Additivity of Measures

Additivity of measures is not exactly a data warehouse design problem. However, you have to realize which aggregate functions you will use in reports for which measure, and which aggregate functions you will use when aggregating over which dimension.

The simplest type are measures that can be aggregated with the *SUM* aggregate function across all dimensions, such as amounts or quantities. For example, if sales for product A were $200.00 and sales for product B were $150.00, then the total sales was $350.00. If yesterday’s sales were $100.00 and sales for the day before yesterday was $130.00, then the total sales amounted to $230.00. Measures that can be summarized across all dimensions are called *additive* measures.

Some measures are not additive over any dimension. Examples include prices and percentages, such as a discount percentage. Typically, you use the *AVERAGE* aggregate function for such measures, or you do not aggregate them at all. Such measures are called *non-additive* measures. Often, you can sum additive measures and then calculate non-additive measures from the additive aggregations. For example, you can calculate the sum of sales amount and then divide that value with the sum of order quantity to get the average price. On higher levels of aggregation, the calculated price is the average price; on the lowest level, it’s the data itself—the calculated price is the actual price. This way, you can simplify queries.

For some measures, you can use *SUM* aggregate functions over all dimensions but time. Some examples include levels and balances. Such measures are called *semi-additive* measures. For example, if customer A has $2,000.00 in a bank account, and customer B has $3,000.00, together they have $5,000.00. However, if customer A had $5,000.00 in an account yesterday but has only $2,000.00 today, then customer A obviously does not have $7,000.00 altogether. You should take care how you aggregate such measures in a report. You can use *SUM* aggregate function over all dimensions but time. Over time, you can calculate average value, or use the last value as the aggregate.

SSAS is out of the scope of this book; however, you should know some facts about SSAS if your data warehouse is the source for SSAS databases. SSAS has support for semi-additive and non-additive measures. The SSAS database model is called the *Business Intelligence Semantic Model* (BISM). Compared to the SQL Server database model, BISM includes much additional metadata. SSAS has two storages: *dimensional* and *tabular*. Tabular storage is quicker to develop, because it works through tables like a data warehouse. The dimensional model more properly represents a cube. However, the dimensional model includes even more metadata than tabular. In BISM dimensional processing, SSAS offers semi-additive aggregate functions out of the box. For example, SSAS offers the *LastNonEmpty* aggregate function, which properly uses the *SUM* aggregate function across all dimensions but time, and defines the last known value as the aggregate over time. In BISM tabular, you use the *Data Analysis Expression* (DAX) language. The DAX language includes functions that let you build semi-additive expressions quite quickly as well.