Information Systems & Grid Technologies

Sixth International Conference ISGT'2012

Sofia, Bulgaria, June 1–3., 2012.

ST. KLIMENT OHRIDSKI UNIVERSITY PRESS

ISGT'2012 Conference Committees

Co-chairs

- Prof Ivan SOSKOV
- Prof Kiril BOYANOV

Program Committee

- Míchéal Mac an AIRCHINNIGH, Trinity College, University of Dublin
- Pavel AZALOV, Pennsylvania State University
- Marat BIKTIMIROV, Joint Supercomputer Center, Russian Academy of Sciences
- Marko BONAČ, Academic and Research Network of Slovenia
- Marco DE MARCO, Catholic University of Milan
- Milena DOBREVA, University of Strathclyde, Glasgow
- Viacheslav ILIN, Moscow State University
- Vladimir GETOV, University of Westminster
- Jan GRUNTORÁD, Czech Research and Academic Network
- Pavol HORVATH, Slovak University of Technology
- Seifedine KADRY, American University of the Middle East, Kuwait
- Arto KARILA, Helsinki University of Technology
- Dieter KRANZMUELLER, University of Vienna
- Shy KUTTEN, Israel Institute of Technology, Haifa
- Vasilis MAGLARIS, National Technical University of Athens
- Ivan PLANDER, Slovak Academy of Science
- Dov TE'ENI, Tel-Aviv University
- Stanislaw WRYCZA, University of Gdansk
- Fani ZLATAROVA, Elizabethtown College

Organizing Committee

- Vladimir DIMITROV
- Maria NISHEVA
- Kalinka KALOYANOVA
- Vasil GEORGIEV

Vladimir Dimitrov (Editor)

Information Systems & Grid Technologies

Sixth International Conference ISGT'2012

Sofia, Bulgaria, June 1–3., 2012.

Proceedings

St. Kliment Ohridski University Press

Preface

This conference was being held for the sixth time in the beginning of June, 2012 in the mountain resort Gyolechica near Sofia, Bulgaria. It is supported by the National Science Fund, by the University of Sofia “St. Kliment Ohridski” and by the Bulgarian Chapter of the Association for Information Systems (BulAIS). The Organizing Committee consists of scientists from the Faculty of Mathematics and Informatics of the University of Sofia. Traditionally this conference is organized in cooperation with the Institute of Information and Communication Technologies of the Bulgarian Academy of Sciences.

Total number of papers submitted for participation in ISGT’2012 was 53. They undergo the due selection by at least two of the members of the Program Committee. This book comprises 36 papers of 35 Bulgarian authors and 15 foreign authors included in one of the three conference tracks. In order to facilitate access to the results reported in this conference the papers with higher referee’s score will be uploaded at AIS (Association of Information Systems) chapter “Conferences”. Responsibility for the accuracy of all statements in each peer-reviewed paper rests solely with the author(s). Permission is granted to photocopy or refer to any part of this book for personal or academic use providing credit is given to the conference and to the authors.

The Editor

ISSN 1314-4855
St. Kliment Ohridski University Press

TABLE OF CONTENTS

INFORMATION SYSTEMS

INTELLIGENT SYSTEMS

DISTRIBUTED SYSTEMS

INFORMATION SYSTEMS

Tendencies in Data Warehouse Scope Development

Zlatinka Kovacheva,

Higher College of Telecommunications and Post, Sofia, Bulgaria
Higher College of Technology, Muscat, Oman,
zkovacheva@hotmail.com

Abstract. A data warehouse is a repository of an organization's electronically stored data. Data warehouses are designed to facilitate reporting and analysis. The Data Warehouse environment positions a business to utilize an enterprise-wide data store to link information from diverse sources and make the information accessible for a variety of user purposes, most notably, strategic analysis. Data Warehousing requires both business and technical expertise and involves many activities. In the beginning it is necessary to identify the business information and to prioritize subject areas to be included in the Data Warehouse. This paper concerns the management of the scope of DataWarehouse.

Keywords: data warehouse, data mart, scope, gathering data, operational data, filtering, subject orientation

1 Introduction

Data warehousing began in the 1980s as a response to gather the information provided by the many application systems that were being built. Online applications served the needs of a limited community of users, and they were rarely integrated with each other. Additionally, online applications had no appreciable amount of historical data because they jettisoned their historical data as quickly as possible in the name of high performance. Thus, corporations had lots of data and very little information. Data warehousing began as a way to reduce users' frustration with their inability to get integrated, reliable, accessible data.

Usually Data warehouse is defined as a collection of data that is subject oriented, integrated and time variant for the purpose of management's decision processes [3,4]. Since the 1980s, data warehousing has gone from being a concept that was derided by the database theoreticians to conventional wisdom. Once the idea of the data warehouse became popular, vendors and consultants latched onto the concept as a good way to sell their products.. As a result, there was much confusion over what was and was not a data warehouse. Nowadays, people built different manifestations of data warehouses which fulfilled a real need in the marketplace.

The data in the Warehouse comes from the operational environment and external sources. Data Warehouses are physically separated from operational systems, even though the operational systems feed the Warehouse with source data. All data in the data warehouse is identified with a particular time period.

A data warehouse was the first attempt at architecture that most organizations had ever encountered. Prior to data warehousing, everything had been a new application; however, it became apparent that applications were not going to get the organization where it needed to go over time. The solution was to build an architecture or at least the first fledgling steps of an architecture.

Data Warehouse applications are designed primarily to support executives, senior managers, and business analysts in making complex business decisions. Data Warehouse applications provide the business community with access to accurate, consolidated information from various internal and external sources [10].

The Data Warehouse process guides the development team through identifying the business requirements, developing the business plan and Warehouse solution to business requirements, and implementing the configuration, technical and application architecture for the overall Data Warehouse. It then specifies the iterative activities for the cyclical planning, design, construction, and deployment of each project [8].

One of the first stages of this process is the establishment of the scope of the Warehouse and its intended use. To paraphrase data warehousing author W. H. Inmon [6,7], traditional projects start with requirements and end with data. Data warehousing projects start with data and end with requirements. Once warehouse users see what they can do with the technology, they will want much more. That's why the Data Warehouse scope development is a never ending process.

2 Requirements Gathering

2.1 Source-Driven Requirements Gathering

Source-driven requirements gathering is a method based on defining the requirements by using the source data in production operational systems. This is done by analyzing an ER model of source data if one is available or the actual physical record layouts and selecting data elements deemed to be of interest. The major advantage of this approach is that it is known from the beginning what data can be supplied because it depends on the availability. Another benefit is that it minimizes the time required by the users in the early stages of the project. Of course, by minimizing user involvement, the risk of producing an incorrect set of requirements is increased. Depending on the volume of source data, and the availability of ER models for it, this can also be a very time-consuming approach. Perhaps, some of the user's key requirements may need data that is currently unavailable. Without the opportunity to identify such requirements, there is no chance to investigate what is involved in obtaining external data from outside the organization. Even so, external data can often be of significant value to the business users [1].

A comparison between operational data and Data Warehouse data is given on Table 1[2,4].

Table 1. Operational vs. Data Warehouse data

Main characteristics	**Operational data**	**Data Warehouse data**
Type	Current, transient	Historical, periodic
Form	Raw, detailed, not normalized	Summarized, normalized
Quality	Inconsistencies and errors are included	Quality controlled – accurate with full integrity
Scope	Restricted	Comprehensive
Access	Specialized	Flexible
Update	Many updates	Periodic updates
Usage	Run the business on a current basis	Support managerial decision making

Although similar in nature to modeling and design in the operational world, the actual steps in data warehousing are different. Operational models are typically ER models and the data warehousing models are dimensional models. The data warehouse model looks more physical in nature than a model of an operational system. Probably the feature that most differentiates the data warehouse model from the logical operational model is the denormalization of the dimensions. They have to be organized for the purposes of management and suitable for the end users points of view to the business data.

The result of the source-driven approach is to provide the user with what is available. Relative to dimensional modeling, it can be used to drive out a fairly comprehensive list of the major dimensions of interest to the organization. This could minimize the proliferation of duplicate dimensions across separately developed data marts. Also, analyzing relationships in the source data can identify areas on which to focus your data warehouse development efforts.

2.2 User-Driven Requirements Gathering

User-driven requirements gathering is a method based on defining the requirements by investigating the functions the users perform. This is usually done through a series of meetings and/or interviews with users. The major advantage to this approach is that the focus is on providing what is needed, rather than what is available. In general, this approach has a smaller scope than the source-driven approach. Therefore, it generally produces a useful data warehouse in a shorter time. The users must clearly understand that it is possible that some of the data they need can simply not be made available. If a user is too tightly focused, it is possible to miss useful data that is available in the production systems [1].

User-driven requirements gathering is the approach of choice, especially when developing data marts. For a full-scale data warehouse, both methods are applicable. It would be worthwhile to use the source-driven approach to break the project into manageable pieces, which may be defined as subject areas. The user-driven approach could then be used to gather the requirements for each subject area.

2.3 Requirements Gathering Problems

Usually around 80% of the time building a data warehouse has been spent on extracting, cleaning and loading data [12]. A lot of problems appear during the process of requirements gathering.

Some problems come from OLTP (On-line transaction processing) systems. Some of their errors can be fixed in the Data Warehouse. Sometimes the data in OLTP systems are not validated. Typically once data are in warehouse many inconsistencies are found with fields containing 'descriptive' information. For example, many times no controls are put on customer names. This is going to cause problems for a warehouse user who expects to perform an ad hoc query selecting on customer name. The warehouse developer, again, may have to modify the transaction processing systems or develop (or buy) some data scrubbing technology.

Usually OLTP systems are built by different software tools and the problem of comparability appears. In every big company there are many OLTP systems for the purposes of different activities, e.g. financial, billing, marketing, human recourses and other systems. They are developed by different departments of the company, using different software environments and very often they produce contradictions. The process of uploading the data from these systems into the data warehouse has to solve such conflicts.

A very common problem is to find the need to store data that are not kept in any transaction processing system. For example, when building sales reporting data warehouses, there is often a need to include information on off–invoice adjustments not recorded in an order entry system. In this case the data warehouse developer faces the possibility of modifying the transaction processing system or building a system dedicated to capturing the missing information [10].

Another problem is so called 'granularity' problem. Some transaction processing systems don't contain details. This problem is often encountered in customer or product oriented warehousing systems. Often it is found that a system which contains information that the designer would like to feed into the warehousing system does not contain information down to the product or customer level.

In addition to understanding the feeder system data, it can be advantageous to build some of the “cleaning” logic on the feeder system platform if that platform is a mainframe. Often cleaning involves a great deal of sort/merging – tasks at which mainframe utilities often excel. Also, it is possible to build aggregates on the mainframe because aggregation also involves substantial sorting.

Very often, the end users increase their requirements when they started to use the data warehouse and realize its real advantages. It comes about because the query and report tools allow the user the users to gain a much better appreciation of what technology could do.

Many warehouse users define conflicting business rules or they don’t know how to use the data even if they passed training courses. That’s why periodically training of the users is required, to provide a contemporary view to the information of the updated data warehouse.

Large scale data warehousing can become an exercise in data homogenizing. A popular way to design a decision support relational databases is with star or snowflake schemas. Persons taking this approach usually also build aggregate fact tables. If there are many dimensions to the data, the combination of the aggregate tables and indexes to the fact tables and aggregate fact tables can use many times more space than the raw data. If multidimensional databases are used, certain products pre–calculate and store summarized data. As with star/snowflake schemas, storage of this calculated data can use far more storage than the raw data.

Security problems are also very important, especially if the data warehouse is Web-accessible. The information of the data warehouse can be uploaded and downloaded by many authorized users for different purposes. That’s why the access to the data should be managed properly and the data on each level of summarization should be well protected.

Reorganizations, product introductions, new pricing schemes, new customers, changes in production systems, etc. are going to affect the warehouse. If the warehouse is going to stay up to date, changes to the warehouse have to be made fast. Customer management issues require a proper maintenance of the data.

3 Data Warehouse Scope Establishment as a Stage of the Data Warehouse Process

The Data Warehouse process is conducted in an iterative fashion after the initial business requirements and architectural foundations have been developed with the emphasis on populating the Data Warehouse with functional subject-area information in each iteration. The process guides the development team through identifying the business requirements, developing the business plan and Warehouse solution to business requirements, and implementing the configuration, technical, and application architecture for the overall Data Warehouse. It then specifies the iterative activities for the cyclical planning, design, construction, and deployment of each population project.

The most important strategic initiative is analyzed to determine the specific business questions that need to be answered through a Warehouse implementation. Each business question is assessed to determine its overall importance to the organization, and a high-level analysis of the data needed to provide the answers is undertaken. A business question can be answered through objective analysis of the data that is available. The data is assessed for quality, availability, and cost associated with bringing it into the Data Warehouse. The business questions are then revisited and prioritized based upon their relative importance and the cost and feasibility of acquiring the associated data. The prioritized list of business questions is used to determine the scope of the first and subsequent iterations of the Data Warehouse, in the form of population projects. Iteration scoping is dependent on source data acquisition issues and is guided by determining how many business questions can be answered in a three to six month implementation time frame. A "business question" is a question deemed by the business to provide useful information in determining strategic direction.

3.1 Validating the Model

Before investing a lot of time and effort in designing the warehouse, it is absolutely necessary to validate the model with the end users. The purpose of such a review is twofold. First, it serves to confirm that the model can actually meet the users' requirements. Second, and just as important, a review should confirm that the user can understand the model. Once the warehouse is implemented, the user will be relying on the model on a regular basis to access data in the warehouse. No matter how well the model meets the users' requirements, the warehouse will fail if the user cannot understand the model and, consequently, cannot access the data. Validation at this point is done at a high level. This model is reviewed with the user to confirm that it is understandable. Together with the user, the model has to be tested by resolving how it will answer some of the questions identified in the requirements. It is almost certain that the model will not meet all of the users' requirements.

Either the requirements need to be better understood or, as is often the case, they have changed and need to be redefined. Usually, this will lead to additions, and possibly changes, to the model already created. In the mean time, the validated portion of the model will go through the design phase and begin providing benefits to the user. The iteration of development and the continued creation of partially complete models are the key elements that provide the ability to rapidly develop data warehouses.

3.2 Identifying the Sources

Once the validated portion of the model passes on to the design stage, the first step is to identify the sources of the data that will be used to load the model. These sources should then be mapped to the target warehouse data model. Mapping should be done for each dimension, dimension attribute, fact, and measure. For dimensions and facts, only the source entities (for example, relational tables, flat files, IMS DBDs and segments) need be documented. For dimension attributes and measures, along with the source entities, the specific source attributes (such as columns and fields) must be documented.

Conversion and derivation algorithms must also be included in the metadata. At the dimension attribute and measure level, this includes data type conversion, algorithms for merging and splitting source attributes, calculations that must be performed, domain conversions, and source selection logic [9]. A domain conversion is the changing of the domain in the source system to a new set of values in the target. Such a conversion should be documented in the metadata.

In some cases target attributes are loaded from different source attributes based on certain conditions. At the fact and dimension level, conversion and derivation metadata includes the logic for merging and splitting rows of data in the source, the rules for joining multiple sources, and the logic followed to determine which of multiple sources will be used.

Identifying sources can also cause changes to the model. This will occur when a valid source is not available. Some time there is no source that comes close to meeting the users' requirements. This should be a very rare case, but it is possible. If only a portion of the model is affected, it is recommended to remove that component and continue designing the remainder. A better scenario happens when there will be a source that comes close but doesn't exactly meet the user's requirements. In this case it will be necessary only to modify slightly the model.

4 Some Tendencies in Data Warehouse scope Development

Data Warehouse is developed in an environment where technology could be changed as fast as the business requirements change. The Data Warehouse environment has features like life cycle, integration of structured and unstructured data, metadata, etc. In order to adapt to the ongoing change of the business, semantically temporal data (data that undergoes changes) should be separated from semantically static data, which does not change over time. When this happens a new snapshot of the data is created. Each snapshot is delimited by time and a historical record for the semantically temporal data is created. In order to meet the changes in the environment, a new generation of Data Warehouse has been developed – DW 2.0 [5].

The analysis of the characteristics of DW 2.0 presents the basic differences between the two generations data warehouses. They cover crucial aspects of the data warehousing and span from the building methodology, through the development, to the implementation of the final product.

Several fundamental aspects underlie the difference between DW 2.0 and the first-generation data warehousing. The new features of DW 2.0 concerning the scope of the Data Warehouse are the following [6,7]:

4.1 The Lifecycle of Data

As data ages, its characteristics change. As a consequence, the data in DW 2.0 is divided into different sectors based on the age of the data. In the first generation of data warehouses, there was no such distinction.

According to the concept of "life cycle of data" data passes through four different sectors:

- The Interactive Sector
- The Integrated Sector
- The Near Line Sector
- The Archival Sector

The Interactive Sector is the place where the data enters the DW 2.0 environment either from an application outside DW 2.0 or as part of a transaction from an application which is located in the Interactive Sector. On leaving the Interactive Sector the data goes to the Integrated Sector. At this moment "data quality processing" like "domain checking and range checking" is done. Data is collected and kept with maximum time span of three to five years. After leaving the Integrated Sector the data can go either to the Near Line Sector or to the Archival Sector.

4.2 Structured and Unstructured Data

Another aspect of the DW 2.0 environment is that it contains not only structured but also unstructured data. Unstructured data is a valid part of the DW 2.0 [14]. Some of the most valuable information in the corporation resides in unstructured data. The first generation of data warehouses did not recognize that there was valuable data in the unstructured environment and that the data belonged in the data warehouse.

Two basic forms of unstructured data – "textual" and "non-textual" are supported. Textual unstructured data could be found in emails, telephone conversations, spreadsheets, documents and so forth. Non-textual unstructured data occurs as graphics and images. The integration of structured and unstructured data in DW 2.0 enables different kinds of analysis - against unstructured data or a combination of structured and unstructured data.

Unstructured data exists in several forms in DW 2.0 - actual snippets of text, edited words and phrases, and matching text. The most interesting of these forms of unstructured data in the DW 2.0 environment is the matching text. In the structured environment, matches are made positively and surely. Not so with unstructured data. In DW 2.0, when matches are made, either between unstructured data and unstructured data or between unstructured data and structured data, the match is said to be probabilistic. The match may or may not be valid, and a probability of an actual match can be calculated or estimated. The concept of a probabilistic match is hard to fathom for the person that has only dealt with structured systems, but it represents the proper way to link structured and unstructured data.

4.3 A New Concept of Metadata

Metadata is the glue that holds the data together over its different states [11]. The first generation of data warehousing omitted metadata as part of the infrastructure.

Metadata is found in many places today - multidimensional technology, data warehouses, database management system catalogs, spreadsheets, documents, etc. There is little or no coordination of metadata from one architectural construct to another; however, there is still a need for a global repository. These sets of needs are recognized and addressed architecturally in DW 2.0.

In DW 2.0 every sector has metadata of its own. The metadata for the Archival Sector is placed directly with the archival data while the metadata for the other sectors is kept separately from the data. The different sectors support metadata for the structured and for the unstructured data.

4.4 Integrity of the Data

The need for integrity of data is recognized as data passes from online processing to integrated processing [13].

The DW 2.0 environment has features like integration of structured and unstructured data and metadata. In order the technology to adapt to the ongoing change of the business, semantically temporal data (data that undergoes changes) should be separated from semantically static data, which does not change over time. When this happens a new snapshot of the data is created. Each snapshot is delimited by time and has to date and from date. In this way a historical record for the semantically temporal data is created.

4.5 The Spiral Methodology

One of the most prominent characteristics of DW 2.0 is the spiral methodology, which adopts the so called “seven streams approach”. Different activities are initiated and completed in each stream [5]:

- The Enterprise reference model stream concerns the creation and continued maintenance of a corporate data model which consists of Entity Relationship Diagrams and background information;
- The Enterprise knowledge coordination stream relies on various artifacts that come out of the corporate data modeling, information factory development, and data profiling streams;
- The Information factory development stream is built “topic by topic”;
- The Data profiling and mapping stream includes the analysis of the data from the source systems in terms of quality and completeness;
- The Data correction stream - the data is examined and decided which part of it will be corrected;
- The Infrastructure management stream addresses the decision what people resources and tools will be used;
- The Total information quality management stream provides data quality monitoring and process improvement.

5 Conclusion

The analysis of the characteristics of DW 2.0 presents the basic differences between the two generations data warehouses. They cover crucial aspects of the data warehousing from the building methodology, through the development, to the implementation of the final product.

The huge amount of unstructured and unprocessed data from different sources from one side and a large number of on-line transaction processing systems organized on different platforms from the other side have to be integrated uniformly in order to satisfy the rapidly increasing business requirements.

The tendencies concerning Data Warehouse Scope development are discussed in this paper. They follow the process of enlarging and complexness of the data stored nowadays, the increasing number of data sources of different type and the necessity of integration of the data in order to provide more powerful information for the purposes of the management, marketing, financial and all other activities in the fast growing and rapidly changing business environment.

References

1. Ballard Ch., Herreman D., Schau D., Bell R., Kim E., Valencic A.: Data Modeling Techniques for Data Warehousing International Technical Support Organization, February 1998, http://www.redbooks.ibm.com ,SG24-2238-00 IBM
2. Browning D., Mundy J.: Data Warehouse Design Considerations, Microsoft® SQL Server™ 2000, December 2001
3. Chaudhuri, Surajit and Dayal, Umeshwar: An Overview of Data Warehousing and OLAP Technology, ACM New York, NY, USA (1997)
4. Hoffer J. A. , Prescott M. B., McFadden F.R.:2 Modern Database Management, 7th Edition, Prentice Hall, (2005)
5. Hristov N., Kaloyanova K.: Applying a Seven Streams Approach for Building a DW 2.0 , Proceedings of the Fourth International Conference On Information Systems & Datagrids, Sofia, Bulgaria, May 2010
6. Inmon, William and Strauss, Derek and Neushloss, Genia: DW 2.0: The Architecture for the Next Generation of Data Warehousing, Elsevier Inc. (2008)
7. Inmon, Bill: Different Kinds of Data Warehouses. http://www.b-eye network.com/view/10416, (2009)
8. Naydenova I. , Kaloyanova K.: Basic Approaches Of Integration Between Data Warehouse And Data Mining,
   Proceedings of the First International Conference On Information Systems & Datagrids , Sofia, Bulgaria, (2005), http://isgt.fmi.uni-sofia.bg/index.php?contentPage=proceeding
9. Naydenova I. , Kaloyanova K.: An Approach Of Non-Additive Measures Compression In Molap Enviroment", Proceedings of the IADIS Multi Conference on Computer Science and Information Systems, 2007, 3 - 8 July 2007, Lisbon, Portugal, ISBN: 978–972–8924–35-5,pp. 394-399, http://www.mccsis.org/2007/
10. Naydenova I. , Kaloyanova K., Ivanov St.: A Multi-Source Customer Identification", Proceedings of the Third International Conference On Information Systems & Datagrids, Sofia, Bulgaria, pp.77-85, 28 - 29 May 2009
    http://ci.fmi.uni-sofia.bg/isgt2009/
11. Poole J.: Model-Driven Data Warehousing, Hyperion Solutions Corporation, Burlingame, CA, January 2003
12. Rahm, Erhard and Do,H. H.: Data Cleaning: Problems and Current Approaches, (2000)
13. Russom, Ph.: Complex Data: A New Challenge for Data Integration, (2007)
14. Santos, Ricardo Jorge, Bernardino, Jorge: Optimizing Data Warehouse Loading Procedures for Enabling Useful-Time Data Warehousing, (2009)