Data Warehousing and Knowledge Discovery

Book *in* Lecture Notes in Computer Science · January 2008

DOI: 10.1007/978-3-540-85836-2

Il-Yeol Song
Drexel University
304 PUBLICATIONS **4,053** CITATIONS

Johann Eder
Alpen-Adria-Universität Klagenfurt
282 PUBLICATIONS **2,739** CITATIONS

Tho Manh Nguyen
TU Wien
32 PUBLICATIONS **280** CITATIONS

A Min Tjoa   Juan Trujillo (Eds.)

Data Warehousing and Knowledge Discovery

7th International Conference, DaWaK 2005
Copenhagen, Denmark, August 22-26, 2005
Proceedings

Springer

Volume Editors

A Min Tjoa
Vienna University of Technology
Institute of Software Technology and Interactive Systems
Favoriten Str. 9-11/188, 1040 Vienna, Austria
E-mail: amin@ifs.tuwien.ac.at

Juan Trujillo
University of Alicante
Dept. of Language and Information Systems
Apto. Correos 99 E-03080, C.P. 03690 Alicante, Spain
E-mail: jtrujillo@dlsi.ua.es

Library of Congress Control Number: Applied for

CR Subject Classification (1998): H.2, H.3, H.4, C.2, H.5, I.2, J.1

ISSN 0302-9743
ISBN-10 3-540-28558-X Springer Berlin Heidelberg New York
ISBN-13 978-3-540-28558-8 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11546849 06/3142 5 4 3 2 1 0

Table of Contents

Data Warehouse I

A Tree Comparison Approach to Detect Changes in Data Warehouse Structures
Johann Eder, Christian Koncilia, Karl Wiggisser 1

Extending the UML for Designing Association Rule Mining Models for Data Warehouses
José Jacobo Zubcoff, Juan Trujillo 11

Event-Feeded Dimension Solution
Tho Manh Nguyen, Jaromir Nemec, Martin Windisch 22

XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses
Byung-Kwon Park, Hyoil Han, Il-Yeol Song 32

Data Warehouse II

Graph-Based Modeling of ETL Activities with Multi-level Transformations and Updates
Alkis Simitsis, Panos Vassiliadis, Manolis Terrovitis, Spiros Skiadopoulos 43

Extending UML 2 Activity Diagrams with Business Intelligence Objects
Veronika Stefanov, Beate List, Birgit Korherr 53

Automatic Selection of Bitmap Join Indexes in Data Warehouses
Kamel Aouiche, Jérôme Darmont, Omar Boussaïd, Fadila Bentayeb 64

Evaluating Data Warehouses and Tools

A Survey of Open Source Tools for Business Intelligence
Christian Thomsen, Torben Bach Pedersen 74

DWEB: A Data Warehouse Engineering Benchmark
Jérôme Darmont, Fadila Bentayeb, Omar Boussaïd 85

A Set of Quality Indicators and Their Corresponding Metrics for Conceptual Models of Data Warehouses
Gema Berenguer, Rafael Romero, Juan Trujillo, Manuel Serrano, Mario Piattini 95

Design and Development of a Tool for Integrating Heterogeneous Data Warehouses
Riccardo Torlone, Ivan Panella 105

Schema Transformations

An Evolutionary Approach to Schema Partitioning Selection in a Data Warehouse
Ladjel Bellatreche, Kamel Boukhalfa 115

Using Schema Transformation Pathways for Incremental View Maintenance
Hao Fan 126

Data Mapper: An Operator for Expressing One-to-Many Data Transformations
Paulo Carreira, Helena Galhardas, João Pereira, Antónia Lopes 136

Materialized Views

Parallel Consistency Maintenance of Materialized Views Using Referential Integrity Constraints in Data Warehouses
Jinho Kim, Byung-Suk Lee, Yang-Sae Moon, Soo-Ho Ok, Wookey Lee 146

Selective View Materialization in a Spatial Data Warehouse
Songmei Yu, Vijayalakshmi Atluri, Nabil Adam 157

PMC: Select Materialized Cells in Data Cubes
Hongsong Li, Houkuan Huang, Shijin Liu 168

Aggregates

Progressive Ranking of Range Aggregates
Hua-Gang Li, Hailing Yu, Divyakant Agrawal, Amr El Abbadi 179

On Efficient Storing and Processing of Long Aggregate Lists
Marcin Gorawski, Rafal Malczok 190

Data Warehouse Queries and Database Processing Issues

Ad Hoc Star Join Query Processing in Cluster Architectures
Josep Aguilar-Saborit, Victor Muntés-Mulero, Calisto Zuzarte, Josep-L. Larriba-Pey 200

A Precise Blocking Method for Record Linkage
Patrick Lehti, Peter Fankhauser 210

Flexible Query Answering in Data Cubes
Sami Naouali, Rokia Missaoui 221

An Extendible Array Based Implementation of Relational Tables for Multi Dimensional Databases
K.M. Azharul Hasan, Masayuki Kuroda, Naoki Azuma, Tatsuo Tsuji, Ken Higuchi 233

Data Mining Algorithms and Techniques

Nearest Neighbor Search on Vertically Partitioned High-Dimensional Data
Evangelos Dellis, Bernhard Seeger, Akrivi Vlachou 243

A Machine Learning Approach to Identifying Database Sessions Using Unlabeled Data
Qingsong Yao, Xiangji Huang, Aijun An 254

Hybrid System of Case-Based Reasoning and Neural Network for Symbolic Features
Kwang Hyuk Im, Tae Hyun Kim, Sang Chan Park 265

Data Mining

Spatio-temporal Rule Mining: Issues and Techniques
Győző Gidófalvi, Torben Bach Pedersen 275

Hybrid Approach to Web Content Outlier Mining Without Query Vector
Malik Agyemang, Ken Barker, Reda Alhajj 285

Incremental Data Mining Using Concurrent Online Refresh of Materialized Data Mining Views
Mikołaj Morzy, Tadeusz Morzy, Marek Wojciechowski, Maciej Zakrzewicz 295

A Decremental Algorithm for Maintaining Frequent Itemsets in Dynamic Databases
Shichao Zhang, Xindong Wu, Jilian Zhang, Chengqi Zhang 305

Association Rules

Discovering Richer Temporal Association Rules from Interval-Based Data
Edi Winarko, John F. Roddick 315

Semantic Query Expansion Combining Association Rules with Ontologies and Information Retrieval Techniques
Min Song, Il-Yeol Song, Xiaohua Hu, Robert Allen 326

Maintenance of Generalized Association Rules Under Transaction Update and Taxonomy Evolution
Ming-Cheng Tseng, Wen-Yang Lin, Rong Jeng 336

Prince: An Algorithm for Generating Rule Bases Without Closure Computations
Tarek Hamrouni, Sadok Ben Yahia, Yahya Slimani 346

Text Processing and Classification

Efficient Compression of Text Attributes of Data Warehouse Dimensions
Jorge Vieira, Jorge Bernardino, Henrique Madeira 356

Effectiveness of Document Representation for Classification
Ding-Yi Chen, Xue Li, Zhao Yang Dong, Xia Chen 368

2-PS Based Associative Text Classification
Tieyun Qian, Yuanzhen Wang, Hao Long, Jianlin Feng 378

Miscellaneous Applications

Intrusion Detection via Analysis and Modelling of User Commands
Matthew Gebski, Raymond K. Wong 388

Dynamic Schema Navigation Using Formal Concept Analysis
Jon Ducrou, Bastian Wormuth, Peter Eklund 398

Security and Privacy Issues

FMC: An Approach for Privacy Preserving OLAP
Ming Hua, Shouzhi Zhang, Wei Wang, Haofeng Zhou, Baile Shi 408

Information Driven Evaluation of Data Hiding Algorithms
Elisa Bertino, Igor Nai Fovino 418

Patterns

Essential Patterns: A Perfect Cover of Frequent Patterns
Alain Casali, Rosine Cicchetti, Lotfi Lakhal 428

Processing Sequential Patterns in Relational Databases
Xuequn Shang, Kai-Uwe Sattler 438

Optimizing a Sequence of Frequent Pattern Queries
Mikołaj Morzy, Marek Wojciechowski, Maciej Zakrzewicz 448

A General Effective Framework for Monotony and Tough Constraint Based Sequential Pattern Mining
Enhong Chen, Tongshu Li, Phillip C-y Sheu 458

Cluster and Classification I

Hiding Classification Rules for Data Sharing with Privacy Preservation
Juggapong Natwichai, Xue Li, Maria Orlowska 468

Clustering-Based Histograms for Multi-dimensional Data
Filippo Furfaro, Giuseppe M. Mazzeo, Cristina Sirangelo 478

Weighted K-Means for Density-Biased Clustering
Kittisak Kerdprasop, Nittaya Kerdprasop, Pairote Sattayatham 488

Cluster and Classification II

A New Approach for Cluster Detection for Large Datasets with High Dimensionality
Matthew Gebski, Raymond K. Wong 498

Gene Expression Biclustering Using Random Walk Strategies
Fabrizio Angiulli, Clara Pizzuti 509

Spectral Kernels for Classification
Wenyuan Li, Kok-Leong Ong, Wee-Keong Ng, Aixin Sun 520

Data Warehousing and Knowledge Discovery: A Chronological View of Research Challenges
Tho Manh Nguyen, A Min Tjoa, Juan Trujillo 530

Author Index 537

Author Index

Abbadi, Amr El 179
Adam, Nabil 157
Agrawal, Divyakant 179
Aguilar-Saborit, Josep 200
Agyemang, Malik 285
Alhajj, Reda 285
Allen, Robert 326
An, Aijun 254
Angiulli, Fabrizio 509
Aouiche, Kamel 64
Atluri, Vijayalakshmi 157
Azuma, Naoki 233

Barker, Ken 285
Bellatreche, Ladjel 115
Bentayeb, Fadila 64, 85
Berenguer, Gema 95
Bernardino, Jorge 356
Bertino, Elisa 418
Boukhalfa, Kamel 115
Boussaïd, Omar 64, 85

Carreira, Paulo 136
Casali, Alain 428
Chen, Ding-Yi 368
Chen, Enhong 458
Chen, Xia 368
Cicchetti, Rosine 428

Darmont, Jérôme 64, 85
Dellis, Evangelos 243
Dong, Zhao Yang 368
Ducrou, Jon 398

Eder, Johann 1
Eklund, Peter 398

Fan, Hao 126
Fankhauser, Peter 210
Feng, Jianlin 378
Fovino, Igor Nai 418
Furfaro, Filippo 478

Galhardas, Helena 136
Gebski, Matthew 388, 498
Gidófalvi, Győző 275
Gorawski, Marcin 190

Hamrouni, Tarek 346
Han, Hyoil 32
Hasan, K.M. Azharul 233
Higuchi, Ken 233
Hu, Xiaohua 326
Hua, Ming 408
Huang, Houkuan 168
Huang, Xiangji 254

Im, Kwang Hyuk 265

Jeng, Rong 336

Kerdprasop, Kittisak 488
Kerdprasop, Nittaya 488
Kim, Jinho 146
Kim, Tae Hyun 265
Koncilia, Christian 1
Korherr, Birgit 53
Kuroda, Masayuki 233

Lakhal, Lotfi 428
Larriba-Pey, Josep-L. 200
Lee, Byung-Suk 146
Lee, Wookey 146
Lehti, Patrick 210
Li, Hongsong 168
Li, Hua-Gang 179
Li, Tongshu 458
Li, Wenyuan 520
Li, Xue 368, 468
Lin, Wen-Yang 336
List, Beate 53
Liu, Shijin 168
Long, Hao 378
Lopes, Antónia 136

Madeira, Henrique 356
Malczok, Rafal 190
Mazzeo, Giuseppe M. 478
Missaoui, Rokia 221
Moon, Yang-Sae 146

Morzy, Mikołaj 295, 448
Morzy, Tadeusz 295
Muntés-Mulero, Victor 200

Naouali, Sami 221
Natwichai, Juggapong 468
Nemec, Jaromir 22
Ng, Wee-Keong 520
Nguyen, Tho Manh 22, 530

Ok, Soo-Ho 146
Ong, Kok-Leong 520
Orlowska, Maria 468

Panella, Ivan 105
Park, Byung-Kwon 32
Park, Sang Chan 265
Pedersen, Torben Bach 74, 275
Pereira, João 136
Piattini, Mario 95
Pizzuti, Clara 509

Qian, Tieyun 378

Roddick, John F. 315
Romero, Rafael 95

Sattayatham, Pairote 488
Sattler, Kai-Uwe 438
Seeger, Bernhard 243
Serrano, Manuel 95
Shang, Xuequn 438
Sheu, Phillip C-y 458
Shi, Baile 408
Simitsis, Alkis 43
Sirangelo, Cristina 478
Skiadopoulos, Spiros 43
Slimani, Yahya 346
Song, Il-Yeol 32, 326
Song, Min 326
Stefanov, Veronika 53
Sun, Aixin 520

Terrovitis, Manolis 43
Thomsen, Christian 74
Tjoa, A Min 530
Torlone, Riccardo 105
Trujillo, Juan 11, 95, 530
Tseng, Ming-Cheng 336
Tsuji, Tatsuo 233

Vassiliadis, Panos 43
Vieira, Jorge 356
Vlachou, Akrivi 243

Wang, Wei 408
Wang, Yuanzhen 378
Wiggisser, Karl 1
Winarko, Edi 315
Windisch, Martin 22
Wojciechowski, Marek 295, 448
Wong, Raymond K. 388, 498
Wormuth, Bastian 398
Wu, Xindong 305

Yahia, Sadok Ben 346
Yao, Qingsong 254
Yu, Hailing 179
Yu, Songmei 157

Zakrzewicz, Maciej 295, 448
Zhang, Chengqi 305
Zhang, Jilian 305
Zhang, Shichao 305
Zhang, Shouzhi 408
Zhou, Haofeng 408
Zubcoff, José Jacobo 11
Zuzarte, Calisto 200

A Set of Quality Indicators and Their Corresponding Metrics for Conceptual Models of Data Warehouses

Gema Berenguer[1], Rafael Romero[1], Juan Trujillo[1], Manuel Serrano[2], and Mario Piattini[2]

[1] Dept. de Lenguajes y Sistemas Informáticos,
Universidad de Alicante,
Apto. Correos 99. E-03080. Alicante.
{gberenguer, romero, jtrujillo}@dlsi.ua.es
[2] Escuela Superior de Informática,
University of Castilla – La Mancha,
Paseo de la Universidad, 4, 13071 Ciudad Real
{Manuel.Serrano, Mario.Piattini}@uclm.es

Abstract. The quality of Data Warehouses is absolutely relevant for organizations in the decision making process. The sooner we can deal with quality metrics (i.e. conceptual modelling), the more willing we are in achieving a data warehouse (DW) of a high quality. From our point of view, there is a lack of more objective indicators (metrics) to guide the designer in accomplishing an outstanding model that allows us to guarantee the quality of these data warehouses. However, in some cases, the goals and purposes of the proposed metrics are not very clear on their own. Lately, quality indicators have been proposed to properly define the goals of a measurement process and group quality measures in a coherent way. In this paper, we present a framework to design metrics in which each metric is part of a quality indicator we wish to measure. In this way, our method allows us to define metrics (theoretically validated) that are valid and perfectly measure our goals as they are defined together a set of well defined quality indicators.

Keywords: Quality indicators, quality metrics, conceptual modelling, data warehouses, multidimensional modelling

1 Introduction

Data Warehouses (DWs), which are the core of current decision support systems, provide companies with many years of historical information for the decision making process [10]. The term data warehouse is defined as "a subject-oriented, integrated, time-variant, non-volatile collection of data supporting management's decisions" [8]. A lack of quality in the data warehouse can be disastrous consequences from both a technical and organizational point of view. Therefore, it is crucial for an organization to guarantee the quality of the information contained in these DWs. The information quality of a DW is determined by (i) the quality of the DBMS (Database Management System), (ii) the quality of the data models used in their design, (iii) the quality of the data themselves contained in the data warehouse (see figure 1).

A Min Tjoa and J. Trujillo (Eds.): DaWaK 2005, LNCS 3589, pp. 95 – 104, 2005.

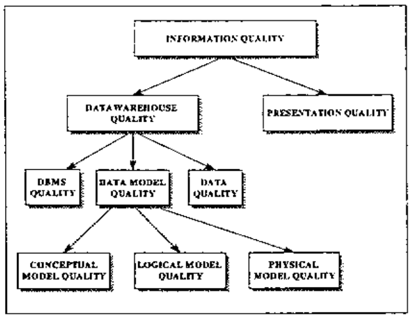

Fig. 1. Quality of the information and the data warehouse

In order to guarantee the quality of the DBMS, we can use an International Standard such as ISO/IEC 9126 [9] or one of the comparative studies of existing products. The quality of the datawarehouse model also strongly influences information quality. The model can be considered at three levels: conceptual, logical and physical. Due to space constraints, we refer the reader to [1] for a deep comparison of conceptual, logical and physical models proposed for data warehouses. At the logical level several recommendations exist in order to create a good dimensional data model [11] and in recent years we have proposed and validated both theoretically and empirically several metrics that enable the evaluation of the complexity of star models. At the physical model depends on each system and consist of selecting the physical tables, indexes, data partitions, etc. [2] [11].

However, from our point of view, we claim that design guidelines or subjective quality criteria are not enough to guarantee the quality of multidimensional models for DWs. Therefore, we believe that a set of formal and quantitative measures should be provided to reduce subjectivity and bias in evaluation, and guide the designer in his work. However, we cannot assure that quality measures interpret a measurable concept on their own with guarantee. So, lately, quality indicators have been proposed to define the concept to be measured and group the quality measures needed to measure that indicator [7]. Otherwise, we may propose metrics that cannot measure what they are defined for and they may overlap the measurable concept.

In this paper, we firstly propose a set of quality indicators to measure the quality of conceptual schemas for DWs. Then, once these indicators clearly establish the set of concepts to be measured, we define the set of the corresponding quality metrics that will measure that indicator. On defining these indicators and quality metrics, we use our conceptual modelling approach, based on the Unified Modelling Language (UML), to properly accomplish the conceptual modelling of data warehouses [13]. In this paper, we have focused in the first step of the conceptual modelling of DWs and we will use the package diagrams to model complex and huge DWs thereby facilitating their modelling and understanding [13]. Then, we use our quality indicators and measures to an example to show the benefit of our proposal. Due to space constraints, we cannot provide the theoretical validation we have accomplished using both the (i) axiomatic approach and (ii) the measure theory.

The rest of the paper is structured as follow: section 2 presents the method we follow for defining and obtaining correct quality indicators and metrics. Section 3 presents a summary of UML package diagrams we use in this paper for the conceptual modelling of data warehouses. In Section 4, we define the proposed quality indicators and metrics and present some examples to show how to apply them. Finally, section 5 draws conclusions and immediate future works arising from the conclusions reached in this work.

2 A Method to Define Quality Indicators and Metrics

A measurable concept is an abstract relation between attributes of one or more entities, and a necessity of information. Some examples of measurable concepts are: quality, reliability, accessibility and so on. Metrics cannot interpret on their own a measurable concept, and therefore, it is essential to use quality indicators [7]. A metric assess a characteristic of an object while an indicator will use one or more metrics to measure something. Thus, indicators are the basis for (i) quantify measurable concepts for a necessity of information, (ii) quantitative methods of evaluation or prediction, and (iii) to provide information to take decisions.

The definition of quality indicators and metrics has to be accomplished in a methodological way, which makes necessary to accomplish a set of stages to be able to assure their liability. Next, we will present a modification of the methods proposed in [5] to define quality metrics, and the method proposed in MMLC (Measure Model Life Cycle) [6]; to allow us to incorporate the definition of quality indicators and metrics in an overall approach.

This method can be structured into three main phases: (i) creating the indicator, (ii) defining the required metrics for the indicator and (iii) applying these metrics to measure a conceptual schema. In the first phase, we have to find the main objective that we pursue, and then, define the corresponding indicator to achieve that objective. Next, in the second phase, we define the list of required metrics that will allow us to measure the indicator. On creating a metric, we will firstly define it, and then, we have to accomplish the theoretical and empirical validation [5]. At the end of this paper, we will present a summary of the frameworks we use for the theoretical validation of our metrics.

To accomplish the empirical validation of metrics, we need to set a family of experiments [5], from which we will obtain a set of thresholds that will be later applied to the indicator algorithm. Once the metric has been properly defined, we pass to the third phase by applying the obtained metrics to a conceptual schema. With the valid metrics and the thresholds obtained from the empirical validation, we will define the algorithm to measure the indicator. Finally, after analyzing the results obtained by the indicator algorithm, we will store and communicate these results.

2.1 Indicator Template

There are some organizations that do not achieve the expected benefits of applying quality indicators due to the fact that these quality indicators have not been properly specified or they are not properly interpreted [7]. Therefore, we will document the specification of indicators, their interpretation and use as proposed in [7], in order to avoid inconsistencies in their definitions.

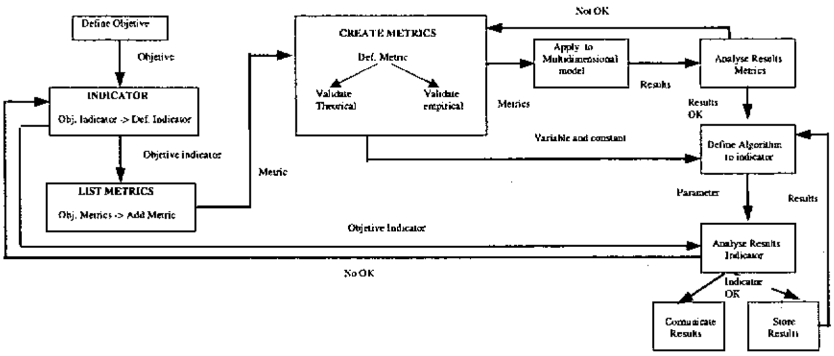

Fig. 2. Method for defining quality indicators and metrics

The Software Engineering Institute (SEI) has found that an indicator template can help an organization to improve its software measurement processes and infrastructure [7]. In this work, authors describe a template that can be used to precisely describe, document, and report *who, what, when, where, why*, and *how* to define organization's indicators. Moreover, they also describe the use of the indicator template within the context of the Goal-Driven Software Measurement (GQ(I)M) methodology and the Capability Maturity Model Integration (CMMI) framework.

Therefore, due to the high importance of quality indicators in our proposal, in the following, we will present the indicator template we have followed – proposed in [7]. Thus, our indicator template consists of:

- *Indicator objective*: the objective or purpose of the indicator
- *Questions*: the questions that the user of the indicator is trying to answer
- *Visual display*: a graphical view of the indicator
- *Perspective or viewpoint*: the description of the audience for whom the indicator is intended
- *Inputs*: the list of the measures required to construct the indicator and its definitions
- *Algorithms*: the description of the algorithm used to construct the indicator from the measures
- *Assumptions*: the list of assumptions about the organization, its processes, life-cycle model, and so on that are important conditions for collecting and using the indicator.
- *Data collection information:* information pertaining to how, when, how often, by whom, etc. the data elements required to construct the indicator are collected.
- *Data reporting information*: information on who is responsible for reporting the data, to whom, and how often.
- *Data storage*: information on storage, retrieval, and security of the data.
- *Analysis and interpretation of results:* information on how to analyze and interpret as well as to not misinterpret the indicator.

At this point, we have stated the reason why we use quality indicators and the corresponding template use to define them. Thus, in Table 1, we match each relevant step of our method (see Figure 2) with the corresponding indicator template issue.

Table 1. Correspondence between our indicator template issues and method phases

Indicator Template	Phase Method
Indicator Name / Title	Indicator
Objective	Indicator
Questions	Indicator
Visual Display	Communicate results
Inputs	Create metrics
Data Collection	Apply multidimensional model
Data Reporting	Communicate results
Data Storage	Store results
Algorithm	Define algorithm indicator
Interpretation	Store results
Analysis	Analyze Data

3 Multidimensional Modelling with Package Diagrams of UML

In previous works, we have proposed a DW development method [12], based on the Unified Modelling Language (UML) and the Unified Process (UP), to properly design all aspects of a DW. More specifically, we have dealt with the modelling of different aspects of a DW by using the UML: MD modelling [13] (i.e. the aim of this paper), modelling of the ETL processes, modelling data mappings between data sources and targets [12], modelling physical aspects of DWs at the conceptual level etc. In this section, we outline our approach of using UML package diagrams for the conceptual modelling of large data warehouses [13], which is the approach in which we based on in this paper for the definition of quality indicators and metrics. Based on our experience in real-world cases, we have developed a set of design guidelines for using UML packages in MD modelling. Our approach proposes the use of UML packages in order to group classes together into higher level units creating different levels of abstraction, and therefore, simplifying the final multidimensional (MD) model. In this way, when modelling complex and large DW systems, the designer is not restricted to use flat UML class diagrams. We refer to [13] for a complete description of all design guidelines we have defined.

In Figure 3, we summarize the three main levels in which we structure the conceptual modelling of large data warehouses. At level 1, we define one package for each different star schema[1] we consider in our design and we call them star package. A dependency between two packages at this level represents that they share at least one dimension or one fact. Then, at level 2 we define one package for each dimension and fact considered in our design and we call them *dimension package* and *fact package*, respectively. There is always a dependency between the fact package and the dimension packages meaning that one fact consists on the corresponding dimensions. A dependency between two dimension packages means that they share at least one classification hierarchy level. Finally, at level 3, we specify the whole content of both dimension and fact packages. As seen in this Figure 3, at level 3, each

[1] Although star schema is a logical schema, we refer to star schema to the abstract definition of one fact and several dimensions.

dimension package will contain the definition of the corresponding dimension and their classification hierarchy levels. We should notice that the dependencies between packages allow us to define one element (package, fact or dimension) just once in our design and then re-utilise it whenever convenient.

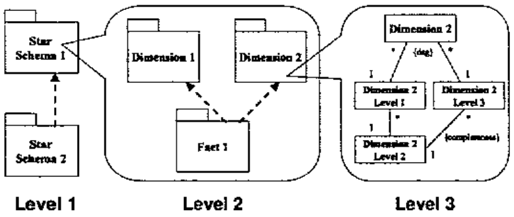

Fig. 3. The three levels of our MD modelling approach with UML package diagrams

Our whole approach for the conceptual modelling of DWs has been specified by means of a UML profile that contains the necessary stereotypes in order to carry out conceptual modelling successfully. Due to space constraints, we refer the reader to [13] for further details on the profile.

4 Quality Indicators and Metrics

Prior to the definition of an indicator we must clearly and precisely know the goal of what we want to measure. The structural properties such as the structural complexity of a schema have an impact on its cognitive complexity [4] and on the mental burden of the persons who have to deal with the artefact. High cognitive complexity leads an artefact to reduce their understandability, analyzability and modifiability. Leading to undesirable external quality attributes [9] [4]. For this reason, it is desirable that a schema has excellent structural properties to be able to achieve good quality. In this paper, our goal will be to minimize the structural complexity of the conceptual schemas to guarantee their quality.

Once the main goal has been set, we have to define the corresponding indicator to measure it. As in this paper, we work with levels 1 and 2 or our package diagram proposal (see Figure 2), we need to define one indicator for each level. If we are able to obtain the minimum structural complexity in both levels, we will therefore obtain the minimum structural complexity in the final conceptual schema.

After having defined the indicators, we must establish the elements we need to measure to further define the corresponding metrics to measure them:

Number of input and output relationships per package.
Number of input and output relationships between two packages
Number of output relationships of a package with regard to the total relationships that exist on the model.

Thus, we will proceed with the definition of the required metrics. These metrics will be applied at level 1 (diagram) and 2 (package) of our approach (Figure 2).

Table 2. Diagram level metrics

Metric	Description
NP(S)	Number of packages of the diagram S
NRES1(S)	Number of input and output relationships of the diagram level
NRESP(S)	Number of input and output relationships between two packages
RESP(S)	Ratio of input and output relationships per number of packages RESP(S) = NERS1(S) / NP(S)

Table 3. Package level metrics

Metric	Description
NRS(P)	Number of output relationships of a package P
RST(P)	Ratio of output relationships of a package P by the total relationships of this package RST(P) = NRS(P) / NRES1(S)

As one of the goals was to obtain the minimum complexity of diagrams at level 2, we define an indicator to measure the structural complexity of diagrams at this level. On defining the indicator, the next step is to know what we need to measure:

Number of output relationships of a dimension package with regard to the total input and output relationships of this package.
Number of input and output relationships between two packages.
Number input and output relationships between two dimension packages by the number of dimension packages that exits.

In tables 4 and 5 we can find the metrics we have defined:

Table 4. Package level metrics

Metric	Description
NREDP(P)	Number of input relationships to a package dimension P
NRSDP(P)	Number of output relationships of a package dimension P
RSDT(P)	Ratio of relationships out of a dimension package P with regard to the total number of input and output relationships to this package RSDT(P) = NRSDP(P) / (NREDP(P) + NRSDP(P))

Table 5. Diagram level metrics

Metric	Description
NIDP(S)	Number of dimension packages imported from another diagrams
NDDP(S)	Number of dimension packages defined in the diagram
NDP(S)	Number total of packages of the diagram S NDP(S) = NIDP(S) + NDDP(S) + 1
NRTDP(S)	Number of input and output relationships between dimension packages
NRESDP(S)	Number of input and output relationships between two dimension packages
RDP(S)	Ratio of input and output relationships between dimension packages by the number of the dimension packages. RDP(S) = NRTDP(S) / (NDP(S) -1)

4.1 Example

In this section we apply the previously-defined metrics to an example. We have applied our package diagram approach to a supply value chain example completely developed in [13]. In Figure 4, we show the level 1 of the model that is composed

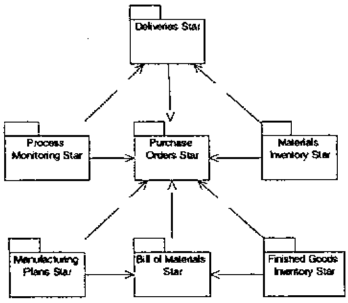

Fig. 4. Level 1: different star schemas of the supply value chain example

Table 6. Level (level 1) package metrics

	NRS	RST
Deliveries	1	1/10
Process Monitoring	2	2/10
Purchase Orders	0	0/10
Materials Inventory	2	2/10
Manufacturing Plans	2	2/10
Bill of Materials	1	1/10
Finished Inventory	2	2/10

Table 7. Metric NERSP[2]

NERSP	Value
Deliveries – Process Monitoring	1
Deliveries – Purchase Orders	1
Deliveries – Materials Inventory	1
Purchase Orders – Materials Inventory	1
Purchase Orders – Process Monitoring	1
Purchase Orders – Manufacturing Plans	1
Purchase Orders –Bill of materials	1
Purchase Orders – Finished Inventory	1
Manufacturing Plans – Bill of materials	1
Bill of materials – Finished inventory	1

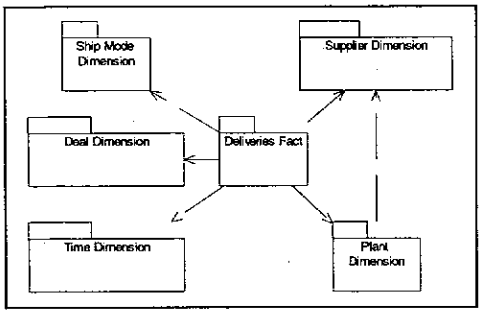

Fig. 5. Level 2: Deliveries Star

[2] Only represent the metrics that value NERSP is different zero.

by seven packages that represent the different star schemas. Then, in Tables 6, 7 and 8 we present the obtained values for the proposed metrics.

In Figure 5, we show the content of the package *Deliveries Star* (level 2). Tables 9, 10 and 11 show the values for the proposed metrics.

The theoretical validation helps us to know when and how apply the metrics. There are two main tendencies in metrics validation: the frameworks based on axiomatic approaches [15] [3] and the ones based on the measurement theory [14][16]. We have validated our metrics by using both frameworks. However, due to space constraints, we cannot provide these theoretical validations in this paper.

Table 8. Level (level 1) diagram metrics

	Value
NP(S)	7
NRES1(S)	10
RESP(S)	10/7

Table 9. Metric NERSP[3]

NERSP	Value
Plant - Supplier	1

Table 10. Level (level 2) package metrics

	NREPD	NRSPD	RSDT
Ship Mode	1	0	0/1
Deal	1	0	0/1
Time	1	0	0/1
Plant	1	1	1/2
Supplier	2	0	0/2

Table 11. Level (level 2) diagram metrics

	Value
NIDP	3
NDDP	2
NDP	6
NRTDP	1
RDP(S)	1/5

6 Conclusions

In this paper we have focused on the quality of the conceptual models of data warehouses. We have mainly focused on those models that use UML packages to model data warehouses. We have proposed a set of quality indicators and the metrics on which they are based on in order to assure the quality of the data warehouses conceptual models. These quality indicators have allowed us to clearly define quantifiable elements in which we based on for measuring the quality of the models. In order to obtain high confidence indicators, we have defined the metrics for each indicator we have defined.

Those metrics have been theoretically validated using two formal frameworks, each of them representing a validating approach: axiomatic approaches and those approaches based on measurement theory. This paper has presented the first steps in obtaining a valid set of quality indicators. We are now focusing on develop the empirical validation with the proposed indicators and metrics in order to obtain a valid and useful set of quality indicators for data warehouse conceptual models.

[3] Only represent the metrics that value NERSP is different zero.

References

1. Abelló, A., Samos, J. and Saltor, F. *YAM2 (Yet Another Multidimensional Model): An extension of UML.* in *International Database Engineering & Applications Symposium (IDEAS'02).* 2002.
2. Bouzeghoub, M. and Kedad, Z., *Quality in Data Warehousing*, in *Information and database quality.* 2002, Kluwer Academic Publishers.
3. Briand, L., Morasca, S. and Basili, V., *Property-Based Software Engineering Measurement.* IEEE Transactions on Software Engineering, 1996. **22(1)**: p. 68-86.
4. Briand, L., Wüst, J. and Lounis, H. *A Comprehensive Investigation of Quality Factors in Object-Oriented Designs: an Industrial Case Study.* in *21st International Conference on Software Engineering.* 1999. Los Angeles, California.
5. Calero, C., Piattini, M. and Genero, M. *Method for obtaining correct metrics.* in *3rd International Conference on Enterprise and Information Systems (ICEIS'2001).* 2001.
6. Cantone, G. and Donzelli, P., *Production and maintenance of software measurement models.* Journal of Software Engineering and Knowledge Engineering, 2000. **5**: p. 605-626.
7. Goethert, W. and Siviy, J., *Applications of the Indicator Template for Measurement and Analysis Initiative, Technical Note CMU/SEI-2004-TN-024.* 2004.
8. Inmon, W. H., *Building the Data Warehouse.* 3rd Edition ed. 2002, USA: John Wiley and Sons.
9. ISO/IEC, *9126-1: Software Engineering - Product quality - Part 1: Quality model.* 2001.
10. Jarke, M., Lenzerini, M., Vassiliou, Y. and Vassiliadis, P., *Fundamentals of Data Warehouses.* second edition ed. 2002: Springer-Verlag.
11. Kimball, R. and Ross, M., *The Data Warehouse Toolkit, 2nd edition.* 2002: John Wiley & Sons.
12. Luján-Mora, S. and Trujillo, J. *A Data Warehouse Engineering Process.* in *3rd International Conference in Advances in Information Systems(ADVIS2004).* 2004.Izmir (Turkey).
13. Luján-Mora, S., Trujillo, J. and Song, I.-Y. *Multidimensional Modeling with UML Package Diagrams.* in *21st International Conference on Conceptual Modeling, LNCS 2503.* 2002.
14. Poels, G. and Dedene, G., *Distance-based software measurement: necessary and sufficient properties for software measures.* Information and Software Technology, 2000. **42(1)**: p. 35-46.
15. Weyuker, E., *Evaluating Software Complexity Measures.* IEEE Transactions on Software Engineering, 1988. **14(9)**: p. 1357-1365.
16. Zuse, H., *A Framework of Software Measurement.* 1998, Berlin: Walter de Gruyter.