

Using OVER ... TO clause in MUF expressions

The OVER ... TO clause enables one to write more general MUF expressions which can filter on multiple datasets within the same path. You can dynamically control user access by managing one dataset where the filter mappings are stored (i.e. which user can see what data). Unlike the standard approach, you only need to add new MUF objects for new users without updating the existing MUFs. The OVER ... TO clause applied to a path also minimizes the number of MUFs (or their complexity respectively). This makes MUF handling easier and more straightforward.

Let's assume this simplified data model with 4 fact tables and one many-to-many relation (but it would work with any other model as well):

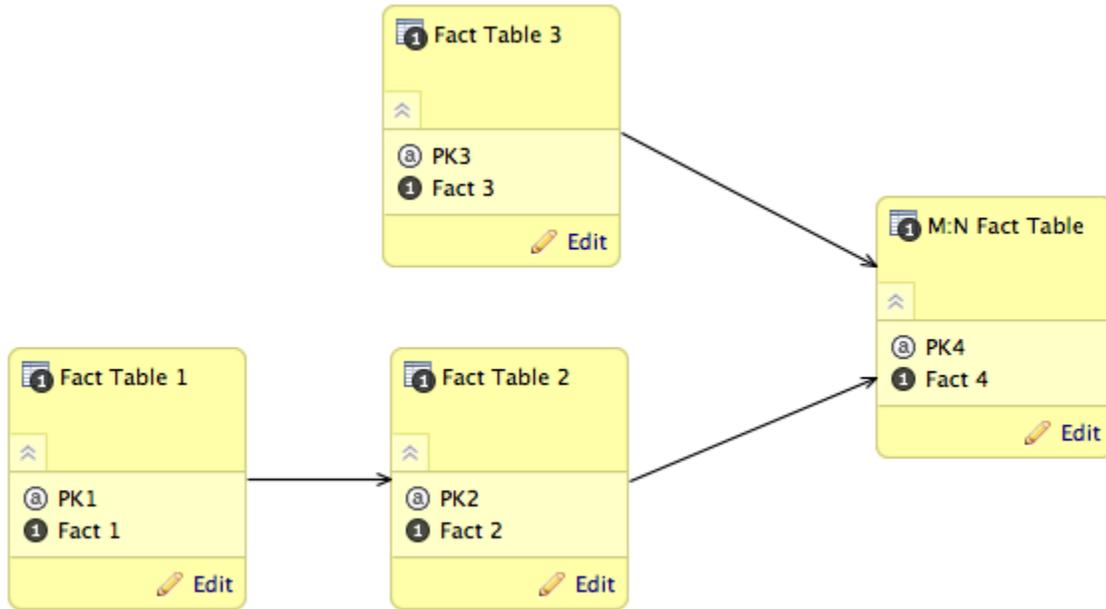


Figure 1

One wants to control data access by connection point of the M:N Table. There are 2 paths that end at the M:N Fact Table, the first one has two nodes (datasets), the second one 3 datasets. Here's an example how to filter data on 3 datasets within the second path by using only one MUF per user.

First, we need to add a mapping dataset to the model where users are assigned to any number of "M:N" records (each record is a combination of a key of Fact Table 2 and a key of Fact Table 3). It is basically another many-to-many relationship - this time between users and identifiers of records they can see.

User ID	User Login	PK4 (connection point of M:N table)
1	one@users.com	acceaaafcabbbe4959bb13e1d73b2f99a9
1	one@users.com	4a52e256a68b4c3893c10c2961ba6ccd
2	two@users.com	5ae83bc8addf47bfb8c00808c30a7d10

The next step is to create a dataset for such mappings. Figure 2 has a User dataset in addition but it isn't necessary.

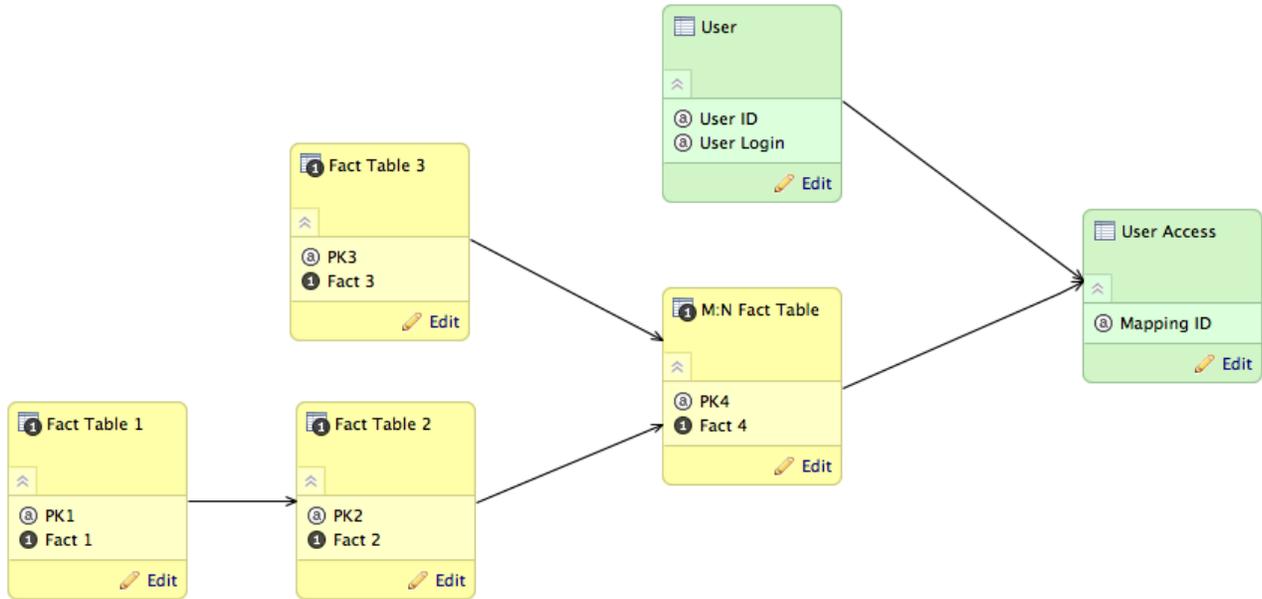


Figure 2

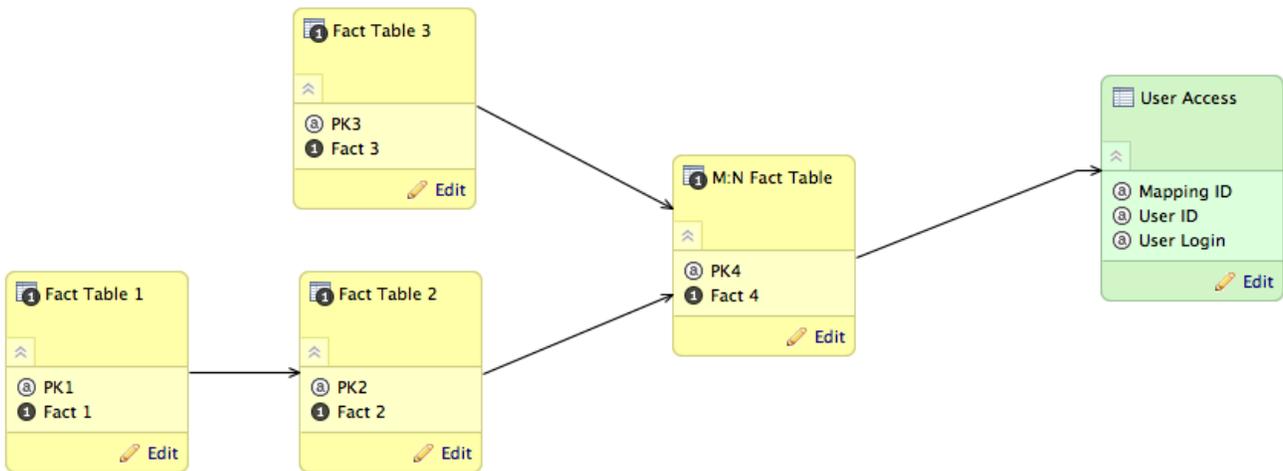


Figure 3

We will use the "User_Access" dataset to filter all parent datasets within the same path (Fact Table 1 Fact Table 2 M:N Fact Table User Access). This can be done by the following MUF expression:

```
(USER_ATTRIBUTE = VALUE) OVER CP_OF_ACCESS_DATASET TO CP_OF_FILTERED_DATASET
```

Note - the condition needs to be in the brackets

To create the MUF we need all the components in form of URIs which is the case of any other muf.

```
([URI_1] = [URI_2]) OVER [URI_3] TO [URI_4]
```

In our scenario, URI_1 will be User ID field (i.e. connection point of User dataset), URI_2 will be concrete user, URI_3 will be a connection point of the mapping dataset (User Access) and URI_4 will be a connection point of Fact Table 1. If we include User fields in User Access dataset, URI_1 will refer to field User ID in the User Access dataset.

Here's the example of entire MUF expression (the parentheses are required).

```
( [/gdc/md/vcs6tmt5prilun2515chz9q0qc5rpd2x/obj/368475]=[/gdc/md/vcs6tmt5prilun2515chz9q0qc5rpd2x/obj/368475/elements?id=3004]) OVER
[/gdc/md/vcs6tmt5prilun2515chz9q0qc5rpd2x/obj/368479] TO
[/gdc/md/vcs6tmt5prilun2515chz9q0qc5rpd2x/obj/7061]
```

To demonstrate what will happen, let's populate the M:N table with some records:

PK4	FK2	FK3
acceaaafcabbe4959bb13e1d73b2f99a9	A1	B2
4a52e256a68b4c3893c10c2961ba6ccd	A2	B2
5ae83bc8addf47bf8c00808c30a7d10	A1	B3

FK2: Connection point of Fact Table 2. FK3: Connection point of Fact Table 3.

As a result, User 1 will see only data related to "acceaaafcabbe4959bb13e1d73b2f99a9" and "4a52e256a68b4c3893c10c2961ba6ccd". This means, that a report with an attribute from the M:N table will show only those 2 records. It works in the same way for both Fact Table 2 and Fact Table 1. In the first case, only records A1 and A2 will be showed. In the latter case, only records associated with A1 and A2 will be showed. In other words, the values are propagated from child dataset to parent datasets (from User Access to Fact Table 1).

It's important to note, that data in the Fact Table 3 won't be filtered since the dataset isn't on the path. When creating a report with an attribute (from the Fact Table 3) and without any facts, all data will be showed to all users. If we want to filter the Fact Table 3 as well, we have 2 options:

- create new MUF with OVER ... TO clause (for path Fact Table 3 M:N Fact Table User Access)
- include the Fact Table 3 in the M:N Fact Table (Figure 4)

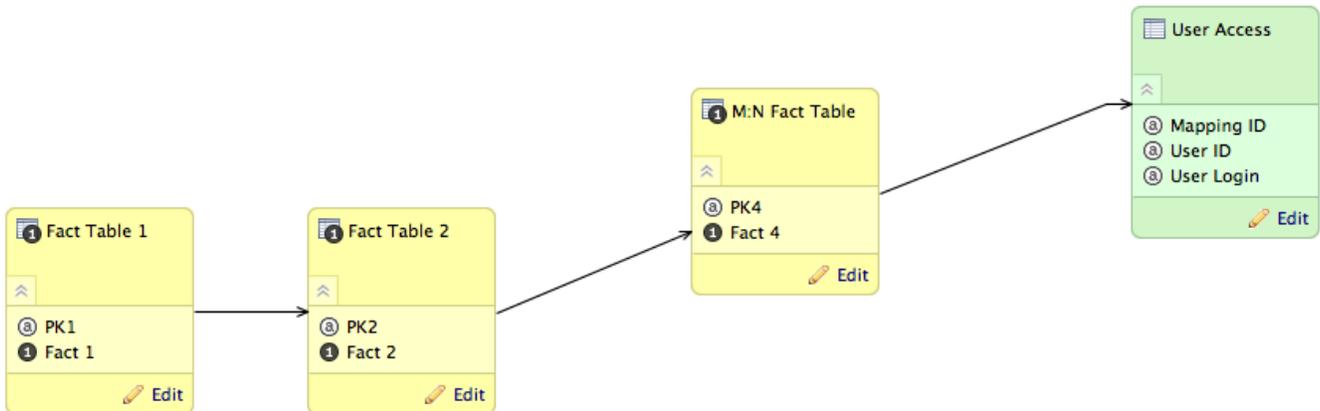


Figure 4