# Data Warehouse Schema Evolution Perspectives

Danijela Subotić

University of Rijeka, Department of Informatics, Rijeka, Croatia
dsubotic@inf.uniri.hr

**Abstract.** The paper presents a short analysis of research related to the problem of the data warehouse (DW) evolution. The main contributions of the paper are: a) analysis of existing methods and approaches to the DW schema evolution, and b) characterization of the general research idea for the DW schema evolution problem. The general research idea includes a meta-Data Vault (DV) model that will integrate the DW with the master data management (MDM) system. We believe the following issues could be resolved: a) tracking the origin of data, b) tracking the history of changes, c) avoiding loss of data, d) faster and simpler migration and transformation of data, and e) trend projections. Also, due to long-term storage of historical data and tracking the origin of data, the DW could be used as a complete system of records and as the basis for the data governance (MDM integrated with the DW).

**Keywords:** data warehouse evolution, schema evolution, schema versioning, view maintenance, master data management, data vault.

## 1 Introduction

A data warehouse (DW) integrates a number of heterogeneous data sources and provides for a quick and efficient analysis of business needs. However, today's data sources often change their content and structure, which greatly affects the DW - it should contain the latest data to be able to reflect the evolving state of the real world and our evolving understanding of it. Because of this, it is necessary to properly manage all types of changes and appropriately update the DW. With respect to the literature, the DW evolution research can be grouped into research on managing the data and schema changes in the data warehouse, managing the data changes in the data mart and managing the schema changes in the data mart. However, we will observe the DW schema evolution more broadly, through three approaches - schema evolution, schema versioning and view maintenance. The aim of this paper is to present a short analysis of research related to the problem of the DW schema evolution and to briefly describe our general idea for the future research. The paper is organized as follows; in section 2, research related to the DW evolution is analyzed; in section 3, a characterization of the general research idea is presented; and section 4 contains the conclusion and directions for future work.



N. Bassiliades et al. (eds.), *New Trends in Database and Information Systems II*,
Advances in Intelligent Systems and Computing 312, DOI: 10.1007/978-3-319-10518-5_26

## 2 Analysis of Related Work

We will mainly observe schema evolution and schema versioning approaches in this paper, as they are the most relevant approaches for our future research. More detailed state of the art will not be presented here, as this is only a short paper. The process of schema evolution [1,2,3,4] and versioning [5,6,7,8,9,10,11] is still demanding in terms of invested time and resources. Perhaps the biggest problem here is the preservation of schema consistency and data integrity (there is still a lack of an integrated system-of-records). Also, migration and transformation of data is still slow and expensive, the loss of data during these processes is still present and there is a lack of effective integration, organization and management of metadata. Although the academic community has made steps towards solving these problems, there is still room for improvement, as well as for defining general solutions and fully effective commercial solutions. Different approaches to solving the DW schema evolution problem are presented in literature, but there is still no widely accepted solution for managing DW schema changes. We believe it is necessary to find a simple and complete solution for preserving schema consistency and data integrity. All this issues will serve as requirements for a new approach in which we will try to make a departure from previous research and try to find a new perspective and solution to the DW schema evolution problem.

## 3 General Research Idea

We observe the problem of the schema evolution as a double issue or a dual problem- from the DW perspective, and from the master data management (MDM) perspective. MDM represents the master and reference data (golden copy) and related metadata, set of policies, governance, standards, processes and tools that define and manage the master and reference data (of an organization) to provide a single point of reference, with the purpose of increasing data and business analysis quality. From the DW perspective every fact that is associated with dimensions is observed and star schemas are standard representation used for visualization. From the MDM perspective every master entity (dimension) that is associated with events (facts) is observed and an inverse star-like schema can be used for visualization [16]. In the case of MDM, as in a DW case, there is the problem of schema evolution after changes in the data sources or user requirements, and as such we will address it together. Our general research idea includes a Data Vault meta-model based modeling approach [12,13] that will integrate the DW with the MDM metadata in a common model to serve as an extension of a generic DBMS catalog. The relational model is largely responsible for physical data independence, but we can conclude that it is not convenient to simply and effectively support the evolution of the logical structure of the DB schema. Data Vault (DV) is a data modeling method that supports design of data warehouses for long-term storage of historical data collected from various data sources. Its main advantage is the separation of the structural data from

descriptive attributes, which makes the model flexible to changes in business environment. Also, it highlights the need for tracking the origin of data contained in the database and the history of changes, through empirically defined set of metadata (record_source, load_datetimestamp). Furthermore, any change is implemented in the model as an independent extension of the existing model, which means that the changes do not affect current applications and all versions of the model are a subset of the DV model.

With a development and implementation of our research idea we expect that the following issues will be resolved (mainly through the use of a DV modeling method): a) tracking the origin of data, b) tracking the history of changes, c) avoiding the loss of data, d) slow and expensive migration and transformation of data, and e) trend projections. Due to the long-term storage of historical data and tracking the origin of data, the preservation of DW integrity would be facilitated and the DW would then contain both a "single version of the fact" [12] and a "single version of the truth" [14]. Also, it could be used as a complete and integrated dual solution and system of records [15] with the support for data sources evolution, user requirements evolution and data security evolution.

### 3.1 Characterization of the General Research Idea

Figure 1 shows a diagram of the proposed DW architecture. The proposed architecture consists of four parts: a) data sources, b) enterprise DW, c) reporting DW and d) user analysis. Data sources are usually distinguished (in the literature and practice) by their place of origin and maintenance. Accordingly, we make a distinction between internal and external reference data (which is obtained from internal or external data sources). Reporting DW (RDW) consists of a derivative (summarized, aggregated and computed) data stored in materialized or virtual DM. User analysis is the user side of the system architecture

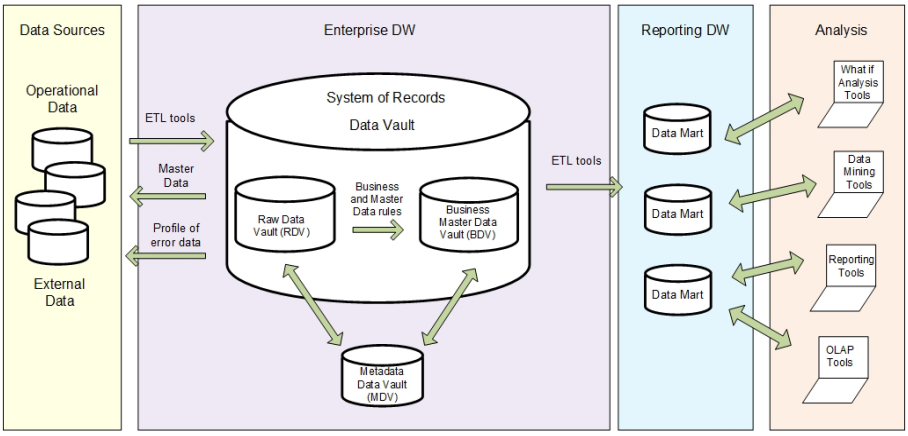


**Fig. 1.** Suggested Data Warehouse Architecture Diagram

where the tools for analysis and reporting are located. With the help of these tools the user is directly accessing the RDW. Our research will mainly focus on the Enterprise DW part of the proposed architecture. Enterprise DW (EDW) includes the raw data vault (RDV) and business master data vault (BDV), which are integrated via metadata data vault (MDV). The RDV is oriented toward data sources. With the help of ETL tools and processes data is extracted and loaded into the RDV. The RDV contains actual copies of the originals from the data sources. Once the data is entered into the RDV, it is no longer deleted (all changes are implemented through additions-only). This means that copies of the originals are kept permanently in RDV. With this persistent preservation of history the loss of data is avoided and a solid basis for the audit process is provided. The BDV is created by upgrading the RDV with the application of standardized master data and business rules, for the purpose of business integration. The RDV and the BDV are shown as separate systems in Figure 1, but that is only a logical representation. Physically it is possible to implement them individually or as a single system. Our EDW will consist of a single DV model which is partially oriented towards the data sources side (RDV), and partially oriented towards the MDM and reporting side (BDV). Also, because they are now physically separated, we can distinguish reversible (light) and irreversible (heavy) transformations [15]. Reversible transformations are used for loading data from a data source into the RDV, and it is possible to reverse their effects, in order to obtain the system-of records. They allow RDV to reach the exact copies of the original from the data source. Irreversible transformations are mainly based on the business and master rules and are usually irreversible. In this case, both the transformations and the original data must be preserved in order to trace exits back to the source and reconstruct them if necessary. The proposed architecture moves irreversible transformations downstream - after RDV, towards BDV and reporting DW (they are loading materialized DMs). Because of those two types of separation (raw/business data and light/heavy transformations), we can target only the needed set of data at a given time which will reduce the time and the cost of data migration and transformation processes, and we can preserve the whole history of changes (of the schema and the data) in the EDW. This is the key idea for getting an integrated EDW system of records which will also serve as the basis for the data governance [15,18]. Finally, MDV is a key component of the proposed architecture, which serves to integrate the RDV and BDV in the EDW system of records (we can say that MDV is the DW on the DW). MDV logical model is based on the DV model. The aim of our research is to define and formalize the MDV model and a final set of change cases for the DW and MDM schema evolution scenarios (DW and MDM – dual solution). We have a preliminary, true working draft version of the MDV for now, but it will not be shown here. However MDV will keep historicized hubs, links, satellites, attributes, domains and reference tables as a hub in a DV model, which will serve as a mechanism for monitoring the data sources evolution. We also plan to resolve (at the meta-level) the transformation from a source into the RDV (via the extraction rules and light transformations), the transformation from the

RDV into the BDV (via the business and master rules and heavy transformations) and finally, the transformation from the integrated RDV/BDV into the DM (also via the business rules and heavy transformations). This way MDV will provide a mechanism for monitoring the user requirements evolution. Furthermore on a meta-level, the security aspect will be resolved, i.e. the user's access rights will be historicized and managed so a mechanism for monitoring the data security evolution will be provided.

## 4 Conclusion

The paper presents a short analysis of previous DW schema evolution research and characterization of our general research idea. The DW evolution process is still quite complex, error prone and requires a lot of time and resources. There is still room for research and improvement regarding the process of: a) transformation and migration of a DB on a new schema (system overload and system downtime are still present, and these processes are still slow and expensive), b) preservation of information during migration (the data is often lost, which affects schema consistency and data integrity), c) rewriting queries and applications in order to run on the new schema (current solutions have high maintenance cost) and d) effective integration, organization and management of metadata. Also the lack of defined mechanism for monitoring the data source model evolution, the lack of support for model relativism and the lack of support for the data security evolution can be noticed. Furthermore, only few of the approaches aim to solve all these DW evolution problems together - the majority focus on just one aspect. We believe that the problem of the DW and MDM schema evolution must be addressed at the general level and that all these problems must be dealt with together - as a dual solution. We are not resolving the problem of the DW and the MDM schema evolution "on the fly" at the operational level (where it occurs), but suggest a permanent general solution situated on a higher (meta) level using the DV model. A key component here is MDV (metadata repository model - metadata data vault) which in this context can be observed as a DW on the DW. MDV will serve to integrate raw and persistent DW with the business aligned MDM in order to obtain one consolidated EDW system of records. We expect the end result to be a flexible, modular, general solution which will track and manage changes in data and metadata, as well as their schemas. We just started our research, after completing state of the art review, and are working on a meta-Data Vault model and general requirements for desired dual solution model. Among directions of ongoing and/or future research are development and formalization of a final set of evolution change cases for the proposed architecture, formalization of a data vault based metadata catalog and incremental development of an Meta-Data Vault implementation prototype with various aspects included (source, rules/transformation, materialized view for DMs, and security) for an empirical evaluation of a proposed solution, as well as an experimental benchmark test based on a completed case study model.

**Acknowledgements.** This paper is based upon work supported by the University of Rijeka under project titled "*Metode i modeli za dizajn i evoluciju skladita podataka*".

## References

1. Hurtado, C.A., Mendelzon, A.O., Vaisman, A.: Maintaining Data Cubes under Dimension Updates. In: Proceedings of the 15th International Conference on Data Engineering (ICDE), Sydney, Australia, pp. 346–355 (1999)
2. Blaschka, M., Sapia, C., Höfling, G.: On Schema Evolution in Multidimensional Databases. In: Mohania, M., Tjoa, A.M. (eds.) DaWaK 1999. LNCS, vol. 1676, pp. 153–164. Springer, Heidelberg (1999)
3. Marotta, A., Ruggia, R.: Data Warehouse Design: A Schema-Transformation Approach. In: 22nd International Conference of the Chilean Computer Science Society (SCCC), Copiapo, Chile (2002)
4. Fan, H., Poulovassilis, A.: Schema Evolution in Data Warehousing Environments – A Schema Transformation-based Approach. In: Atzeni, P., Chu, W., Lu, H., Zhou, S., Ling, T.-W. (eds.) ER 2004. LNCS, vol. 3288, pp. 639–653. Springer, Heidelberg (2004)
5. Eder, J., Koncilla, C.: Evolution of Dimension Data in Temporal Data Warehouses. Technical Report (2000)
6. Body, M., Miquel, M., Bedard, Y., Tchounikine, A.: A Multidimensional and Multiversion Structure for OLAP Applications. In: 5th ACM International Workshop on Data Warehousing and OLAP (DOLAP 2002), McLean, Virginia, USA, pp. 1–6. ACM Press (2002)
7. Golfarelli, M., Lechtenbörger, J., Rizzi, S., Vossen, G.: Schema Versioning in Data Warehouses. In: Wang, S., et al. (eds.) ER Workshops 2004. LNCS, vol. 3289, pp. 415–428. Springer, Heidelberg (2004)
8. Bebel, B., Eder, J., Koncilia, C., Morzy, T., Wrembel, R.: Creation and Management of Versions in Multiversion Data Warehouse. In: 19th ACM Symposium on Applied Computing (SAC 2004), Nicosia, Cyprus, pp. 717–723. ACM Press (2004)
9. Papastefanatos, G., Vassiliadis, P., Simitsis, A., Vassiliou, Y.: What-if Analysis for Data Warehouse Evolution. In: Song, I.-Y., Eder, J., Nguyen, T.M. (eds.) DaWaK 2007. LNCS, vol. 4654, pp. 23–33. Springer, Heidelberg (2007)
10. Solodovnikova, D.: Data Warehouse Evolution Framework. In: Proceedings of the Spring Young Researcher's Colloquium on Database and Information Systems, SYRCoDIS, Moscow, Russia (2007)
11. Malinowski, E., Zimányi, E.: A conceptual model for temporal data warehouses and its transformation to the ER and the object-relational models. Data & Knowledge Engineering 64, 101–133 (2008)
12. Linstedt, D.: SuperCharge Your Data Watehouse: Invaluable Data Modeling Rules to Implement Your Data Vault. CreateSpace Independent Publishing Platform, USA (2011)
13. Jovanovic, V., Bojiéic, I.: Conceptual Data Vault Model. In: Proceedings of the Southern Association for Information Systems Conference, Atlanta, USA (2012)
14. Inmon, W.H., Strauss, D., Neushloss, G.: DW 2.0: The Architecture for the Next Generation of Data Warehousing. Morgan Kaufmann Publishers, Burlington (2008)
15. Jovanovic, V., Bojiéic, I., Knowles, C., Pavlic, M.: Persistent Staging Area Models For Data Warehouses. Issues in Information Systems 13(1), 121–132 (2012)
16. Berson, A., Dubbov, L.: Master Data management and Data Governance, 2nd edn. McGraw Hill (2011)