

Quality metrics Validation in View Maintenance Models of Data Warehouse

Anjana Gosain
University School of Information
Technology
GGS Indraprastha University
Delhi, India

Sangeeta Sabharwal
Netaji Subhas Institute of
Technology
Delhi University
Delhi, India

Rolly Gupta
Netaji Subhas Institute of
Technology
Delhi University
Delhi, India

ABSTRACT

Data warehouses are huge repositories designed to enable the knowledge workers to make better and faster decisions. Due to its significance in strategic decision making, there is a need to assure data warehouse quality in the presence of evolution events which may be generated as result of change in schema / software or data warehouse requirements. One of the factors affecting the data warehouse quality is view maintenance models quality. Although there are some useful guidelines for designing good view maintenance models, but objective indicators, i.e., metrics are needed to help designers to develop quality view maintenance models. In our previous work, a quality metric for View maintenance models of data warehouse is proposed [25] However, the proposal overall lacks theoretical and empirical validation of the metric proposed. Hence, the metric practical utility could not be established. This paper validates the metrics both theoretically and empirically. The theoretical validation is performed using Zuse framework [7] while empirical validation is carried out using MVPP (Multiple View Processing Plan) to explore the relationship between the proposed metrics and cost efficiency of View maintenance models. The results show that all the four metrics NBR, NVM, NAMV and NFMV have significant impact on the cost efficiency of View maintenance models.

General Terms

Zuse framework, MVPP, Validation.

Keywords

Data Warehouse, Data Warehouse Evolution, View maintenance models.

1. INTRODUCTION

Having accurate and up-to-date data warehouses is essential for Business Intelligence and Decision Support. A data warehouse (DW) design, apart from performance guarantees, should also provide correctness guarantees. Every time an evolution event occurs anywhere in the warehouse environment (e.g., a design change at the operational sources) it should be smoothly absorbed without causing any further inconvenience. For achieving this, the warehouse and its counterparts should be easily maintainable and the process of populating it should not be destructed by evolution events.

The Evolution of a DW constitute the backbone of a typical data warehouse architecture. Most of the research for improving DW evolution designs has focused solely on improving performance. However, based on practical experience, maintenance makes up for up to 60 % of the resources spent in a warehouse project [34], and therefore, maintainability is an important factor for the determination of the quality of a design [19,36]. Although practitioners are well aware of this problem, still, we miss a formal and concrete

answer to deals with the quality perspective for View Maintenance models of DW. George et.al. [49] proposed a set of metrics for the evaluation of the vulnerability warehouse modules to future changes and for the assessment of the quality of alternative designs of the warehouse. These proposed metrics are based on graph-theoretic properties of the warehouse graph to assess the sensitivity of the graph to a set of possible events.

Most of the researches consider only structural properties of the DW evolution or constructs internal to the underlying databases. But the employed approach neither accounts for the constructs surrounding the database into their models, nor the fact that a software construct evolves over time. In practice, the problem is hard since changes in the schema of database-centric systems affect not only its internals but also the surrounding deployed applications. Hence, the minimal interdependence of these software modules results in higher tolerance to subsequent changes and should be measured with a principled theory. Related work for evolution of data-intensive applications [9], view redefinition [10,17,26], and data warehouse evolution [4,5,11,14] has provided rewriting techniques and theoretical cost models.

Related work [30] includes an approach to impact analysis and management of schema evolution, which represents the structural properties of the data warehouse schema, along with any views and queries defined over this schema, as a graph . Graph-based model [49] captures all the parts (or, modules) of an environment, i.e., relations, views, and queries (which are practically the parts of ETL scripts that work the underlying data, or the elementary activities of a GUI based scenario that are involved in the ETL process). In [59] authors focused on a set of graph-theoretic metrics for the prediction of evolution impact and their fitness into real-world ETL scenarios. However, these research works are dealing at design level only without considering the maintenance and quality perspective for VM models of DW.

Following this consideration, we have defined a set of metrics for the VM models of DW [25]. In this paper, we built upon the aforementioned approaches with the goal of theoretically and empirically validating the proposed metrics. We formally present these metrics and show that such metrics typically act as predictors for the vulnerability of a VM module of a data warehouse (e.g., a dimension table, or external, an aggregated measure etc.) to future changes to the structure of the warehouse. Secondly, they facilitate the assessment of the quality of alternative designs of the data warehouse VM models with a particular viewpoint on the evolution of the data warehouse.

Theoretical validation assures that the metric is correctly defined and the metric actually measures what it purports to

measure [26]. The theoretical validation helps us to know when and how to apply the metrics. In this work, we will be characterising our metrics using Zuse framework [7].

Empirical validation involves carrying out case studies or controlled experiments etc. to prove practical utility of the metrics. In this work, empirical validation is performed using MVPP (Multiple View Processing Plan – A Cost Based approach) [11]. The proposed metrics have been defined for measuring the cost efficiency of data warehouse VM models.

The remainder of this paper is structured as follows: Section 2 presents proposed metrics. Section 3 summarizes the theoretical validation of the proposed metrics. Section 4 deals with the empirical validation of the proposed metrics and threats. Finally, Section 5 draws conclusions and sketches the immediate future works arising from the conclusions reached in this work.

2. PROPOSED METRICS

Metrics should be defined on the basis of clear measurement goals. So, we have defined a set of metrics for the VM models of DW. The proposed sets of metrics are dealing with two major characteristics. Firstly, they act as predictors for the vulnerability of a VM module of a data warehouse (e.g., a dimension table, or external, an aggregated measure etc.) to future changes to the structure of the warehouse. Secondly, they facilitate the assessment of the quality of alternative designs of the data warehouse VM models with a particular viewpoint on the evolution of the data warehouse. The proposed metrics have been defined for measuring the cost efficiency of data warehouse VM models [25].

1. Number of Base relations: These metric counts the number of base Relations in the view maintenance models for data warehouse.
2. Number of Views materialized: These metric counts the Number of Views Materialized in the view maintenance models for data warehouse.
3. Number of Attributes in materialized views: This metric records the total number of attributes considered for data warehouse VMM.
4. Number of Foreign Keys in materialized views: This metric counts the number of Foreign keys in data warehouse VMM.

3. THEORETICAL VALIDATION OF METRICS USING ZUSE FRAMEWORK

Zuse (1998) describes measurement as a detour. Measurement is more than producing numbers, it is the combination of

empirical entities with numerical entities. These empirical relations can be denoted with the symbols “ $\cdot >$ ” and “ $\cdot >=$ ”, respectively. An empirical relational system is a triple: $A = (A, \cdot >, o)$

where A is a non-empty set of objects, $\cdot >=$ is an empirical relation on A , and o is a closed binary (concatenation) operation on A . In many cases we are not able to produce directly relevant empirical results due to the difficulty of the question we deal with. With the aid of mathematics and statistics “this intelligence barrier” can be overcome: the empirical objects and relationships are mapped into proper numerical objects and relationships. A numerical relational system can be defined as $B = (R, >=, +)$, where R are the real numbers, $>=$ a relation on R , and $+$ a closed binary operation on R . A measure is then a mapping $u : A \rightarrow R$ such that

$$a \cdot >= b \Leftrightarrow u(a) >= u(b); \forall a, b \in A$$

Once the mapping is established, mathematics and statistics can then be used to process the information (e.g., working out means or variances). Measurement theory also leads to conditions where numerical statements can be translated back into empirical statements. To check whether the measure satisfies the users needs, Zuse proposes an internal validation, based on the comparison between the empirical interpretation of numbers and the empirical statements in the real world. The combination rule must be defined as:

$$U(A1oA2) = f(u(A1), u(A2))$$

where $A1, A2, A1oA2 \in A$ and $f(A1, A2) : A \times A \rightarrow A$. This concatenation operation (o) can be contra-intuitive in the area of software engineering because it is not necessary to combine objects in reality. However, it provides a means for building up complex measurement structures, giving a more precise interpretation of numbers.

On this framework, Zuse defines a set of axioms for measures which gives rise to distinct structures. In Table 1, we present the most important ones. In software measurement, there are sufficient frameworks with the next five scale types that are defined by admissible transformations. They are, in hierarchical order: nominal, ordinal, interval, ratio and absolute. Each scale type is defined by admissible transformations. Software measurement starts with the ordinal scale. Measures may be classified in a scale type, depending on whether or not they assume an extensive structure. When a measure accomplishes this structure, it also accomplishes the independence conditions and can be used on the ratio scale levels. If a measure does not satisfy the modified extensive structure, the combination rule will exist or not depending on the independence conditions. When the independence conditions is assumed, but the modified extensive structure is not assumed, such type of scale is the ordinal scale.

Table1: Zuse Framework [7]

Modified extensive structure	Independence conditions	Modified relation of belief
Axiom1: $(A, \bullet \succsim)$ (weak order)	C1: $A1 = A2 \Rightarrow A1 \circ A = A2 \circ A$ and $A1 = A2 \Rightarrow A \circ A1 = A \circ A2$	MRB1: $\forall A, B \in \mathcal{F} : A \bullet \succsim B$ or $B \bullet \succsim A$ (completeness)
Axiom2: $A1 \circ A2 \bullet \succsim A1$ (positivity)	C2: $A1 = A2 \Leftrightarrow A1 \circ A = A2 \circ A$ and $A1 = A2 \Leftrightarrow A \circ A1 = A \circ A2$	MRB2: $\forall A, B, C \in \mathcal{F} : A \bullet \succsim B$ and $B \bullet \succsim C \Rightarrow A \bullet \succsim C$ (transitivity)
Axiom3: $A1 \circ (A2 \circ A3) = (A1 \circ A2) \circ A3$ (weak associativity)	C3: $A1 \bullet \succsim A2 \Rightarrow A1 \circ A \bullet \succsim A2 \circ A$, and $A1 \bullet \succsim A2 \Rightarrow A \circ A1 \bullet \succsim A \circ A2$	MRB3: $\forall A \subseteq B \Rightarrow A \bullet \succsim B$ (dominance axiom)
Axiom4: $A1 \circ A2 = A2 \circ A1$ (weak commutativity)	C4: $A1 \bullet \succsim A2 \Leftrightarrow A1 \circ A \bullet \succsim A2 \circ A$, and $A1 \bullet \succsim A2 \Leftrightarrow A \circ A1 \bullet \succsim A \circ A2$	MRB4: $\forall (A \supset B, A \cap C = \phi) \Rightarrow (A \bullet \succsim B \Rightarrow AUC \bullet \succsim BUC)$ (partial monotonicity)
Axiom5: $A1 \bullet \succsim A2 \Rightarrow A1 \circ A \bullet \succsim A2 \circ A$ (weak monotonicity)		MRB5: $\forall A \in \mathcal{F} : A \bullet \succsim 0$ (positivity)
Axiom6: If $A3 \bullet \succ A4$ then for any $A1, A2$, then there exists a natural number n , such that $A1 \circ nA3 \bullet \succ A2 \circ nA4$ (Archimedean axiom)		
As we know, binary relation $\bullet \succsim$ is called weak order if it is transitive and complete: $A1 \bullet \succsim A2$, and $A2 \bullet \succsim A3 \Rightarrow A1 \bullet \succsim A3$ $A1 \bullet \succsim A2$ or $A2 \bullet \succsim A1$	Where $A1 = A2$ if and only if $A1 \bullet \succsim A2$ and $A2 \bullet \succsim A1$, and $A1 \bullet A2$ if and only if $A1 \bullet \succsim A2$ and not $(A2 \bullet \succsim A1)$.	

In the following section, we adapt this framework to databases to verify the fulfilment of the axioms for the metrics proposed.

3.1 Characterization of metrics

In relational database systems, and for our purposes, the empirical relational system could be defined as $T = (T, \bullet \succsim, \circ)$

where T is a non-empty set of relations (tables), $\bullet \succsim$ is the empirical relation “more or equal complex than” on T , while \circ is a closed binary (concatenation) operation on T . In our case we will choose “natural join” as the concatenation operation.

Natural join is defined generally as (Elmasri and Navathe (1999)):

$$Q \rightarrow T_{((list1)^*(list2))} S$$

where (list1) specifies a list of i attributes of T and (list2) is a list of i attributes of S . These lists are used in order to make the comparison equality conditions between pairs of attributes. These conditions are afterwards related with the AND operator. Only the list corresponding to the T relation is preserved in Q . All these characteristics of the natural join will be useful to design the combination rule of the metrics.

1. Number of attributes in materialized views:

The NA measure is a mapping: $NA : T \rightarrow R$ such that the following holds for all relations T_i and $T_j \in T : T_i \bullet \succsim T_j \Leftrightarrow NA(T_i) \succsim NA(T_j)$.

So, the combination rule for NA can be defined as:

$$NA(T_i \circ T_j) = NA(T_i) + NA(T_j) - NA(T_i \cap T_j)$$

where $NA(T_i \cap T_j)$ is the number of attributes which are common to (belong to the intersection of) T_i and T_j .

Making the formal verification, NA can be characterized as a measure above the level of the ordinal scale, assuming the modified relation of belief.

2. Number of Foreign Key in materialized views:

The RD measure is a mapping: $RD : T \rightarrow R$ such that the following holds for all relations T_i and $T_j \in T : T_i \bullet \succsim T_j \Leftrightarrow RD(T_i) \succsim RD(T_j)$.

In order to obtain the combination rule for RD, we have to consider if the concatenation (by natural join) between tables is made by foreign key, i.e. the number of foreign keys affected (decreasing in one), and not affected in other cases.

So, we can characterize the combination rule for RD as

$$RD(T_i \circ T_j) = RD(T_i) + RD(T_j) - v$$

The formal validation, therefore is analogous to that previously presented. So, in summary, we can characterize RD as a measure above the level of the ordinal scale, assuming the modified relation of belief.

3. Number of Views Materialized:

The NV measure is a mapping: $NV: T \rightarrow R$ such that the following holds for all relations T_i and $T_j \in T$: $T_i \bullet \succeq T_j \Leftrightarrow NV(T_i) \succeq NV(T_j)$.

So, the combination rule for NV can be defined as:

$$NV(T_i \cap T_j) = NV(T_i) + NV(T_j) - NV(T_i \cap T_j)$$

where $NV(T_i \cap T_j)$ is the number of attributes which are common to (belong to the intersection of) T_i and T_j .

Making the formal verification, NV can be characterized as a measure above the level of the ordinal scale, assuming the modified relation of belief.

4. Number of Base Relations:

The BR measure is a mapping: $BR: T \rightarrow R$ such that the following holds for all relations T_i and $T_j \in T$: $T_i \bullet \succeq T_j \Leftrightarrow BR(T_i) \succeq BR(T_j)$.

So, the combination rule for BR can be defined as:

$$BR(T_i \cap T_j) = BR(T_i) + BR(T_j)$$

Making the formal verification, BR can be characterized as a measure above the level of the ordinal scale, assuming the modified relation of belief.

4. EMPIRICAL VALIDATION

After proving theoretical validity of metrics defined in Section 3, in this section, we intend to corroborate that these metrics are really related to cost efficiency of View Maintenance Models of Data Warehouse, i.e. measure of internal consistency will be applied. This is done on the basis of empirical validation. We have conducted controlled experiment for this reason. The dependent variable in this study is cost efficiency and the independent variables are the View Maintenance metrics defined in Section 3.

Twenty One multidimensional schemas were collected to carry out this controlled experiment. The domains of these selected models were general and well known in order to avoid the problems arising out of domain understanding. The metrics values for each schema were collected manually using MVPP approach [11]. There is no subjectivity in calculation of metrics values, as the metrics considered in this work are calculated by simply counting (e.g. NBR is calculated by counting the base relations present in the schemas). The collected data was shown in Table2.

Table 2: Metrics Values

NBR	NVM	NAMV	NFMV	Cost (Query processing and Maintenance cost)
23	13	46	3	70
24	13	46	2	75
31	9	47	6	80
27	5	35	6	63
18	7	30	3	41
24	7	37	5	54
40	8	54	7	91
46	9	64	9	95
22	9	38	5	56
18	5	28	2	35
14	5	23	1	37
13	7	25	1	39
19	9	34	3	45
23	10	42	4	66
26	13	49	3	62

44	16	75	6	98
23	13	25	2	44
17	11	38	3	43
26	5	35	4	48
27	11	49	3	64
38	12	60	6	88

A scatterplot is plotted between the dependent and independent variables. It is aimed to display a relationship between two sets of variables. Depending on the fact, that how tightly the points cluster together, a trend can be discerned.

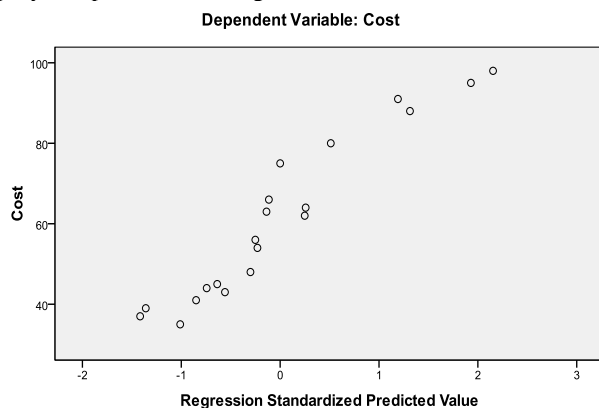


Figure 1: Scatterplot

In figure 1, the variable data points make a straight line, starting from near the origin upto the high y-values. This shows that there exists a positive correlation between the set of variables. The data points are closer enough, to make a line, showing higher correlation. It is therefore interpreted that there is statistically significant correlation between the metrics and the cost efficiency parameter of the data warehouse View Maintenance models. So, it is necessary to investigate the quantitative impact of View Maintenance metrics on the dependent variable.

5. CONCLUSION AND FUTURE WORK

In this work, we validated the metrics, to to evaluate the cost efficiency of VM models for data warehouse. These metrics are theoretically validated using Zuse’s framework [7] and it is concluded that these metrics are theoretically sound and are characterised as either size or length measure. We have also conducted controlled experiment in order to provide the empirical validation of the proposed metrics. The empirical validation using controlled experiments suggests that the metrics have significant effect on the cost efficiency parameter.

Hence, this study shows that these metrics are significantly contributing towards the cost efficiency of the VM models. Although, this is not a complete set of metrics to assess cost efficiency of VM models but these metrics along with other already proposed may act as objective indicators for the quality attribute. This set of metrics may not be comprehensive and other consecutive research could further complete this set by defining new metrics from other different perspectives. Replicated studies with more data need to be carried out to generalise the results. These metrics need to be empirically validated with the help of case studies or

When plotted, the closer is the data points, coming to form a straight line, the higher is the correlation between the two set of variables, or their exist a stronger relationship.

industrial data and by involving professionals to draw strong conclusion which can be applied in practice.

6. REFERENCES

- [1] Inmon, W.H., *Building the Data Warehouse*. John Wiley, 1992.
- [2] Bellahsene, Z.: Schema evolution in data warehouses. *Knowl. and Inf. Syst.* 4(2) (2002)
- [3] Kimball, R. *The Data Warehouse Toolkit*. John Wiley, 1996.
- [4] Blaschka, M., Sapia, C., Höfling, G.: On Schema Evolution in Multidimensional Databases. In: Mohania, M., Tjoa, A.M. (eds.) *DaWaK 1999*. LNCS, vol. 1676. Springer, Heidelberg (1999)
- [5] Fan, H., Poulouvasilis, 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. Springer, Heidelberg (2004)
- [6] Favre, C., Bentayeb, F., Boussaid, O.: Evolution of Data Warehouses’ Optimization: A Workload Perspective. In: Song, I.-Y., Eder, J., Nguyen, T.M. (eds.) *DaWaK 2007*. LNCS, vol. 4654. Springer, Heidelberg (2007)
- [7] Zuse, H. A.: ‘Framework of Software measurement’, Walter de Gruyter, Berlin, 1998.
- [8] Gupta, A., Mumick, I.S., Rao, J., Ross, K.: Adapting materialized views after redefinitions: Techniques and a performance study. *Information Systems* (26) (2001)
- [9] Gupta, A., I.S. Mumick, “Maintenance of Materialized Views: Problems, Techniques, and Applications.” *Data Eng. Bulletin*, Vol. 18, No. 2, June 1995.
- [10] Nica, A., Lee, A.J., Rundensteiner, E.A.: The CSV algorithm for view synchronization in evolvable large-scale information systems. In: Schek, H.-J., Saltor, F., Ramos, I.,
- [11] Alonso, G. (eds.) *EDBT 1998*. LNCS, vol. 1377. Springer, Heidelberg (1998)
- [12] Jian Yang, Kamalakar Karlapalem, Qing Li, “A Framework for Designing Materialized Views in Data Warehousing Environment “Proceedings of the 17th International Conference on Distributed Computing Systems (ICDCS ’97), IEEE 1997
- [13] Yousry Taha, Arsany S. Sawiros, Noha Adly, “an efficient data warehousing framework” computing -

- Technology and engineering, e-Publisher: CiteSeerX , 2009.
- [14] Miranda Chan, Hong Va Leong, Antonio Si, "Incremental Update to Aggregated Information for Data Warehouses over Internet" DOLAP 2000 ACM, ISBN 1-58113-323-5/00/0011
- [15] José A. Rodero, José A. Toval, Mario G. Piattini, "The audit of the Data Warehouse Framework" Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'99) Heidelberg, Germany, 14. - 15. 6. 1999
- [16] M. Golfarelli, S. Rizzi, "A Methodological Framework for Data Warehouse Design", Proceedings of First International Workshop on Data Warehousing and OLAP (DOLAP, in connection with CIKM'98), Washington, D.C., USA, November 1998.
- [17] Darja Solodovnikova and Laila Niedrite, "Evolution-Oriented User-Centric Data Warehouse", Proceedings of the 19th International Conference on Information Systems Development by Springer, 2010
- [18] C. Quix. "Repository Support for Data Warehouse Evolution". In Proc. of the Intl. Workshop DMDW, Heidelberg, Germany (1999)
- [19] Anisoara Nica, Elke A. Rundensteiner, "Using Containment Information for View Evolution in Dynamic Distributed Environments" DEXA '98 Proceedings of the 9th International Workshop on Database and Expert Systems Applications ,Page 212 , IEEE Computer Society Washington, DC, USA1998
- [20] Dragan Sahpaski, Goran Velinov, Boro Jakimovski, Margita Kon-Popovska, "Dynamic Evolution and Improvement of Data Warehouse Design" Fourth Balkan Conference in Informatics, IEEE,2009
- [21] Claudine Bréant, Gérald Thurler, François Borst, Antoine Geissbuhler, "Design of a Multi Dimensional Database for the Archimed DataWarehouse" Connecting Medical Informatics and Bio-Informatics R. Engelbrecht et al. (Eds.) ENMI, 2005
- [22] George Papastefanatos, Panos Vassiliadis, Alkis Simitsis, Yannis Vassiliou, "Design Metrics for Data Warehouse Evolution" ER 2008, LNCS 5231, pp. 440–454, 2008.
- [23] Matthias Jarke, Christoph Quix, Diego Calvanese, Maurizio Lenzerini, Enrico Franconi, Spyros Ligoudistianos, Panos Vassiliadis, Yannis Vassiliou, "Concept Based Design of Data Warehouses: The DWQ Demonstrators". SIGMOD Conference 2000: 591
- [24] David Botzer, Opher Etzion, "Optimization of Materialization Strategies for Derived Data Elements," IEEE Transactions on Knowledge and Data Engineering, vol. 8, no. 2, pp. 260-272, April, 1996.
- [25] Golfarelli, M., Lechtenböcker, J., Rizzi, S., Vossen, G.: Schema versioning in data warehouses: Enabling cross-version querying via schema augmentation. *Data Knowl. Eng.* 59(2), 435–459 (2006).
- [26] Gosain A., Sabharwal S., Gupta R., "Quality Metrics for View Maintenance Models of Data Warehouse", ERCICA 2014, Bangalore, India.
- [27] Gosain A., Sabharwal S., Gupta R., "An Efficient Feature Selection Approach for Materialized Views" IEEE, ICCCCM2013, Allahabad, India.
- [28] Zuse, H.: Properties of software measures, *Software Quality Journal*, 1992, 1, pp. 225- 260.
- [29] Darja Solodovnikova and Laila Niedrite, "Evolution-Oriented User-Centric Data Warehouse", Proceedings of the 19th International Conference on Information Systems Development by Springer, 2010
- [30] Dimitri Theodoratos, Mokrane Bouzeghoub, "A General Framework for the View Selection Problem for Data Warehouse Design and Evolution" DOLAP '00 Proceedings of the 3rd ACM international workshop on Data warehousing and OLAP, Pages 1 – 8, ACM New York, NY, USA ©2000
- [31] M. Blaschka. "FIESTA: A Framework for Schema Evolution in Multidimensional Information Systems". In 6th CAiSE Doctoral Consortium, Heidelberg, 1999.
- [32] C. Quix. "Repository Support for Data Warehouse Evolution". In Proc. of the Intl. Workshop DMDW, Heidelberg, Germany (1999)
- [33] Anisoara Nica, Elke A. Rundensteiner, "Using Containment Information for View Evolution in Dynamic Distributed Environments" DEXA '98 Proceedings of the 9th International Workshop on Database and Expert Systems Applications ,Page 212 , IEEE Computer Society Washington, DC, USA1998
- [34] Mahesh B. Chaudhari, Suzanne W. Dietrich, "A Distributed Event Stream Processing Framework for Materialized Views over Heterogeneous Data Sources", VLDB 2010, Singapore.
- [35] Ericka-Janet Rechy-Ramírez, Edgard Benítez-Guerrero, "A Model and Language for Bitemporal Schema Versioning in DataWarehouses" Proceedings of the 15th International Conference on Computing (CIC'06), IEEE2006.
- [36] S. Chen, X. Zhang, E.A. Rundensteiner. "A Compensation-based Approach for Materialized View Maintenance in Distributed Environments". In Computer Science Technical Report, Worcester Polytechnic Institute, Worcester, MA, USA (2004)
- [37] Chuan Zhang, Jian Yang, "Materialized View Evolution Support in DataWarehouse Environment" Sixth International Conference on Database Systems for Advanced Applications (DASFAA'99), 1999.
- [38] Dragan Sahpaski, Goran Velinov, Boro Jakimovski, Margita Kon-Popovska, "Dynamic Evolution and Improvement of Data Warehouse Design" Fourth Balkan Conference in Informatics, IEEE,2009
- [39] Amy J. Lee, Anisoara Nica, Elke A. Rundensteiner, "The EVE Approach: View Synchronization in Dynamic Distributed Environments", *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 5, september/october 2002
- [40] Robert M. Bruckner, Tok Wang Ling, Oscar Mangisengi, A Min Tjoa, "A Framework for a Multidimensional OLAP Model using Topic Maps" IEEE 2002.
- [41] Xin Zhang, Elke A. Rundensteiner, "The SDCC Framework For Integrating Existing Algorithms for

- Diverse Data Warehouse Maintenance Tasks” Database Engineering and Applications, 1999. IDEAS '99. International Symposium Proceedings , Aug 1999 Page 206 - 214
- [42] PAN Ding, PAN Yunshan,” Metadata Versioning for DW2.0 Architecture” Proceedings of the 29th Chinese Control Conference July 29-31, 2010, Beijing, China
- [43] Cécile Favre, Fadila Bentayeb, and Omar Boussaid, “Evolution of Data Warehouses’ Optimization: A Workload Perspective” DaWaK 2007, LNCS 4654, pp. 13–22, 2007.
- [44] Bartosz Bebel, Zbyszko Królikowski, and Robert Wrembel, “Managing Evolution of Data Warehouses by Means of Nested Transactions”, ADVIS 2006, LNCS 4243, pp. 119–128, 2006
- [45] J. A. Nasir, M. Khurram Shahzad, “Architecture for Virtualization in Data Warehouse” Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 243–248. 2007 Springer.
- [46] M.K. Shahzad, J.A. Nasir, M.A. Pasha. “CEV-DW: Creation and Evolution of Versions in Data Warehouse”. In Asian Journal of Information Technology, 4(10) (2005) 910-917
- [47] E. Benítez-Guerrero, C. Collet, and M. Adiba. “The WHES Approach to Data Warehouse Evolution”. Digital Journal e-Gnosis [online], <http://www.e-gnosis.udg.mx>, ISSN No. 1665-5745, 2003.
- [48] Joseph M. Firestone,” Architectural Evolution in DataWarehousing and Distributed Knowledge Management Architecture” White Paper No. Eleven July 1, 1998.
- [49] George Papastefanatos, Panos Vassiliadis, Alkis Simitsis, Yannis Vassiliou,” Design Metrics for Data Warehouse Evolution” ER 2008, LNCS 5231, pp. 440–454, 2008.
- [50] George Papastefanatos, Panos Vassiliadis, Alkis Simitsis, Konstantinos Aggystalis, Fotini Pechlivani, Yannis Vassiliou, “language extensions for the automation of database schema evolution” iceis (1) 2008: 74-81.
- [51] B. Ashadevi, Dr. P. Navaneetham,” A Framework for the View Selection Problem in Data Warehousing Environment” International Journal on Computer Science and Engineering Vol. 02, No. 09, 2010, 2820-2826
- [52] B. BEBEL, Z. KRÓLIKOWSKI, R. WREMBEL,” Formal approach to modelling a multiversion data warehouse” bulletin of the polish academy of sciences technical sciences vol. 54, no. 1, 2006
- [53] Garima Thakur, Anjana Gosain,” DWEVOLVE: A Requirement Based Framework for Data Warehouse Evolution” ACM SIGSOFT Software Engineering Notes, Page 1, November 2011 Volume 36, No.6.
- [54] Resmi Nair, Campbell Wilson, Bala Srinivasan,” A Conceptual Query-Driven Design Framework for Data Warehouse” , World Academy of Science, Engineering and Technology 25 , 2007.
- [55] Solodovņikova D. “The Formal Model for Multiversion Data Warehouse Evolution”, Postconference proceedings of the 8th International Baltic Conference on Databases and Information Systems, Frontiers in Artificial Intelligence and Applications, IOS Press, 2008.
- [56] D. Agrawal, A. El Abbadi, A. Singh, T. Yurek., “Efficient View Maintenance in Data Warehouses”. In Proceedings of the 1997 ACM International Conference on Management of Data, pages 417-427, May 1997.
- [57] Briand, L.C., Morasca, S., Basili, V.R.: ‘Property based software engineering measurement’, IEEE Trans. Softw. Eng., 1996, 22, pp. 68–86.
- [58] H. Gupta, ‘Selection of views to materialize in a data warehouse’, ICDT’97, Springer 1997.
- [59] George Papastefanatos, Panos Vassiliadis, Alkis Simitsis, Yannis Vassiliou, ‘Metrics for the Prediction of Evolution Impact in ETL Ecosystems: A Case Study’ Springer-Verlag 2012.