COMBINING THE DATA WAREHOUSE AND OPERATIONAL DATA STORE

Ahmed Sharaf Eldin Ahmed, Yasser Ali Alhabibi
Faculty of Computer and Information Sciences, Helwan University, Cairo, Egypt

Abdel Badeeh M. Salem
Faculty of Computer and Information Sciences, Ain Shams University, Cairo, Egypt

Keywords: Data Warehousing (DW), Operational Data Stores (ODS).

Abstract: Many of small business organizations tend to combine the operational data stores (ODS) and data warehousing (DW) in one structure in order to save the expenses of building two separate structures for each of them. The purpose of this paper is investigating the expected obstacles that may affect organizations that try to combine the ODS and DW in one structure. Both the analytical and comparative analysis are used to investigate the obstacles and drawbacks that have been faced in combining the ODS and DW in one structure.

1 INTRODUCTION

The classical structure for delivering business intelligence contains two levels of data that are of interest the data warehouse data (DW) and the data mart data (Mattison, 1996). The data warehouse, often called the current level of detail, contains the bulk of the detailed data that has been collected and integrated from the applications environment. The data mart is a departmental subset of the current detailed data that is shaped to meet the decision support system (DSS) processing needs of that particular department. For all the benefits of a data warehouse and its associated data marts, there is still a need for collective, integrated operational, DSS/informational processing. When this need arises, an operational data store (ODS) is in order. An ODS is a hybrid structure that has equally strong elements of operational processing and DSS processing. This dual nature of the ODS easily makes it the most complex architecture structure in the corporate information factory (Harry, 1999).

Trying to combine the data warehouse (DW) (Singh, H., 1999) and the operational data store (ODS) (Dhor Vasant and Stein Roger, 1996) into the same structure gets appreciation from many of small business organizations (Inmon et al., 1998) (specially financial managers and owners) for saving expenses point of view, although it may be dispute with the specifications of the corporate information factory (CIF) (Farzad Shafiei, David Sundaram, 2004). It is theoretically possible to build such a structure, and under very limited circumstances, such a combination structure can be made to work (Bolloju, 2001). Where there is a very small amount of data and processing, and an abundance of processing power, it is possible to merge an ODS and DW into the same structure (Delic et al., 2001). The capacity levels for the processor under normal hours of utilization in order for the merger to function properly. Unfortunately, it is not economically feasible to purchase a powerful and versatile machine and use it so sparsely. The remainder of this paper is organized as follows. The next section presents an overview about MTI multidimensional database, DW, and ODS, the third section demonstrates the experiments that are applied to investigate the expected obstacles when combining ODS and DW as incompatible transaction types, in the first part of this section, the principle and workload extraction are introduced, followed by discussing the experiments' results. The last section summarizes the conclusion and future work.

Sharaf Eldin Ahmed A., Ali Alhabibi Y. and Badeeh M. Salem A. (2006).
COMBINING THE DATA WAREHOUSE AND OPERATIONAL DATA STORE.
In *Proceedings of the Eighth International Conference on Enterprise Information Systems - DISI*, pages 282-288
DOI: 10.5220/0002490802820288

2 MTI DATABASE, DW, AND ODS

MTI company is one of Seoudi Group (Seoudi Group, 1998) (SG) companies that is dealing in vehicle business (marketing, sales, distribution, customer support, and after sales services). MTI drives its business for all vehicle brands in Seoudi Group (NISSAN, FIAT, SUZUKI, DATCIA, and BLAC) and in all categories of sales types (retail, fleet, government, and wholesales). MTI has two outlets for sales, client/server distributed system (7 client machines Pentium III-500 running under windows 98, 2000, and XP platforms with 10 GB HDD capacity, and 1 server machine Pentium III-500 running under windows NT platform with 20 GB HDD capacity), Ethernet LAN network. MTI has strong built in multi databases system (about 80 MB) for marketing, sales, and customer support (relational multidimensional database system running under MS access software and windows XP platform). MTI has detailed historical data for the period from Jan-1999 till Apr-2004. Metadata (Nestorov and Jukic, 2003) and data marts technologies are available for all varieties of departmental and activities of business that covers all analytical analysis for marketing, sales, and customer support purposes. Figure 1 shows MTI multidimensional databases model.

3 EXPERIMENTS

The following four experiments are run to investigate the expected obstacles when combining ODS and DW (Nestorov and Jukic, 2003) as incompatible transaction types:

a. Combination of incompatible transaction types (ODS and DW) in one machine (server) such as current month sales figures and historical sales figures for the previous month and the same month last year as shown in figures 2 and 3.

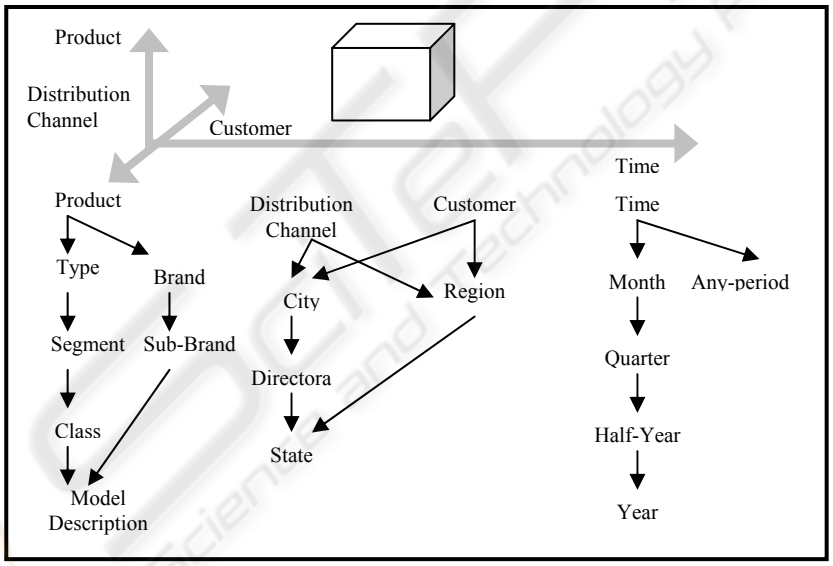

Figure 1: MTI Multidimensional Databases Model.

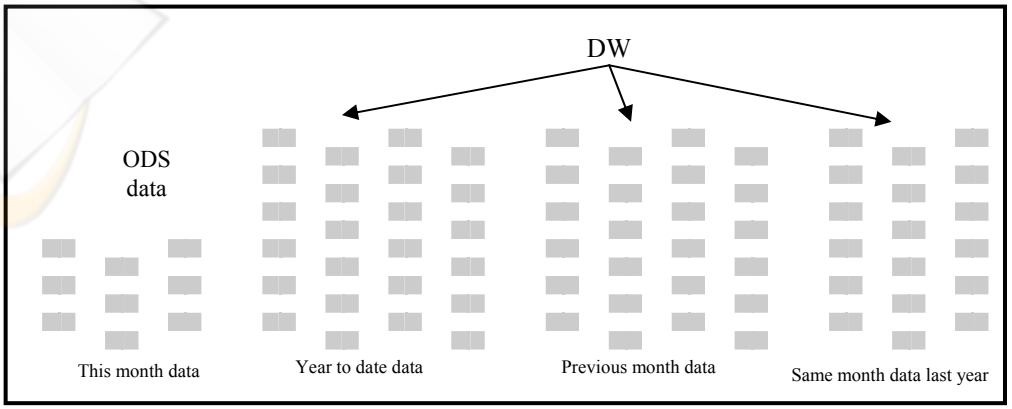

Figure 2: Current data is mixed with historical data when the ODS is mixed with the DW.

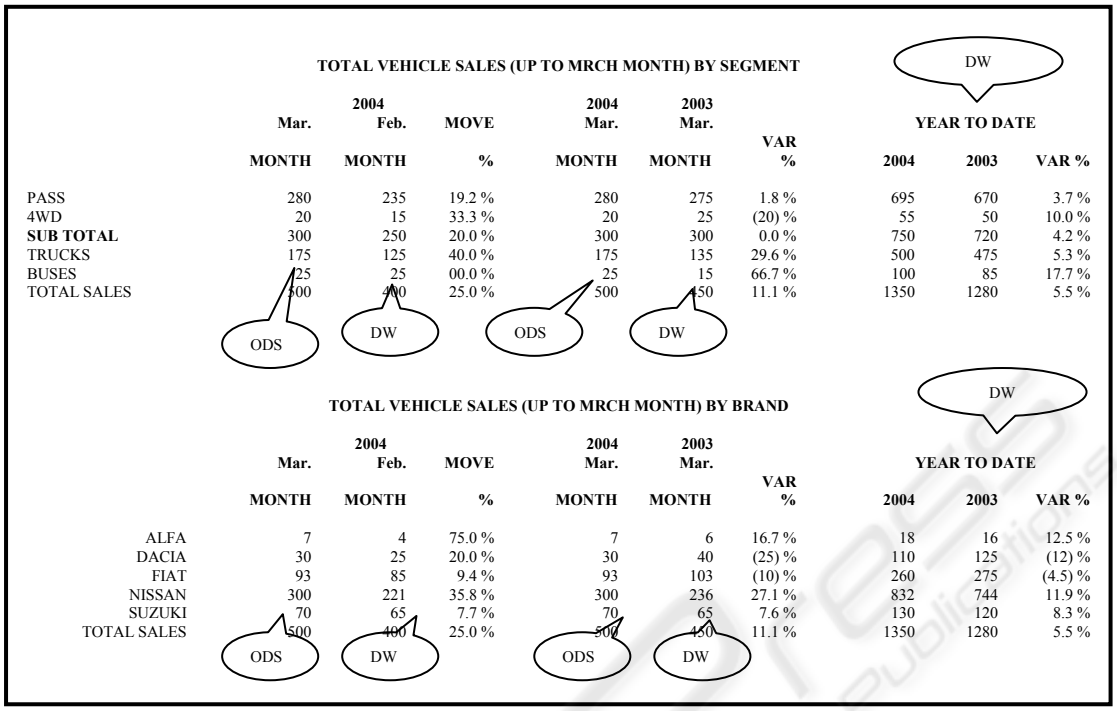

Figure 3: Samples of Integrating Combined/Separated ODS and DW Information for DSS Purposes.

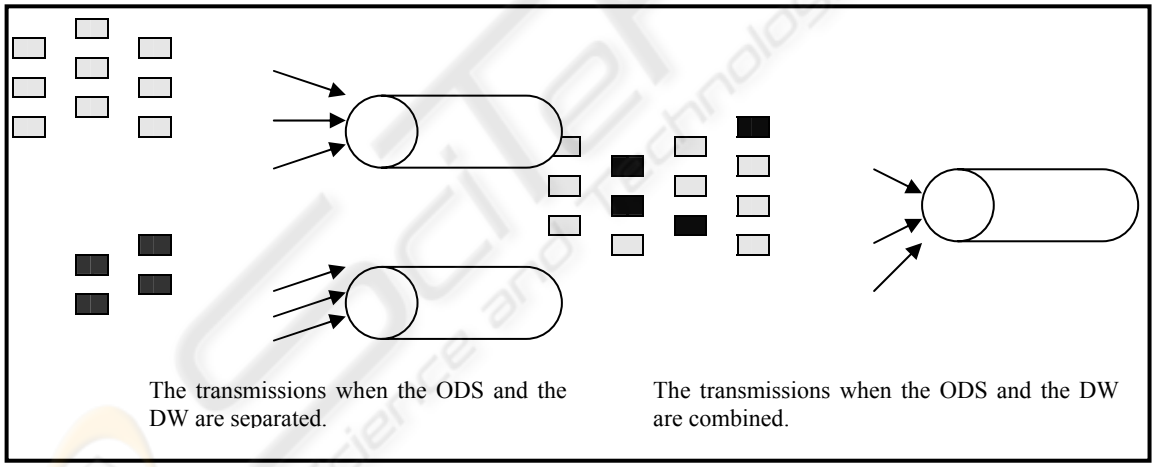

Figure 4: Mixing Combined Workloads across the Communication Lines.

b. Separation of incompatible transaction types (ODS and DW) in two machines (ODS transactions in server machine such as On Line Transaction Processing – OLTP for the current month sales figures and DW transaction such as analytical transactions for historical sales figures for the previous month and the same month last year in administrator client machine) as shown in figures 3 and 4.

c. Forced combination of incompatible types (ODS and DW), firstly in large processor-mini computer-S400 machine that belong to a sister company in SG (Seoudi Group, 1998) called MM company, and secondly, in small processor-server machine Pentium III-500 as shown in figure 3.

d. Using a small machine-server to house a mixed workload of ODS and DW as shown in figure 3.

3.1 Workload Extraction

The queries from the transaction log constitute a workload that is treated by an SQL query analyzer. The SQL query analyzer extracts all the attributes that may be indexed (indexable attributes). It is assumed that a workload similar to the one presented in figure 5. Such a workload can be easily

```
Q1: SELECT * FROM T1, T2 WHERE A BETWEEN 1 AND 10AND C=D
Q2: SELECT * FROM T1, T2 WHERE B LIKE THIS%' AND C=5 AND E<100
Q3: SELECT * FROM T1, T2 WHERE A=30 AND B>3 GROUP BY C HAVING SUM(E) >2
Q4: SELECT * FROM T1 WHERE B>2 AND E IN (3, 2, 5)
Q5: SELECT * FROM T1, T2 WHERE A=30 AND B>3 GROUP BY C HAVING SUM (E)>2
Q6: SELECT * FROM T1, T2 WHERE B>3 GROUP BY C HAVING SUM (E)>2
```

Figure 5: Sample Workload.

```
// Cold run (no timing)
FOR each query in the workload DO
        Execute current query
END FOR
// Warm run
FOR i = 1 To number of replications DO
        FOR each query in the workload DO
                Execute current query
                Compute response time for current query
        END FOR
END FOR
Compute global mean response time and confidence interval
```

Figure 6: Test Protocol Algorithm.

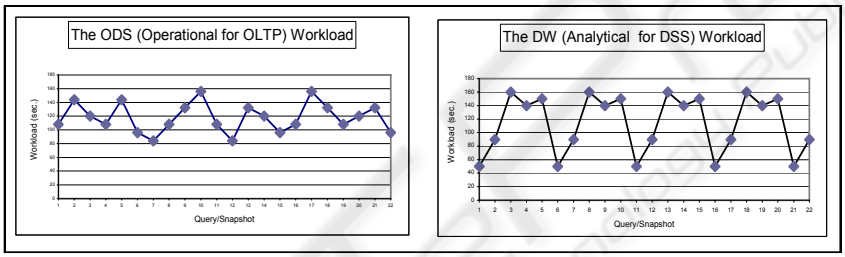

Figure 7: Combining ODS and DW into the same structure.

obtained from the DBMS transaction log. To reduce response time when running a database query, it is better to build indexes on attributes that are used to process query. These attributes belong to the WHERE, ORDER BY, GROUP BY, and HAVING clauses of SQL queries. In order to validate this approach, it has been applied on a test ODS and a test DW to measure the variance in response time in the following four experimental situations: combination of ODS and DW in one machine (server); separation of ODS and DW in two machines; forced combination of ODS and DW (in smaller and larger processors as explained above); and finally, using a small machine-server to house a mixed workload of ODS and DW. The Test Protocol Close-Result (TPC-R) decision-support benchmark (Poess et al., 2002) (Transaction Processing Council, 1999) is applied for these experiments on a relational database. The TPC-R 1 GB database has been generated and used the benchmark's 22 queries (labeled Q1 to Q22). Taking into account that there is no standard benchmark for DWs yet (TPC-DS is still in development (Poess et al., 2002)). Hence, the work is done on a small data mart. This data mart is composed of an AbcVehInfo fact table and four dimension tables: product, place, customer, and date. It occupies 4 MB on disk. Starting from previous analysis on this datamart, a realistic DSS workload is also designed that is specifically adapted to it. This workload includes both selection and update operations. It cannot be presented in detail here due to limitation of space. Both the TPC-R database and the AbcVehInfo data mart have been implemented within the Access DBMS. The test protocol that was adopted is presented in figure 6.

This algorithm has been executed for various values of the minsup (minimal support) parameters as follows: the number of attributes varies from 1 to 7 and the number of indexes varies from 0 to 3. In practice this parameter helps in limiting the number of indexes to be generated by selecting only those that are the most frequently used by the workload. At each step corresponding to the value of minsup, the mean response time of the input workload is computed.

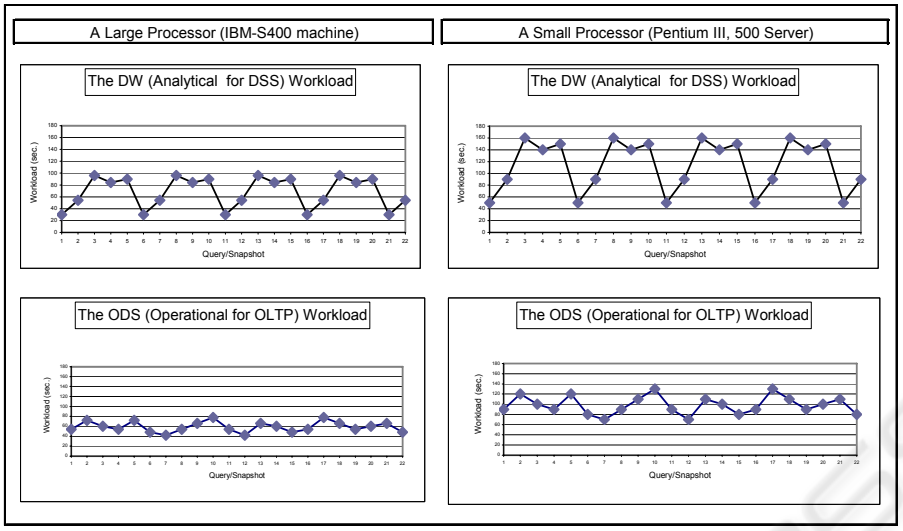

Figure 8: Large Processor Vs. a Small Processor.

Table 1: Tentative Expenses Cost Comparison between Large and Small Processor Machines.

	Large processor such as IBM S400 or equivalent	Small processor such as PC Pentium III 500 Server or equivalent
Hardware estimated price in US $	Minimum 40,000	Minimum 1,500
Software price in US #	Minimum 15,000	Minimum 1,000
Users	Higher experienced and qualified users	Normal experienced users
Technical support	Essentially needed and expensive (100 US $ per visit)	Rarely needed and cheap (10 US $ per visit)

3.2 Experiments' Results

The results obtained are presented in figures 7 and 8. The results from figure 7 correspond to combining ODS and DW (Bhargava et al., 1999) into the same structure.

Figure 7 shows that there are very different workload patterns for the ODS environment and the DSS environment. The ODS workload is one where there are peaks and valleys, but where the mean system utilization is a descriptive number. Peaks and valleys depend on the needed response time for computing the resultant answer for the query or snapshot process. The ODS workload is one that can be predicted and managed, and the DW workload is essentially a binary workload. Either the system is being used heavily or not being used at all. This workload is not predictable and the mean utilization of the system is a useless number in most circumstances. When the ODS and DW are mixed in the same environment and technology, the two workloads are forced into the same box. Figure 8 shows what happens when the ODS and the DW (Awad, 2000) are mixed in the same environment and technology, the two workloads are forced into the same box. As shown in left side diagram in figure 8 that when combining the two workloads in a large processor (IBM S400 machine) the end user will be happy with the response time but the price is as follows: *the unit price of the hardware is as expensive as it gets; the organization has excess capacity left over; and finally, the financial manager considers the purchase to be a disaster because it doubles the expenses minimally several ten times of cost compared with the other solution (see table 1).*

On the other hand, the right side diagram in figure 8 shows that when combining the two workloads in a small processor (PC Pentium III, 500 Server) the end user is suffering a lot of a long response time. In fact the use of this alternative approach may appreciate the finance people (especially when they find out that the machine is being used close to 100 percent of the time). However, the following problems have been arisen when a small machine is used to contain a mixed workload:

a. Response Time

The end user doing ODS simply is unable to cope with the unpredictable and unacceptable levels of response that are achieved (response time is increased by triple in some cases of analytical analysis and more in several other cases).

b. Style Incompatibility

The system cannot be tuned or optimized for any style of processing. This workload incompatibility is solved by separating the ODS and the DW (Little and Gibson, 2003).

c. Mixing of Communities

When the ODS and the DW are mixed, the DSS (Vahidov, 2002) analytical community is thrown in with the clerical community. Unfortunately, when enough people start to use the combined structure, they start to "step on each other's together".

d. Data Flow are Incompatible

When there is a combined workload, data flow across the communication lines are mixed as well. In order to achieve a uniform and efficient flow, the separation of the different data flow types is must needed.

e. Mixed Current and Historical Data

When current data is mixed with historical data (Jennex et al., 2003), the problem of the difference in the probability of data access arises. Current data typically has a much higher probability of access than historical data. But when they are mixed together, current data can hide behind historical data, making current data hard and inefficient to access.

f. Overhead of Update

The overhead of update shows up as check pointing, rollback, logging of transactions, and committing data. When update is a possibility all transactions that are in execution pay the same price of overhead. When the ODS is separated from the DW (Harada et al., 2004), separating update processing from access processing is urgently needed. Update processing is regularly done in the ODS environment while access-only processing (which is very efficient) is done in the DW environment.

g. No Optimal Hardware Architecture

When the ODS and the DW are mixed, there is no optimal hardware architecture. When the ODS is separated from the DW, the ODS environment has a particularly strong affinity for Multiple Processor Platform (MPP) architecture: multiple units of data are tied together by a common backbone, in other words, ODS environment operates optimally on MPP architecture. Depending on the size and the processing done, the DW may or may not run optimally on MPP architecture. Indeed, the DW may operate optimally on Shared Memory Platform (SMP) architecture: multiple processors are tied together in a shared memory configuration. When this is the case, the mixing of the two environments causes a dilemma for the CIF (Brendt et al., 2001) architecture. These are the reasons why the ODS and the DW need to be split. They can violate the CIF (Schwarz et al., 2001) and be combined, but if this is done, the CIF architecture needs to be aware that there are prices to be paid.

4 CONCLUSION AND FUTURE WORK

Attempts to combine the ODS with the DW by some small business organizations aiming to save expenses is limited and poor idea. As the experimental results proved that the obstacles and the drawbacks of combining the ODS and DW destroy the aims of saving expenses but makes the organization to pay the price a lot. The drawback and obstacles of combining the ODS and the DW are severe and might arise in different ways. In case of using forced combination of incompatible transaction types, the experimental results showed that the following problems have been arisen: *system blocks cannot be optimized for one type of transaction or the other; buffers cannot be optimized for one type of processing or the other; system initialization parameters, such as FREESPACE cannot be optimized for one type of activity or the other; and finally, data cannot be distributed across system resources (CPU, disk, etc.) for one type of processing or the other*. In case using a small machine such as PC Pentium III 500 to house a mixed workload, the experimental and analytical results proved that the following problems have been arisen: *much increase in response time, the system cannot be tuned or optimized for any style of processing (style incompatibility), mixing of communities, data flow are incompatible, mixed current and historical data, overhead of update, and finally, no optimal hardware architecture*. In case of using a large processor such as IBM S400 machine the response time is much improved compared with using small machine (PC Pentium III 500) but the price is very severe and the other problems that are mentioned above are still as they are. Finally, the experimental and analytical results conclude that the ODS and the DW need to be physically separate entities and environments in order to ensure long-term viability of the corporation information ecosystem. Future work is developing different hardware platforms and hardware architectures particularly the unit-processor architectures (a single storage device is controlled by a single processor) to work separately better in the ODS, DW, and data marts environments.

REFERENCES

Awad H. K, 2000, A Multi-Entry Methodology for Data Warehouse Develop. Proc. of *8th Cairo Int. Conf on AI Applications,* Cairo, Egypt, pp. 21-22.

Dhor Vasant and Stein Roger, 1996, *Seven Methods for Transforming Corporate Data into Business Intelligence,* Prentice Hall PTR, London, UK

Donald J. Brendt, John W. Fisber, Alan R. Hevner, 2001, *Healthcare Data Warehousing and Quality Assurance,* Computer, December 2001, Vol. 34, No. 12, pp. 56-65, IEEE Computer Society, Piscataway, NJ, USA

Farzad Shafiei, David Sundaram, 2004, *Multi-Enterprise Collaborative Enterprise Resource Planning and Decision, Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 8, pp. 80228a, January 05 - 08, 2004,* Big Island, Hawaii.

Harry Singh, 1999, *Interactive Data Warehousing,* Prentice Hall, Inc, A Simon & Schuster Company, Upper Saddle River, NJ 07458, USA

Hemant K. Bhargava, Suresh Sridhar, and Craig Herrick, 1999, Beyond Spreadsheets: Tools for Building Decision Support Systems, *Computing Practices, March 1999, pp. 31-39, 0018-9162/99 @ 1999 IEEE.*

Holger Schwarz, Ralf Wagner, Brenhard Mitschang, 2001, Improving the Processing of Decision Support Queries: The Case for a DSS Optimizer, 2001, *International Database Engineering & Applications Symposium (IDEAS-01), Jul-01, pp. 0177.*

Inmon W. H., Imhoff Claudia and Sousa Ryan, 1998, *Corporate Information Factory*, Wiley Computer Publishing, New York, USA

Kemal A. Delic, Laurent Douillet, Umeshwar Dayal, 2001, Towards Architecture for Real-Time Decision Support Systems: Challenges and Solutions, *2001 International Database Engineering & Applications Symposium (IDEAS-01), Jul-01, pp. 0303.*

Lilian Harada*,* Yuuji Hotta and Tadashi Ohmori, 2004, Detection of Sequential Patterns of Events for Supporting Business Intelligence Solutions, *International Database Engineering and Application Symposium (IDEAS'04), July, 07-09, 2004, pp. 475-479, Coimbra, Portugal , IEEE, Piscataway, NJ, USA.*

M. Poess, B. Smith, L. Collar, and P. A. Larson, 2002, TPC-DS: Taking Decision Support Benchmarking to the next level. In *2002 ACM SIGMOD International Conference on Management of Data, Madison, USA*

Michael Blaha, 2001, Data Warehouses and Decision Support Systems, *Computer, December 2001, Vol. 34, No. 12, pp. 38-39,* IEEE Computer Society, Piscataway, NJ, USA

Murray E. Jennex, Lorne Olfman and Theophilus B. A. Addo, 2003, the Need for an Organizational Knowledge Management Strategy, *36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 4, pp. 117a, January, 06 - 09, 2003, Big Island, Hawaii.*

N. Bolloju, 2001, Extended Role of Knowledge Discovery Techniques in Enterprise Decision Support Environments, *34th Annual Hawaii International Conference on System Sciences (HICSS-34) - Volume 3, Jan-2001, pp. 3012.*

R. Vahidov, 2002, Decision Station: A Notion for a Situated DSS, *35th Annual Hawaii International Conference on System Sciences (HICSS' 02), Vol. 3, pp. 78b, January, 07-10, 2002,* Big Island, Hawaii.

Rob Mattison, 1996, *Data Warehousing*, Irwin McGraw-Hill, New York, USA

Robert Grover Little*,* Michael Lucas Gibson, 2003, Perceived Influences on Implementing Data Warehousing, *IEEE Transactions on Software Engineering, April 2003 (Vol. 29, No. 4), pp. 290-296,* IEEE, Piscataway, NJ, USA.

Seoudi Group, 1998, *SG Manual Reference*, SG Documents, by permission from GM.

Singh Harry, 1999, *Data Warehousing: Concepts,* Technologies, and Implementation Prentice Hall PTR, London, UK.

Svetlozar Nestorov and Nenad Jukic, 2003, Ad-Hoc Association-Rule Mining within the Data Warehouse, *36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 8, pp. 232a, January 06 - 09, 2003,* Big Island, Hawaii.

T. Hess, J. Wells, 2002, Understanding How Metadata and Explanations Can Better Support Data Warehousing and Related Decision Support Systems: Exploratory Case Study, *35th Annual Hawaii International Conference on System Sciences (HICSS-35) - Volume 8, Jan-2002, pp. 223.*

Transaction Processing Council, 1999, *TCP Benchmark R standard Specification.*