

Improving The User Experience with Global Rates

Almost every planning or forecasting application will have some type of allocation based on a driver or rate that is loaded at a global level. Sometimes these rates are a textbook example of moving data from one department to another based on a driver, and sometimes they are far more complicated. Many times, whether it is an allocation, or a calculation, rates are entered (or loaded) at a higher level than the data it is being applied to.

A very simple example of this would be a tax rate. In most situations, the tax rate is loaded globally and applied to all the departments and business units (as well as level 0 members of the other dimensions). It may be loaded to "No Department", "No Business Unit", and a generic member in the other custom dimensions that exist.

If a user needs the tax rate, in the example above, they have to pull "No Department" and "No Business Unit." Typically, users don't want to take different members in the dimension to get a rate that corresponds to the data (Total Department for taxes, and No Department for the rate). They want to see the tax rate at Total Department, Total Business Unit, and everywhere in-between.

There are a number of ways to improve the experience for the user. An effective solution is to have two members for each rate. One is stored and one is dynamic. There is no adverse effect on the number of blocks, or the block size. The input members can be grouped in a hierarchy that is rarely accessed, and the dynamic member can be housed in a statistics hierarchy.

Using tax rate in the example above, create a "Tax Rate Input" member. Add this to a hierarchy called "Rate Input Members".

Any time data is loaded for the tax rate; it is loaded to Tax Rate Input, No Department, No Business Unit, etc. Under the statistics/memo hierarchy, create a dynamic member called "Tax Rate". "Tax Rate" would be the member referenced in reports.

The formula for this includes a cross-dimensional reference to the "Tax Rate Input" member, and would look something like this.

```
"No Department"->"No Business Unit"->"Tax Rate Input";
```

When a user retrieves "Tax Rate", it always returns the rate that is loaded to "No Department," "No Business Unit," and "Tax Rate Input," no matter what department or business unit the report is set to. The effort involved in creating reports in Financial Reporting or Smart View now becomes easier!

There is an added bonus for the system administrators. Any calculation that uses the rate (you know, the ones with multi-line cross-dimensional references to the rates) is a whole lot easier to write, and a whole lot easier to read because the cross-dimensional references no longer exist.

Before you move the application to production, make sure to set the input rates consolidation method to "Never." Don't expect this change to make great improvements in performance, but it will cause the aggregations to ignore these members when consolidating the hierarchies. A more important benefit is that users won't be confused if they ever do look at the input rates at a rolled up level. The ONLY time they would see the rate would be at level 0, and would be an accurate reflection of the rate.

Note: It is recommended to create member names without spaces. The examples above ignored this rule in an effort to create an article that is more readable.