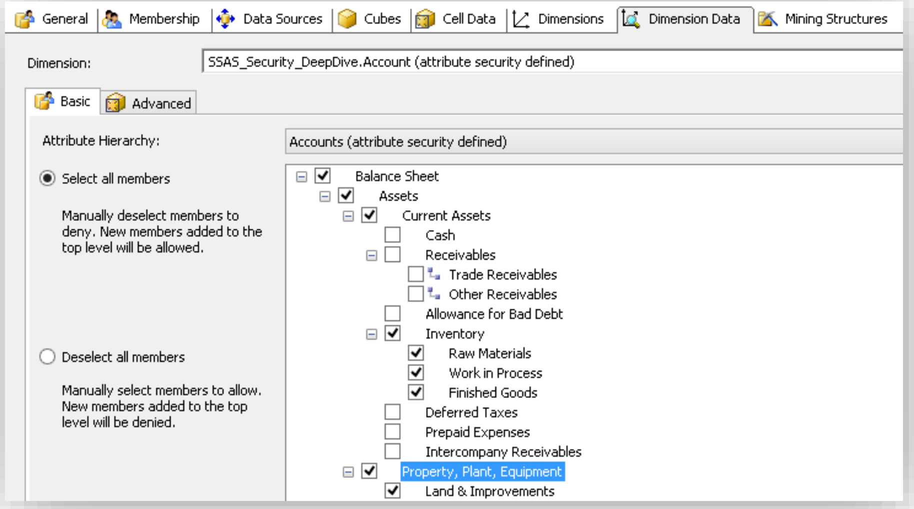
□ Process database

□ Read definition

User, Roles and Permissions

Permissions – Multidimensional

- Database
- Data Sources
- Cubes
- (Cells)
- Dimensions (MDX)
- (Mining Models)



Dimension: SSAS_Security_DeepDive.Account (attribute security defined)

Attribute Hierarchy: Accounts (attribute security defined)

● Select all members

Manually deselect members to deny. New members added to the top level will be allowed.

○ Deselect all members

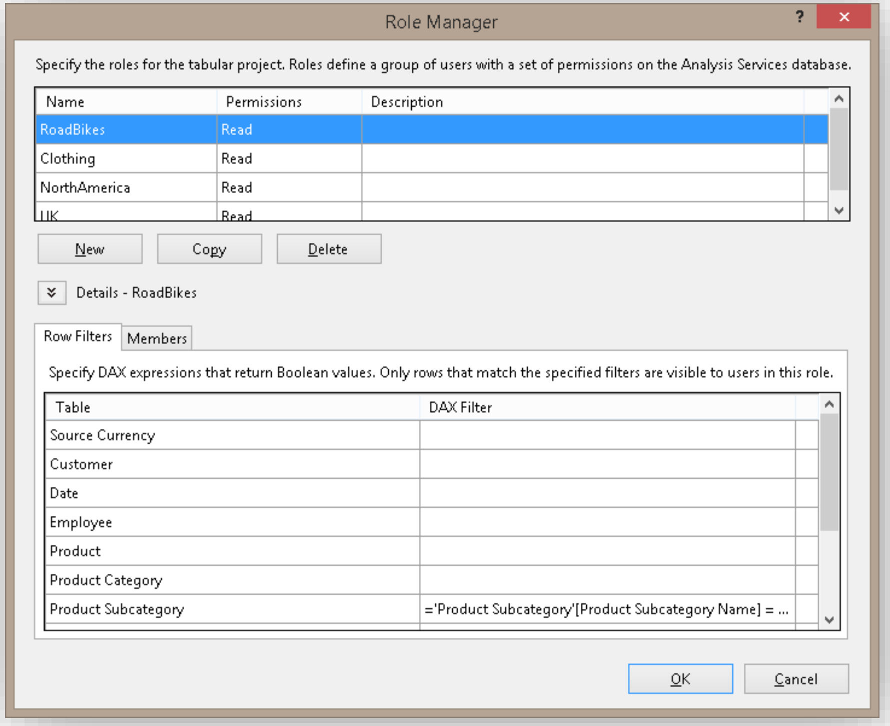
Manually select members to allow. New members added to the top level will be denied.

- ☑ Balance Sheet
  - ☑ Assets
    - ☑ Current Assets
      - □ Cash
      - □ Receivables
        - □ Trade Receivables
        - □ Other Receivables
      - □ Allowance for Bad Debt
      - ☑ Inventory
        - ☑ Raw Materials
        - ☑ Work in Process
        - ☑ Finished Goods
      - □ Deferred Taxes
      - □ Prepaid Expenses
      - □ Intercompany Receivables
    - ☑ Property, Plant, Equipment
      - ☑ Land & Improvements

User, Roles and Permissions

Permissions – Tabular

- Database
- Rows (DAX)



Specify the roles for the tabular project. Roles define a group of users with a set of permissions on the Analysis Services database.

Name	Permissions	Description
RoadBikes	Read	
Clothing	Read	
NorthAmerica	Read	
UK	Read	

Details - RoadBikes

Specify DAX expressions that return Boolean values. Only rows that match the specified filters are visible to users in this role.

Table	DAX Filter
Source Currency	
Customer	
Date	
Employee	
Product	
Product Category	
Product Subcategory	='Product Subcategory'[Product Subcategory Name] = ...

User, Roles and Permissions

Database Permissions

- Is created in the background
- No GUI

- AMO / XMLA
- Power Shell

Deny access without removing Role or User

Demo

Database Permissions

Multiple Roles and Permissions

Multiple Permissions

Multiple Roles are complementary

You cannot “lose” permissions

Roles can be selected using Roles-parameter in the connection string

“…;**Roles=Role1, Role2**;… ”

Same Role – Same Dimension

Secured on …

- same Attribute

→ Union / SQL OR

```
WHERE [Country] = 'United Kingdom'
OR    [Country] = 'Germany'
```

✓

- different Attribute

→ Intersect / SQL AND

```
WHERE [Country] = 'United Kingdom'
AND   [Region]  = 'North America'
```

⚠

Use MDX Union() and Exists()

Same Role – Different Dimensions

1. Security for "Same Dimension" applies

2. Facts get filtered on all secured dimension

   → SQL AND

```
SELECT ...
FROM   [Facts]
WHERE  [Country] = 'United Kingdom'
AND    [ProductCategory] = 'Clothing'
```

✓

Multiple Roles – Same Dimension

Secured on …

- same Attribute

→ Union / SQL OR

```
WHERE [Country] = 'United Kingdom'
OR    [Country] = 'Germany'
```

✓

- different Attribute

→ Union / SQL OR

```
WHERE [Country] = 'United Kingdom'
OR    [Region]  = 'North America'
```

✓

Multiple Roles – different Dimension

1. Security for “Same Dimensi s

2. Final permission set is a all Roles

→ UNION / SQL OR

```sql
SELECT ...
FROM  [Facts]
WHERE [Country] = 'United Kingdom'
OR    [ProductCategory] = 'Clothing'
```

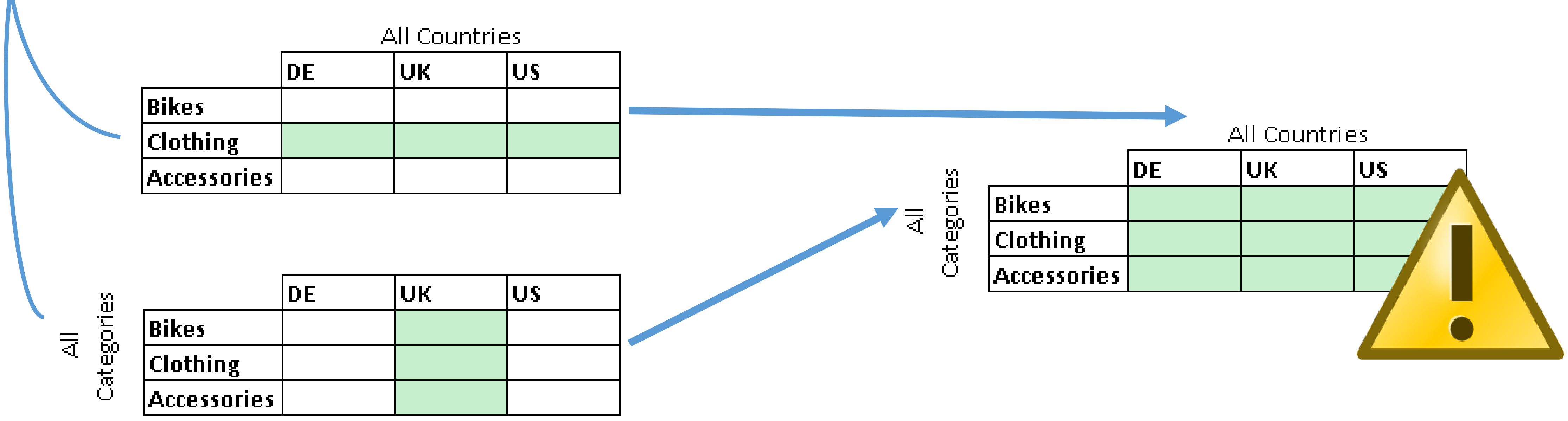
We are in a multidimensional world!

Demo

Multiple Roles

Multiple Roles – different Dimension

Role	Sales Territory	Product
Clothing	No Restrictions	[Category] = ‘Clothing’
UK	[Country] = ‘United Kingdom’	No Restrictions
{Clothing, UK}	**No Restrictions**	**No Restrictions**



All Countries

	DE	UK	US
Bikes			
Clothing	🟩	🟩	🟩
Accessories			

All Categories

	DE	UK	US
Bikes		🟩	
Clothing		🟩	
Accessories		🟩	

All Countries

All Categories

	DE	UK	US
Bikes	🟩	🟩	🟩
Clothing	🟩	🟩	🟩
Accessories	🟩	🟩	🟩

Multiple Roles – Tabular vs. Multidimensional

UK

Clothing

```
SELECT ...
FROM [Facts]
WHERE ([Country] = 'UK' OR                  1 = 1         )
OR    (        1 = 1    OR  [ProductCategory] = 'Clothing')
```

```
SELECT ...
FROM [Facts]
WHERE ([Country] = 'UK'                                    )
OR    (                      [ProductCategory] = 'Clothing')
```

Combining Roles and Permissions in Tabular

- Same Role, Same Table ✓
- Same Role, Different Tables ✓
- Multiple Roles, Same Table ✓
- Multiple Roles, Different Tables ✓

Evaluation of Security

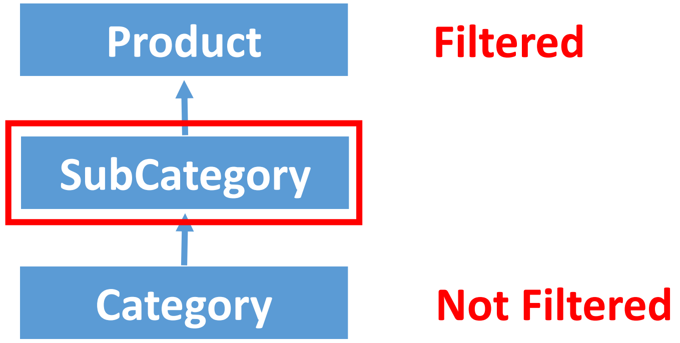
Evaluation of Security

Multidimensional

- At first connect
- Per Dimension
- Propagated to Facts by setting VisualTotals

Tabular

- At query runtime
- Propagated to all related Tables
  - Not relating Tables!

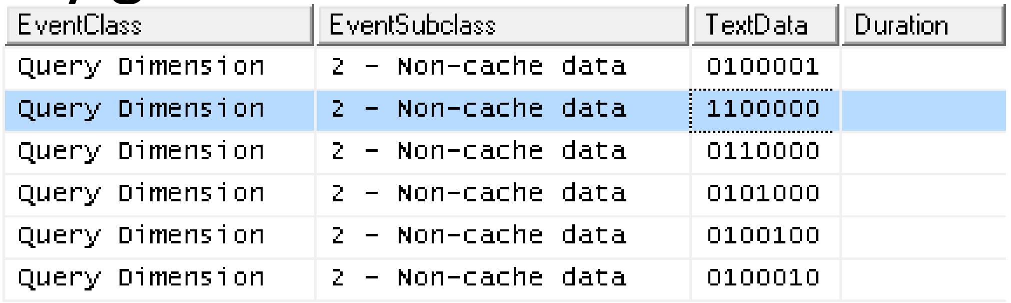


Evaluation of Security

Multidimensional:

Evaluated per Dimension

- Security gets translated to all Attributes → AutoExists



EventClass	EventSubclass	TextData	Duration
Query Dimension	2 - Non-cache data	0100001	
Query Dimension	2 - Non-cache data	1100000	
Query Dimension	2 - Non-cache data	0110000	
Query Dimension	2 - Non-cache data	0101000	
Query Dimension	2 - Non-cache data	0100100	
Query Dimension	2 - Non-cache data	0100010	

- This can take some time!
- Especially for Big Dimensions!

Demo

Evaluation in Multidimensional

Evaluation of Security

Several Issues:

- AutoExists
  - Remove Attributes / Rows
  - Move secured Attributes to separate Dimension → no AutoExists

- Single-Threaded Evaluation
  - Remove Attributes / Rows
  - (Faster CPUs)

- Expensive Allowed/Denied Set
  - Pre-Calculate
  - Cache Sets (Assembly)

Evaluation of Security

Tabular

Simple Expression → at Query Time

- e.g. [Category] = “Bikes”
- AND / OR

Complex Expression → at First Query / Query Time

- DAX
- Username() / CustomData()

Caching of Security

Caching

Multidimensional

	Simple / Fixed	Complex / Dynamic
Evaluated at	First connect	First connect
Cache	Global Scope	Global Scope

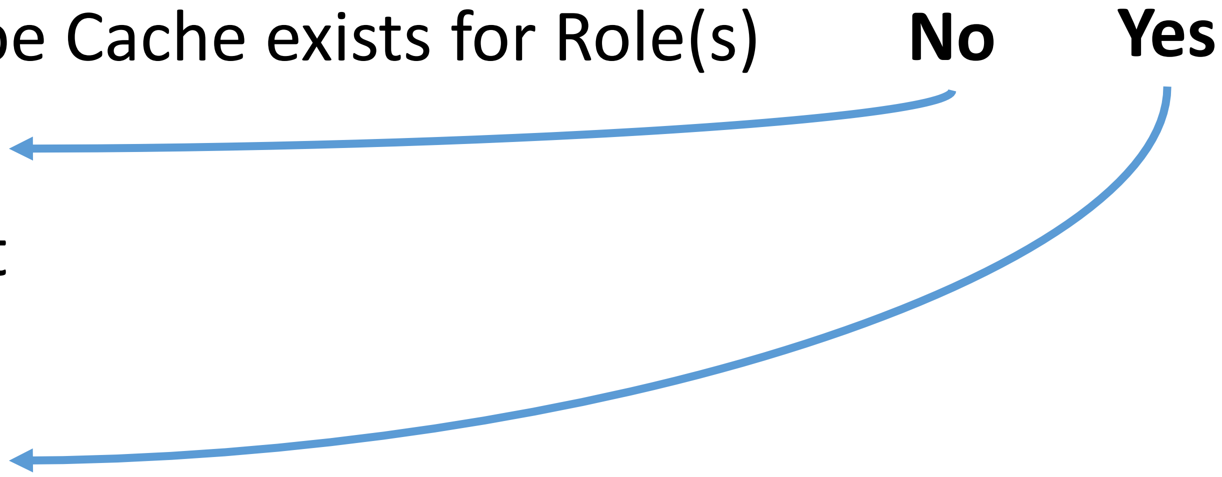
Tabular

	Simple / Fixed	Complex / Dynamic
Evaluated at	Query time	First Query
Cache	No	Yes, RowFilter

Caching – Multidimensional

Global Scope Cache

1) User connects
2) Determine User’s Role(s)
3) Check if Global Scope Cache exists for Role(s)
4) Evaluate Security
5) Evaluate MDX Script
6) …
7) Execute Query



No

Yes

Caching – Multidimensional

Global Scope Cache

For Static Roles:

Cache is shared among Users with same Role(s)

For Dynamic Roles:

Cache is shared if:

- Same Role(s)
- Same UserName()
- Same CustomData()

Demo Caching
Multidimensional

Be aware of the cache size!

Caching – Tabular

In General:

- Only Storage Engine Cache
- No Formula Engine Cache – for DAX Queries
  - But for MDX Queries!

In terms of Security:

- Simple: Pushed down to xVelocity Engine
  - No Caching
- Complex: RowFilter is evaluated once and cached

Demo Caching

Tabular

RowNumber and $RowFilter

[Product Category].$ROWFILTER IN '0x0…004';

$RowFilter

Binary encoded RowNumber

RowNumber

SQLQueryMode=DataKeys

```
SELECT TOP 10 *
FROM [Adventure Works].[$Product Category]
```

RowNumber	Product Category	Binary
0	Bikes	0b0000**1**
1	Components	0b000**1**0
2	Clothing	0b00**1**00
3	Accessories	0b0**1**000

Role	Binary
Bikes + Clothing	0b00**1**0**1**
	0x5

Dynamic Security

Dynamic Security

A Role is dynamic if …

  - USERNAME()
  - CUSTOMDATA()

... is used

Security can be derived from …

- Mapping table
- Assembly (only MD)

Dynamic Security

Using a Mapping table

- Part of the Model
  - MeasureGroup
  - Table

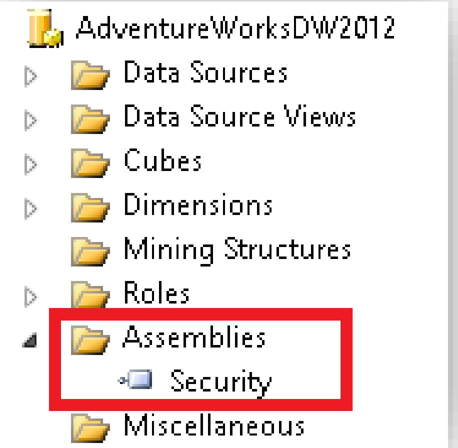
- Strongly tied to Model structures

DomainUser	Company
DOM\User1	A
DOM\User1	B
DOM\User2	A
DOM\User3	C

Dynamic Security

<u>Using an Assembly</u>

- Security is loaded from the outside
  - SQL Table
  - Web Service
- Complex Development
  - C# / .Net
- Great flexibility
  - Not tied to Model structures



AdventureWorksDW2012
- Data Sources
- Data Source Views
- Cubes
- Dimensions
- Mining Structures
- Roles
- Assemblies
  - Security
- Miscellaneous

Dynamic Security

Using an Assembly

```
public static AdomdServer.Set GetAllowedSet(
               string dimension,
               string attribute,
               string username)
{
        // Connect to SQL/WebService/...
        // Pass in Parameters
        // Get ResultSet
        // Convert ResultSet to MDX Set
        // Return MDX Set
}
```

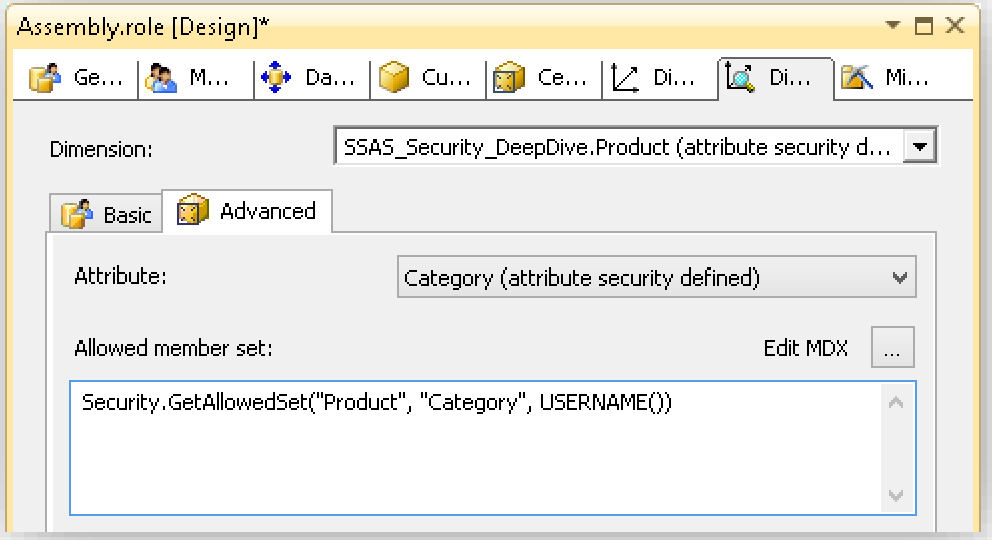
- Compile the DLL

Dynamic Security

Using an Assembly

- Add the DLL to the Project

- Use the Assembly in the role’s security definition



Dimension: SSAS_Security_DeepDive.Product (attribute security d...

Attribute: Category (attribute security defined)

Allowed member set:

```
Security.GetAllowedSet("Product", "Category", USERNAME())
```

MDX Expression for Allowed member set:

Security. GetAllowedSet("Product", "Category", USERNAME())

Dynamic Security

<u>Using an Assembly</u>

Common Pitfalls:

- Assembly throws an Error

- Using an Assembly slows down initial connect

- Security Settings

Demo

Assembly Error

Dynamic Security

Using an Assembly

Initial Connect is slowed down

→ Assembly is called several times

Remember:

“Security gets translated to all Attributes → AutoExists”

→ Assembly is called for each Attribute!

Dynamic Security

<u>Using an Assembly</u>

Single Result/Execution is not cached

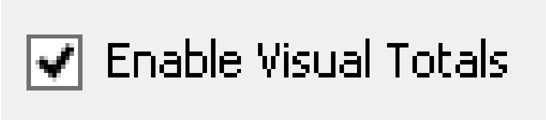
→Cache within Assembly

```
string uniqueToken = username + dimension + attribute;

if (!CachedSecuritySets.TryGetValue(uniqueToken, out s))
{
       s = new Set(...);
       CachedSecuritySets.Add(uniqueToken, s);
}

return s;
```

Dynamic Security



☑ Enable Visual Totals

Multidimensional

Can be set on Attribute-Level

Can be controlled by Assembly

→ only via AMO, not via UI

```
oCubeAttributePermission.VisualTotals = "Security.GetVisualTotals(...)";
```

Tabular

Cannot be controlled → Always On!

Dealing with Multiple Roles

Multiple Roles secured on independent Dimensions

- Combine secured dimensions into one (by Chris Webb)
  - Use Union() + Exists()

- Use an Assembly and CustomData()
  - “Business Role”
  - Only one role at a time

Dealing with Multiple Roles

Assembly and CustomData()

What is a “business role”?

User can select his “business role” using CustomData()

- Connection String
- Custom UI

User has a default “business role”

Dealing with Multiple Roles

Advantages:

- Only one business role at a time
- Flexible, no need to change cube structure
- Change default-role
  - Server-side solution
  - Applied to all client tools
  - (SharePoint Web Part)
- Maintained in SQL

Key Takeaways

- Analysis services offers a rich set of features to secure the cube
  - Use Dimension Security if possible
- Can be further extended by custom code/Assemblies
  - But use with caution!
- Keep evaluation and caching in mind
  - Especially for big cubes/models
- Be aware of the implications of Multiple Roles

Thank You!

Please submit your feedback!
http://www.sqlbits.com/SQLBitsXIISaturday

Gerhard Brueckl

gerhard@gbrueckl.at

http://blog.gbrueckl.at