International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:6, No:11, 2012

The Comparison of Anchor and Star Schema from a Query Performance Perspective

Radek Němec

***Abstract*—**Today's business environment requires that companies have access to highly relevant information in a matter of seconds. Modern Business Intelligence tools rely on data structured mostly in traditional dimensional database schemas, typically represented by star schemas. Dimensional modeling is already recognized as a leading industry standard in the field of data warehousing although several drawbacks and pitfalls were reported. This paper focuses on the analysis of another data warehouse modeling technique - the anchor modeling, and its characteristics in context with the standardized dimensional modeling technique from a query performance perspective. The results of the analysis show information about performance of queries executed on database schemas structured according to principles of each database modeling technique.

***Keywords*—**Data warehousing, anchor modeling, star schema, anchor schema, query performance.

I. Introduction

THROUGH the years of evolution of data warehousing many data modeling techniques were discussed and used for building a data warehouse. First steps in the data warehousing evolution were made by Ralph Kimball who was a pioneer in this field, with his co called bottom-up approach of building individual data marts, at first holding non-integrated data. His work was subsequently largely criticized by another successful propagator of his own data warehousing approach - William H. Inmon. The critique relied on the fact that Kimball's data marts architecture was first based on non-integrated data intended for support of decision making needs of specific departments, divisions, etc. while the data was not integrated in an enterprise-wide fashion. Inmon's Corporate Information Factory (top-down) approach was meant to be the solution of this problem while bringing other drawbacks on the other side, especially in the field of duration of the data warehouse development lifecycle (with derived and subsequent problems). Despite this fact Inmon's approach became popular among many companies.

Meanwhile R. Kimball was working on his data mart approach where he put stress especially on the development lifecycle's duration and easiness of literally piling data marts on the enterprise IT infrastructure stack to support various decision support needs. This approach however emerged as a not very efficient way of development since decision support needs started to increase in the 90's and along with the growing number of data marts and their users the architecture became very messy by the time. R. Kimball decided to revise his concept and come up with the Bus architecture of data marts with integrated metadata and shared database objects (dimensions).

Kimball's concept was based on the dimensional modeling techniques where central fact tables contained facts (data) about transactions, account balances, total products sold etc. while dimensional tables contained detailed data about customers, products and mainly time of related to the facts etc. This approach became also very popular and so today we can encounter mixed (hybrid) architectures where central data warehouse is established (Inmon's concept of data warehouse shaped as a database in 3rd normal form) with historical data inside and linked with various data marts populated with subject oriented data as a subset of the central data warehouse.

In recent years numerous approaches of agile software development came into play (Scrum, Extreme Programming, agile modeling, lean software development, etc.). These approaches naturally became quite popular since they were viewed as a cure to common software development problems. Of course this discussion incorporated both data warehousing construction paradigms and their fathers had to come up with ideas how to alter their approaches so they comply with the agile trend. In the field of OLTP system's database construction there were also specific agile initiatives that were focused on bringing the agile aspects also to the field of transactional data modeling, especially[1] and also[2]that present an approach called the anchor modeling technique which is meant to be a truly agile data modeling technique intended to be used especially in the data warehousing environment according to its authors (the paper focuses on the application of this concept).

Many data warehouses suffer from having a model that does not fulfill requirements of being modular and flexibly alterable and that one third of implemented warehouses have at some point, usually within the first four years, changed their architecture and less than a third quote the warehouses as being a success[3]. This fact supports the importance of modeling the data warehouse in an agile way and agreeing with [7] businesses today have to operate in a world which has become increasingly dynamic so the reliance on ICT is therefore inevitable and dynamicity of data models that back-up decision-support software is a vital asset.

The aim of the paper is to present analysis of anchor and star data warehouse database schemas and their performance in terms of running queries with equal output. The paper servers as a starting point in research activities of our research grant SP2012/184 that deals with the matter of using the

Radek Němec is with the Faculty of Economic, Technical University of Ostrava, Sokolskátřída 33, 701 21 Ostrava 1, Czech Republic (phone: +420 596 992 475; e-mail: radek.nemec@vsb.cz).

anchor modeling in data warehousing environment. First section presents methodology of the paper while the second deals with the introduction of tested database schemas. In the last section there are located results of the query performance testing and also conclusion of the paper.

II. Methodology and Data Sample

3 separate relational database schemas were created and populated with data that are meant to simulate a data mart's database environment. 1 schema is created as a typical star schema (which is typically denormalized) and other 2 schemas are created as anchor schemas, first anchor schema is less decomposed (the schema is almost equal to the default star schema) and the second one is decomposed into more relations and the structure is closer to a snowflake schema, which is a more normalized than the star schema ([5], p.740).

Testing hardware and software configuration:

- *Database server hardware:* CPU 2x Intel XEON E5450 3GHz, 16 GB RAM
- *Database system:* Microsoft SQL Server 2008
- *Test data generator:* RedGate SQL Data Generator 2.0
- *Query performance testing software:* Apache JMeter 2.7

A. The Test Case

Sample data was created according to standard rules of relational database modeling and domain characteristics of the generated test data were based on real values.

Due to a different structure of relations in both anchor schemas, data volumes of the initial star schema had to be adapted for each anchor schema. The emphasis was placed on a condition that the information contained in both anchor schemas had to match the information contained in the star schema.

B. The Query Performance Test Suite

A set of 10 SQL queries was created for testing purposes. Queries were assigned to 2groups according to operators and SELECT statement constructions used in each query. Each query was run 200times and resulting query execution duration was recorded for each query and database schema. Execution time is meant as a time between execution of the query in the database engine and return of the output. Table I shows characteristics of both query groups.

TABLE I
Description of Queries in Query Groups

Group	Characteristics of queries in the group
QG_1	multiple tables, only inner joins, where conditions, grouping and summaries – without subqueries
QG_2	multiple tables, inner joins, where conditions, grouping and summaries – with subqueries in selections and where conditions

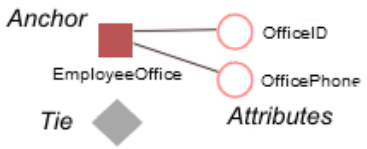

Fig. 1 Visual representation of anchor model's components

C. The Anchor Modeling Technique

The anchor modeling is database modeling technique that facilitates agile development of a database schema, especially focused on data warehouse development [2]. Anchor modeling is based on a finite set of easily understandable principles and concepts while is supposed to bring several benefits, like iterative and incremental development, ease of temporal querying, absence of null values, reusability, and efficient storage and higher performance which should be facilitated by table/join elimination functionality of a database systems' query optimizer (performance was tested by authors in a situation of an anchor model representing a similar ER model). The resulting anchor database schema is then highly decomposed and displays high degree of normalization almost fully satisfying normalization into 6th normal form. Relation in the 6th normal form is also in 5th normal form and thus decomposes the relation in the database schema into irreducible components while adding temporal validity of attributes' values [4].

D. Components of the Anchor Database Schema

In our sample models we used following components of an anchor model.

Anchor – represents set of entities (products, customers, employees). Logical representation: relational table A(K#) with 1 column K, where K is a primary key for A(K#).

Attribute – represents a property of an anchor. Logical representation: a relational table Att(K*, P) with 2 or 3 columns, where K* is primary key of Att and a non-null foreign key to respective anchor A(K#) and domain of P is a non-null data type. Attributes can be historized, static, knotted static and knotted historized. Historization means addition of a temporal validity attribute – table is then extended to Att(K*, P, T), where the domain of T is non-null time type.

Tie – represents association between two and more entities (anchors) – an implicit constructor for a many-to-many relationship. Logical representation: T(K^*_1,….K^*_n), where n = number of associated anchors, and each K_i for i={1,…n} is a foreign key to respective i-th anchor. Primary key of T is a subset of K_i for i={1,…n} according to which anchors are mandatory to be a part of the primary key of T (and thus uniquely identifying each tuple of T relation).

Knot components and related knotted attributes and ties were not used in our example models due to a need to maintain certain degree of simplicity and therefore will not be explained. For more thorough description of all anchor models' components and principles, please refer to[2]. Fig. 1 shows visual representation of all used components.

III. Sample Database Schemas

A. The STAR Schema

Our starting sample database model is a typical simple dimensional model represented by the STAR schema with 1 fact table and 4 related dimensional tables. The fact table contains data about total sales and total units sold by customers, products, employees and time with respective

granularity (level of detail) of dimensional tables. Both metrics in the fact table are fully additive which means that the summarization by each dimension is meaningful. Fig. 2 shows the structure of the sample data mart's star schema.

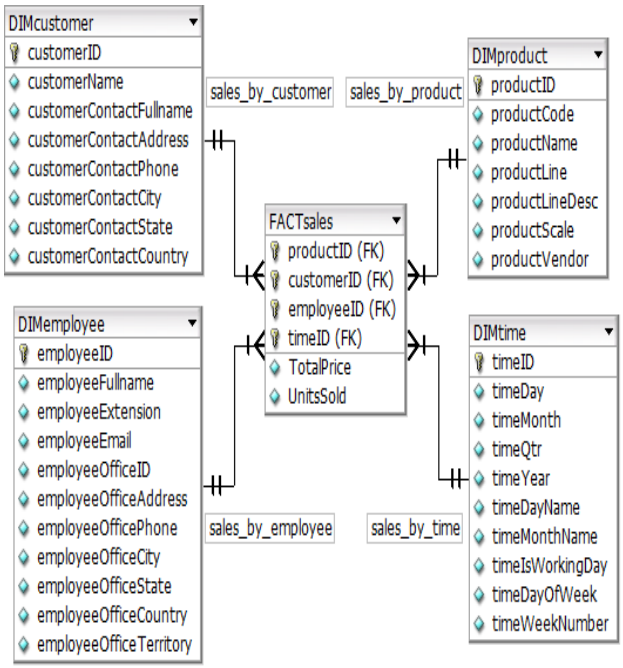

Fig. 2 Initial star schema of the sample data mart

B. The Less Decomposed Anchor Schema (AM_LESS)

Principally, the anchor model can be constructed with different strategies, so 2 anchor models were created. Both anchor schemas are related to the original star schema. First anchor schema is less decomposed (fig. 3), which means that itliterally imitates the original star schema. Attributes were decomposed into respective attribute tables, each related to its anchor table. All anchors are bound in a tie relationship like in a fact table with one difference – metrics contained in the original fact table had to be decomposed into a single anchor table with related attribute tables containing each metric. This fact however may present possibility of adding another metrics to the model more easily and without the need of recreating or altering the structure of a fact table as in STAR schema. All anchors are mandatory to participate in the central tie's primary key.

C. The More Decomposed Anchor Schema (AM_MORE)

The second more decomposed anchor schema (see fig. 4) looks more like the snowflake schema. In the creation process transitive dependencies in the default STAR schema were taken into account as well as possible reusability of selected attributes' values.

The anchor table *ProductLine* is intended to facilitate reusability of product lines' names and descriptions in a traditional ER modeling way as well as the anchor table *EmployeeOffice* was a natural selection for decomposition since it is expected that more than 1 employee could work at each office. Further decomposition of addresses into the *Address*anchor table is meant to facilitate better management of respective address values and also to keep *DIMcustomer* and *DIMemployee* anchor tables in a less complex form (and also possible reuse of addresses, if by chance any of them is common to office and also customer).

Despite these specific changes the schema models the same

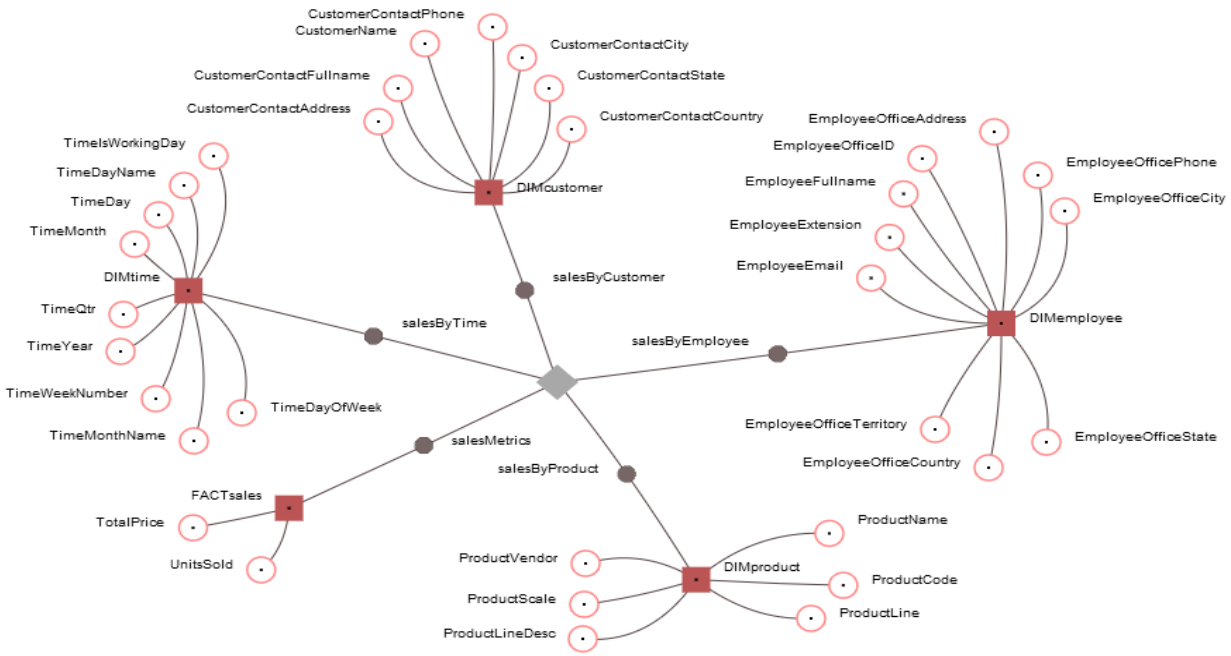

Fig. 3 Initial STAR schema converted into a less decomposed anchor model

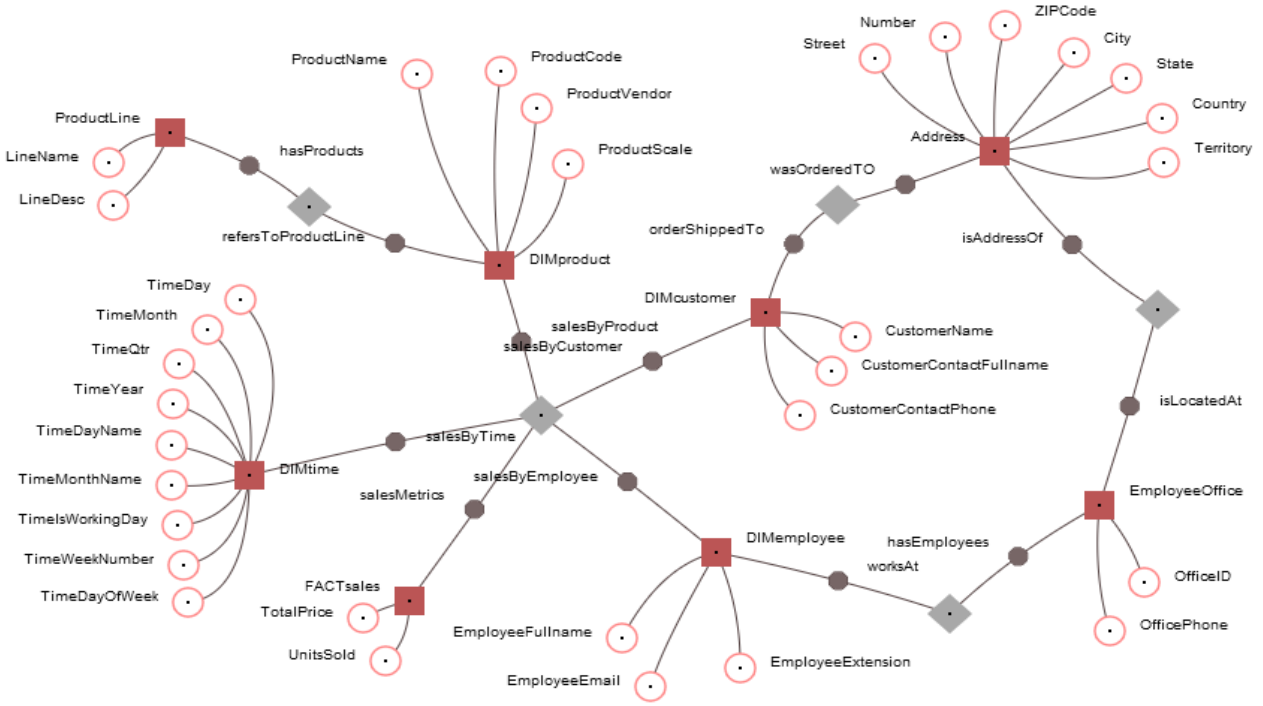

Fig. 4 Initial STAR schema converted into a more decomposed anchor model

information as the default STAR schema and the AM_LESS schema. Table II shows short characteristics of all 3 database schemas after filling database tables with test data.

TABLE II
SHORT CHARACTERISTICS OF TESTED DATABASE SCHEMAS

Database schema	Total tables, excl. indices and views	Total size, incl. indices (MB)	% of STAR's size
STAR	5	2764,56	-
AM_LESS	43	2473,31	89,46
AM_MORE	48	2614,63	94,58

IV. SUMMARY OF QUERY PERFORMANCE RESULTS

In theory, a test run does not fail, if all its requests produce correct answers and the state of the test database is correct after the execution of the test run [6]. So during test runs there were also a percentage of errors recorded. All results show 0% of errors so test runs can be considered as successful. Queries were also validated in terms of output equality. All queries' outputs matched their counterpart in each other database schema. Indices were created according to recommendations of query plan analysis performed before each testing query run was started. Table 3 shows results of query performance tests.

As seen in the table resulting average execution times of 200 queries run in each sequence, both anchor models derived from the STAR schema show higher execution times, in case of query QG_2_4 even 10times slower execution time than on the STAR schema as a possible result of combination of amount of attributes and summarization operations. On the other side queries QG_1_3, QG_2_1 and QG_2_2 were very close in terms of average execution time. Commonly these queries utilized more where conditions with more than 1 filtering value to filter all possible values but still all queries executed on anchor schemas exhibited higher execution time than those executed on the STAR schema despite lower size of both anchor database schemas as seen in Table II. Results gathered during the tests will be further analyzed and the anchor models' lower performance will be researched although only higher performance than traditional ER model based schema was observed [2].

V. CONCLUSION

The paper dealt with the analysis of query performance of anchor and star schemas that modeled the same information structure of data in a sample data mart. Anchor modeling technique is intended to be used as an agile data modeling technique especially in the data warehousing environment. Modularity and flexibility of the model presents possible advantages over traditional ER data modeling when speaking of flexibility of data warehouses' data model. In terms of dimensional modeling and its comparison to anchor modeling the paper show that despite some benefits that anchor modeling brings to the field of information modeling, query performance of database schemas was lower than those built according to standardized dimensional modeling technique. The reasons of this fact will be further analyzed in our research.

TABLE III
SUMMARY OF QUERY PERFORMANCE TESTING RESULTS

	Query group									
	QG_1					QG_2				
Query#	*1*	*2*	*3*	*4*	*5*	*1*	*2*	*3*	*4*	*5*
Wherecond.	3	2	3	2	2	2	3	2	2	2
Sum	2	2	2	2	0	1	2	2	2	2
Subqueries	0	0	0	0	0	1	2	1	1	2
Output cols	3	6	8	10	5	4	3	3	5	4
Output rows	36	450000	14914	140592	404709	65872	200609	11	261844	52
	Average execution time (ms)									
STAR	969	4851	140	2378	2568	522	2338	661	3909	2683
AM_LESS	1728	5905	324	3655	4354	1854	2977	1327	13158	5831
AM_MORE	1714	6150	682	3470	4407	567	2779	1162	10352	5879

ACKNOWLEDGEMENT

This paper was made under financial support of Student Grant Competition, research project SP2012/184 "The analysis of data warehouse's database schema modeling characteristics with a focus on agile approach to Business Intelligence system's development."

REFERENCES

[1] S. Ambler, *Agile Database Techniques: Effective Strategies for the Agile Software Developer*,New Jersey: Wiley, 2003.
[2] O. Regardt, L. Rönnbäck, M. Bergholtz, P. Johannesson, P. Wohed, "Anchor Modeling: An Agile Modeling Technique Using the Sixth Normal Form for Structurally and Temporally Evolving Data", 2009, *ER 2009 [*Lecture Notes in Computer Science, vol. 5829,no.1, pp. 234-250].
[3] H. J. Watson, T. Ariyachandra. *Data Warehouse Architectures: Factors in the Selection Decision and the Success of the Architectures*, Technical Report, Terry College of Business, University of Georgia, Athens, GA, July 2005
[4] C. J. Date, *The Relational Database Dictionary: A Comprehensive Glossary of Relational Terms and Concepts, with Illustrative Examples*,2006, O'Reilly Series Pocket References. O'Reilly Media, Inc.
[5] R. Rob, C. Coronel, K. Crockett, *Database Systems: Design, Implementation & Management*, 2008, London: Cengage Learning EMEA.
[6] A. Askarunisa, P. Prameela, N. Ramraj, "DBGEN- Database (Test) GENerator - An Automated Framework for Database Application Testing". 2009, *International Journal of Database Theory and Application*, vol. 2, no. 3, pp. 27-54.
[7] G. Di Vitantonio, J. Legh-Smith, W. Millar, M. Wilkinson, "Meeting business objectives through adaptive information and communications technology", 2006, *BT Technology Journal*, vol. 24, no. 4, pp. 113-120.