

Kimball Design Tip #8: Perfectly Partitioning History With The Type 2 Slowly Changing Dimension

By Ralph Kimball

The Type 2 slowly changing dimension (SCD) approach provides a different kind of partitioning. You could call this a logical partitioning of history. In the Type 2 approach, whenever we encounter a change in a dimension record, we issue a new record, and add it to the existing dimension table. A simple example of such a change is a revised product description, where something about the product, such as the packaging type, changes but the basic Stock Number (e.g., the bar code) does not change. As keepers of the data warehouse, we have taken a pledge to perfectly track history and so we must track both the new product description as well as the old.

The Type 2 SCD requires special treatment of the dimension key, the product key in our example. We must assign a generalized key because we can't use the same Stock Number as the key. This gives rise to the whole discussion of assigning anonymous surrogate keys, which has been discussed extensively elsewhere. See the articles www.dbmsmag.com/9805d05.html and www.dbmsmag.com/9806d05.html for the complete story on surrogate keys.

Stop and think for a moment about how you have been using the dimension key up to the point where you make the new dimension record described above. Before today, you have been using the "old" surrogate key whenever you have created a fact table record describing some product activity.

Today, two things happen. First, we assume that the changed packaging type goes into effect with the new fact table data that we receive today. Second, this means that after we create the new dimension record with its new surrogate key, then we use that surrogate key with all of today's new fact records.

WE DON'T GO BACK TO PREVIOUS FACT RECORDS TO CHANGE THE PRODUCT KEY.

The old product dimension record still points correctly to all the previous historical data, and the new product dimension record will now point to today's records and subsequent records, until we are forced to make another Type 2 change.

This is what we mean when we say that the Type 2 SCD perfectly partitions history. If you visualize this, then you really understand this design technique.

Notice that when you constrain on something in the dimension like the product name, which is unaffected by the change in the packaging type attribute, then you automatically pick up both the old and new dimension records, and you automatically join to all of the product history in the fact table. It is only when you constrain or group on the packaging type attribute that SQL smoothly divides the history into the two regimes.

This discussion is the heart of the Type 2 approach. The Type 2 approach can be augmented and made even more powerful by carefully placing time stamps in the dimension, but these time stamps are extra "goodies" that are not needed for the basic partitioning of history. See the article www.dbmsmag.com/9802d05.html for one application of a time stamped SCD.