HISTORY MANAGEMENT OF DATA – SLOWLY CHANGING DIMENSIONS

Marek Wancerz, Paweł Wancerz
Lublin University of Technology, Faculty of Electrical Engineering and Computer Science

Abstract: *The article describes few methods of managing data history in databases and data marts. There are many types of dealing with the history of the data. This article will show us some examples, point advantages and disadvantages of each of the method and show us possible scenarios of use.*

Keywords: managing data history, databases, data marts

ZARZĄDZANIE HISTORIĄ DANYCH – SLOWLY CHANGING DIMENSIONS

Streszczenie: *Artykuł opisuje sposób zarządzania historią tabel wymiarowych w bazach danych i hurtowniach danych. Istnieje kilka sposobów na archiwizowanie historii. Artykuł ma na celu przybliżenie ich funkcjonalności popartej przykładami, wskazanie zalet i wad oraz możliwych scenariuszy użycia.*

Słowa kluczowe: zarządzanie historią, bazy danych, hurtownie danych

Introduction

Nowadays, almost everyone use data in they lives. But how to understand word „data”? In IT, we can name it as a set of values or variables belonging to a set of items. It is very often represented in a tabular form (columns + rows), data tree (parent-child relationship) or in a graphical structure (tables models with graphical representation). The data we keep doesn't have to be in a text form. We can keep it as a number or even an image.

The data kept in our databases (or data marts) and its quality gives us a huge advantage for the data visualization and management.

But we have to remember that the dimensional data is not a stable entity and it might change over time. To manage the data history Slowly Changing Dimensions was invented.

1. Slowly Changing Dimensions overview

Slowly Changing Dimensions was invented by Ralph Kimball, who is regarded as one of the original architects of data warehousing. His methodology became a standard. Slowly Changing Dimensions is a set of methods to manage the data history in the Dimension tables.

Data might change over time and we should take it into account while developing our system (Fig. 1):

- Source data–source data should be delivered in a unified form,
- ETL (Extract Transform Load) – the ETL system should have the mechanism to operate with data incoming to the target system. All bigger ETL tools have the option SCD already implemented (as an option to use) – Informatica PowerCenter Tool, SSIS, Oracle Warehous Builder,
- Database structure – based on the method of history management, the database structure has to be adapted.

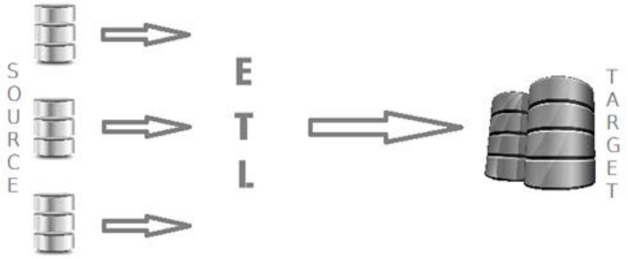


Fig. 1. Data mart load process

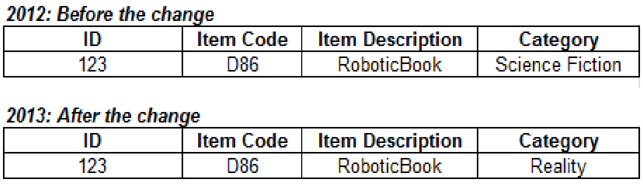
To understand the way of data history management we should take a look on the main methods.

2. Slowly Changing Dimensions – main types

The easiest way to discuss about the Slowly Changing Dimensions types is to go through all of them with some examples, pointing advantages and disadvantages and possible usage scenarios. We have 3 basic types of Slowly Changing Dimensions.

I. SCD Type 1 – overwriting the old values

Figure (Fig. 2) shows the change of data for an item. The Category for an Item RoboticBook was Science Fiction in 2012. In 2013 the Item changed its Category to Reality.



2012: Before the change

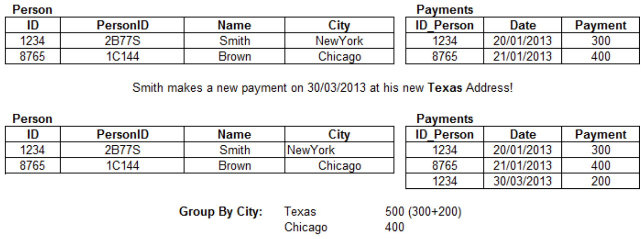
ID	Item Code	Item Description	Category
123	D86	RoboticBook	Science Fiction

2013: After the change

ID	Item Code	Item Description	Category
123	D86	RoboticBook	Reality

Fig. 2. SCD Type 1 overview

The change of Category is a result of an error or a change of structure. Nevertheless it sometimes doesn't meet business requirements. The big advantage for such a method of development is the simplicity of database structure and ETL system. But main disadvantage of this kind of method is the fact that **we lose the data history!** We can only see descriptive attributes as they exist today. To give a better understanding of business requirements inaccuracy after such a change, please have a look the picture below (Fig. 3).



Person

ID	PersonID	Name	City
1234	2B77S	Smith	NewYork
8765	1C144	Brown	Chicago

Payments

ID_Person	Date	Payment
1234	20/01/2013	300
8765	21/01/2013	400

Smith makes a new payment on 30/03/2013 at his new **Texas** Address!

Person

ID	PersonID	Name	City
1234	2B77S	Smith	NewYork
8765	1C144	Brown	Chicago

Payments

ID_Person	Date	Payment
1234	20/01/2013	300
8765	21/01/2013	400
1234	30/03/2013	200

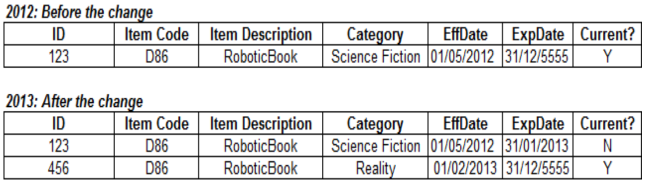
Group By City:	Texas	500 (300+200)
	Chicago	400

Fig. 3. SCD Type 1 example

There are 2 People in dimension Person with their payments in a separate fact table – Payments. Then on 30/03/2013 there is a new payment from the same person – Smith but under a new address (it was changed in the Dimensional table). When we analyze the Payments by City, we will see that Texas has 500 but in fact it has only 200 (300 was NewYork but it is no longer available!).

II. SCD Type 2– new record in the dimension

Let’s take the same example as in Type 1. The same sets of values were assigned to the Item, but with additional fields – Effective Date, Expiration Date, FlagCurrent (Fig. 4).



2012: Before the change

ID	Item Code	Item Description	Category	EffDate	ExpDate	Current?
123	D86	RoboticBook	Science Fiction	01/05/2012	31/12/5555	Y

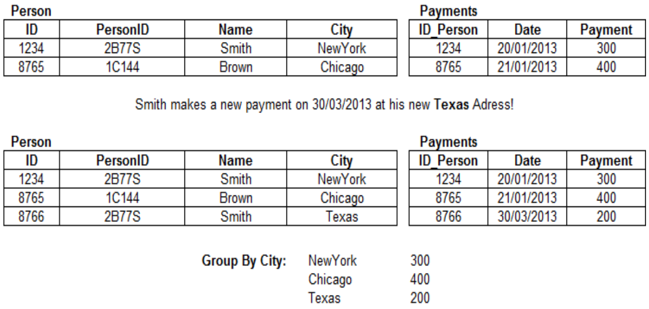
2013: After the change

ID	Item Code	Item Description	Category	EffDate	ExpDate	Current?
123	D86	RoboticBook	Science Fiction	01/05/2012	31/01/2013	N
456	D86	RoboticBook	Reality	01/02/2013	31/12/5555	Y

Fig. 4. SCD Type 2 overview

As we can see, for Type 2 Dimension, we have 3 additional indicators which help us control the data. The dates show us the period of time when the Item is valid. The Current flag is giving us a fast information if the row is Valid (Y) or not (N). The huge advantage for this approach is that we keep all the history rows in the dimension and we track all the historical entries. On the other hand, this approach is more complicated for the end user (report developer). The dimension table growth has also be taken into account while development of the project schema.

Let’s have a look again at the example from SCD Type 1. But here we will use SCD Type 2 for history data management. (Fig. 5)



Person

ID	PersonID	Name	City
1234	2B77S	Smith	NewYork
8765	1C144	Brown	Chicago

Payments

ID_Person	Date	Payment
1234	20/01/2013	300
8765	21/01/2013	400

Smith makes a new payment on 30/03/2013 at his new **Texas** Adress!

Person

ID	PersonID	Name	City
1234	2B77S	Smith	NewYork
8765	1C144	Brown	Chicago
8766	2B77S	Smith	Texas

Payments

ID_Person	Date	Payment
1234	20/01/2013	300
8765	21/01/2013	400
8766	30/03/2013	200

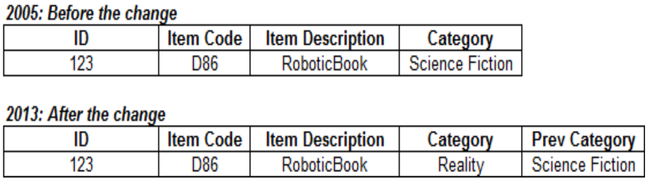
Group By City:		
	NewYork	300
	Chicago	400
	Texas	200

Fig. 5. SCD Type 2 example

This example shows us correct values grouped by Cities. This is because we created a new row for the changed Smith person with updated City.

III. SCD Type 3– new dimension column

Let’s have a look at the last primary SCD – Type 3. The same example will be taken into account while trying to visualize the method. (Fig. 6)



2005: Before the change

ID	Item Code	Item Description	Category
123	D86	RoboticBook	Science Fiction

2013: After the change

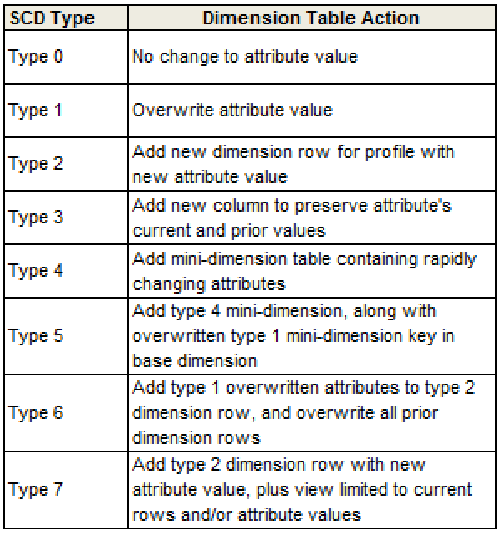
ID	Item Code	Item Description	Category	Prev Category
123	D86	RoboticBook	Reality	Science Fiction

Fig. 6. SCD Type 2 overview

In this method, a new dimension column is created to keep the historical value of the item. This kind of method is used relatively infrequently. Type 3 SCD is good for tracking soft changes, like item or business reorganization. It gives a good view of the situation today and prior the change. But in case of more frequent and important changes this method will lose the historical data, as only current and original values are retained. The history of changes can't be reproduced as it is done in SCD Type 2.

3. Conclusion

We have analyzed 3 types of managing historical data in Dimensional tables. Each of them has advantages and disadvantages and can be used in totally different business needs. It is up the data modeler to set up such an environment to make it easy to implement, maintain and develop. The most popular method (from those 3) is definitely the SCD Type 2 which gives us a full history of a Dimension value and helps us to build reports not only on current but also on the historical data. The types we described together with the whole concept were invented in 90'. But after publish of SCD's 1, 2, 3 Kimball Group started working on modifications of methods, their fusions and new ones. The result is a new book The Data Warehouse Toolkit (Wiley, Jun/Jul 2013) where we can find 7 Types of SCD's! You can check the overview here on the Figure (Fig. 7). I will describe them in the next publication.



SCD Type	Dimension Table Action
Type 0	No change to attribute value
Type 1	Overwrite attribute value
Type 2	Add new dimension row for profile with new attribute value
Type 3	Add new column to preserve attribute's current and prior values
Type 4	Add mini-dimension table containing rapidly changing attributes
Type 5	Add type 4 mini-dimension, along with overwritten type 1 mini-dimension key in base dimension
Type 6	Add type 1 overwritten attributes to type 2 dimension row, and overwrite all prior dimension rows
Type 7	Add type 2 dimension row with new attribute value, plus view limited to current rows and/or attribute values

Fig. 7. New types of SCD [5]

Bibliography

[1] DataModelling: http://www.learndatamodeling.com.

[2] Dimensional Modeling in Depth (Kimball, Ross) – coursebook.

[3] The Data Warehouse Toolkit (Wiley, 2013).

[4] Informatica PowerCenter official: http://www.informatica.com.

[5] Kimball Group webpage: http://www.kimballgroup.com.

Dr inż. Marek Wancerz
e-mail: m.wancerz@pollub.pl

Mark Wancerz graduated from Faculty of Electrical Engineering of the Technical University of Lublin. He currently works in the Department of Network and Security. His research interests revolve around issues of system protection, power system security and the use of information technology and databases in the energy sector. Co-author of many national and international publications.

Mgr inż. Pawel Wancerz
e-mail: pwancerz@gmail.com

Pawel Wancerz is an employee of Atos IT Solutions & Services Company located in Wroclaw. He is currently attending PhD Studies at University of Technology in Lublin. His major interests in IT are Business Intelligence, Data Warehousing and ETL. He participated in many professional courses which enabled him to acquire in-depth knowledge in this field.

otrzymano/received: 14.05.2013 *przyjęto do druku/accepted: 18.07.2013*