[](http://www.kimballgroup.com/)

The Microsoft Data Warehouse Toolkit, 2nd Edition,   
with the SQL Server 2008R2 platform

Web Content

Chapter 2 – Example of Type 1 and Type 2 attribute change tracking techniques

The best way to understand the concepts of Type 1 and Type 2 change tracking, and the substantial impact of accurately tracking changes, is with an example. As we see in the Data Mining chapter, Adventure Works Cycles collects demographic information from its Internet customers. These attributes, such as gender, homeowner status, education, and commute distance, all go into the customer dimension. Many of these attributes will change over time. If a customer moves, her commute distance will change. If a customer bought a bike when he went to college, his education status will change when he graduates. These attributes all certainly qualify as slowly changing, but should you treat them as Type 1 or Type 2 attributes?

If you’ve been paying attention, you know the correct answer is to ask the business users how they’ll use this information. You already know from the requirements gathering information in Chapter 1 that Marketing intends to use some data mining techniques to look for demographic attributes that predict buying behaviors. In particular, they want to know if certain attributes are predictive of higher-than-average buying. They could use those attributes to create targeted marketing campaigns. Let’s examine some example customer-level data to see how decisions on handling attribute changes can affect the information you provide your users.

Table 2.1 shows the row in the customer dimension for a customer named Jane Rider as of January 1, 2011. Notice that the customer dimension includes the business key from the transaction system along with the attributes that describe Jane Rider. The business key allows users to tie back to the transaction system if need be.

Table 2.1: Example Customer Dimension Table Row as of Jan 1, 2011

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Customer\_Key | BKCustomer\_ID | Customer\_ Name | Commute\_ Distance | Gender | Home\_ Owner\_ Flag |
| 1552 | 31421 | Jane Rider | 3 | Female | No |

Table 2.2 shows some example rows from an abridged version of the AWC Orders fact table in the data warehouse database. We have used surrogate keys for customer and product, but we left the order date as a date for the sake of clarity. These rows show all of Jane Rider’s AWC orders.

A marketing analyst might want to look at these two tables together, across all customers, to see if any of the customer attributes predict behavior. For example, one might hypothesize that people with short commute distances are more likely to buy nice bikes and accessories than people with long commute distances. They might ride their bikes to work rather than drive.

Table 2.2: Example Order Fact Table Rows for Jane Rider as of Feb 22, 2011

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Customer\_Key | Product\_Key | Item\_Count | Dollar\_Amount |
| 1/7/2009 | 1552 | 95 | 1 | 1,798.00 |
| 3/2/2009 | 1552 | 37 | 1 | 27.95 |
| 5/7/2010 | 1552 | 87 | 2 | 320.26 |
| 8/21/2010 | 1552 | 33 | 2 | 129.99 |
| 2/21/2011 | 1552 | 42 | 1 | 19.95 |

Overwriting Values: The Type 1 Change

A few additional details help demonstrate the importance of appropriately handling attribute changes. To the untrained eye, attributes such as commute distance and homeowner flag might not seem important to the business, so you might treat them as Type 1 attributes and overwrite the existing values when changes occur. As it turns out, Jane took advantage of low mortgage interest rates and moved into a new home on January 2, 2011, changing her commute distance from 3 miles to 31 miles. If you do a Type 1 overwrite, Jane’s commute distance value in Table 2.1 will change to 31, and her homeowner flag will change to Yes.

The timing of the attribute change events is important to understanding their impact on the analysis. If the marketing analyst decided to investigate the relationship between commute distance and orders on March 1, 2011, after the overwrite took place, all of Jane Rider’s orders would be counted as occurring with a commute distance of 31 miles. By overwriting commute distance when a change occurs, you’re no longer tracking the actual commute distance that was in effect at the time the order was placed. Instead, you’re assigning all historical orders to the current attribute value. This means that any time your analysts try to do historical correlation analysis, they risk drawing incorrect conclusions.

Does this matter? Only the business users can say. If there aren’t a significant number of rows that change, or if the attribute is not considered relevant, it probably doesn’t matter if you overwrite the old value when a change occurs. However, you must be extremely careful in making this decision—it requires detailed exploration of the potential impact with knowledgeable business folks.

The short answer is it usually does matter. In the example case, the analyst may find a significant correlation between long commute distances and larger orders. This finding might then be used to design a marketing campaign targeted at people with long commute distances; a campaign that will likely be ineffective. In this case, the errors introduced by the Type 1 overwrite mainly costs money and time. In some cases, as in health care, overwriting attribute values might cost even more.

Tracking Historical Values—the Type 2 Change

Let’s see how the same example data looks if you tracked the changes using the Type 2 approach. This will affect both the fact and dimension tables. Let’s consider the dimension table first: When Jane Rider moves on January 2, 2011, add a new row to the customer dimension, with a new surrogate key, as shown in Table 2.3.

Table 2.3: Example Customer Dimension Table Row as of Jan 2, 2011 with Type 2 Change Tracking

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Customer\_Key | BKCustomer\_Id | Customer\_Name | Commute\_Distance | Gender | Home\_Owner\_Flag | Eff\_Date | End\_Date |
| 1552 | 31421 | Jane Rider | 3 | Female | No | 1/7/2009 | 1/1/2011 |
| 2387 | 31421 | Jane Rider | 31 | Female | Yes | 1/2/2011 | 12/31/9999 |

The customer dimension table has been augmented to help manage the Type 2 process. Two columns have been added to indicate the effective date and end date of each row. This is how you can tell exactly which row was in effect at any given time.

As Table 2.4 shows, the fact table has to change as well because it has a row that occurred after the Type 2 change occurred. The last row of the fact table needs to join to the row in the dimension that was in effect when the order occurred on February 21, 2011. This is the new dimension row with Customer\_Key = 2387.

Table 2.4: Updated Order Fact Table Rows for Jane Rider as of Feb 22, 2011

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Customer\_Key | Product\_Key | Item\_COUNT | Dollar\_Amount |
| 1/7/2009 | 1552 | 95 | 1 | 1,798.00 |
| 3/2/2009 | 1552 | 37 | 1 | 27.95 |
| 5/7/2010 | 1552 | 87 | 2 | 320.26 |
| 8/21/2010 | 1552 | 33 | 2 | 129.99 |
| 2/21/2011 | 2387 | 42 | 1 | 19.95 |

The correct assignment of dimension keys to fact table rows is handled as a matter of course in the ETL process. Integration Services even has a wizard to define this process. We have to go back and update the fact table in our example because we didn’t start with Type 2 attributes in the first place. It would have been much better if we had designed the customer dimension as a Type 2 SCD originally. This should give you a feel for what you will have to do if you don’t use Type 2 change tracking as well.

Now when the marketing analyst does the analysis, the data will show $2,276.20 in orders associated with a 3-mile commute distance and $19.95 associated with a 31-mile commute distance. The way to decide whether to track attribute changes as Type 1 or Type 2 is to ask, “Does it matter?” From the marketing analyst’s perspective in this example, accurately tracking changes over time obviously matters a lot.