

Kimball Design Tip #42: Combining Periodic And Accumulating Snapshots

By Ralph Kimball

Normally we think of the accumulating snapshot and the periodic snapshot as two different styles of fact tables that we must choose between when we are building a fact table around a data source. Remember that a periodic snapshot (like the monthly summary of a bank account) is a fact table that records activity during a repeating predictable time period. Periodic snapshot records are generally repeated each reporting period as long as the thing being measured (like the account) is in existence. Periodic snapshots are appropriate for long running processes that extend over many reporting periods.

Accumulating snapshots, on the other hand, are used for short processes that have a definite beginning and end, such as an order being filled. For an order, we would usually make a record for each line on the order, and we would revisit the record making updates as the order progressed through the pipeline. The accumulating snapshot is by definition a snapshot of the most recent state of something and therefore the dimensional foreign keys and the facts are, in general, over-written as time progresses.

The simplest implementation of an accumulating snapshot does not give you intermediate points in the history of, for example, an order.

There are at least three ways to capture this intermediate state:

1. Freeze the accumulating snapshots at regular intervals such as month end. These periodic snapshots should probably be in a separate fact table by themselves to keep applications from getting too complicated. Ironically, this approach comes in the back door to mimic a real-time interpretation of a periodic snapshot (where you create a hot rolling current month), but that's another story. The frozen snapshots of the orders can now reflect the use of Type 2 SCDs for the dimensions (like Customer). As in any periodic snapshot, the good news is that you know you have a record for that order each month the order is active. The bad news is that you only see the snapshots of the order at month ends.
2. Freeze the accumulating snapshot and store it in a second fact table if and only if a change to the order occurs. This gives the complete history of an order. It has the same number of records as option 3, below.
3. Maintain a full transaction grain fact table on the order lines. Add a transaction dimension to this fact table to explain each change. This is "fully general" in that you can see every action that has occurred on an order, but be careful. Some of the transactions are not additive over time. For instance, if a line item on an order is cancelled and two other line items are substituted for the original one, it is a complex calculation to correctly reconstruct the order at an arbitrary point in time after these transactions. That's why option #2 may be the best if you need to see every intermediate state of a complete order.

If you are interested in taking a deeper look at all three kinds of fact tables, read my article, "Fundamental Grains," in the Ralph Kimball Group article archived on our website at www.kimballgroup.com. Look in the Advanced Fact Table Topics section.