CLIQ – Intelligent Data Quality Management

Holger Hinrichs

Oldenburg Research and Development Institute
for Computer Science Tools and Systems (OFFIS),
Escherweg 2, 26121 Oldenburg, Germany
holger.hinrichs@offis.de

Abstract

The primary objective of this work is to develop a concept for a holistic data quality management which is based on formal data quality metrics and a well–defined process model. The extensive use of metadata provides a flexible adaptation to various application domains and a maximum degree of automation.

1 Research Question

The increasing popularity of data warehouses [Inm92] reflects the rising requirement to make strategic use of data integrated from heterogeneous sources. While the research subject of schema integration has been extensively discussed for many years, data integration has been neglected up to the recent past. Data integration often reveals deficiencies of data quality, e. g. inconsistency, redundancy, and incompleteness of data. If data do not suffice given quality requirements, their use may lead to wrong decisions with serious consequences ("garbage in, garbage out"). Consequently, some kind of *data quality management (DQM)* is necessary (see Fig. 1).

Two basic approaches are possible:

- *Reactive DQM:* Before data are released for analysis tasks, they are checked whether they suffice specified requirements. If they do not, data cleansing methods are applied to improve data quality as far as possible.
- *Prospective DQM:* The business processes which "produce" the data (especially data acquisition, transformation, and consolidation processes) are tuned dynamically in such a way that only high quality data are produced.

Of course it is always better to strike at the root of a problem, which means in this case to optimise the business processes. Unfortunately, prospective DQM is not always applicable in practice, since the processes in question are often outside the optimiser's sphere of influence. Legacy systems e. g., which often deliver low quality data, usually cannot be extended by appropriate integrity rules belatedly.

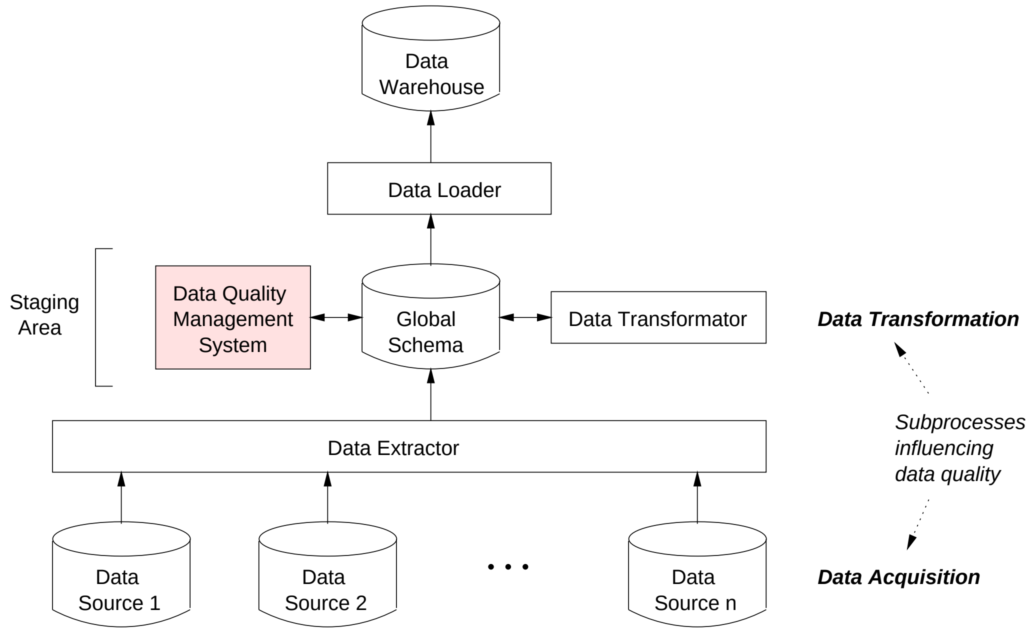

Figure 1: Data integration process in data warehousing

Besides, many data deficiencies cannot be detected until data integration. As a consequence, prospective and reactive DQM have to be employed in combination, complementing each other.

In our opinion, an effective data quality assurance requires a holistic, methodically funded approach that covers the whole spectrum from quality planning via quality measurement and analysis to quality improvement.

2 Significant Research Problems

The importance of data quality for organisational success has been underestimated for a long time. For this reason, quality management in information processing is not nearly as established as it is in manufacturing. There is a serious lack of formally funded methods to measure data quality (even more serious than in the software quality field). Furthermore, there is no well–established process model which defines how to manage data quality.

A software system that realises DQM in an organisation is called a *data quality management system (DQMS)*. Since it cannot be practicable to implement a DQMS from scratch for each and every application domain, there must be means to share and reuse components of a DQMS.

Automation of DQM is another important issue: Since modern data warehouses store gigabytes up to terabytes of data, a manual quality control is not feasible at all. Instead, human interaction should be reduced to cases where conflicts cannot be solved automatically. In some applications, especially in real–time environments, it might be too time–consuming to check each and every data value, necessitating special solutions.

3 Related Work

Contemplating the related work concerning data quality, a clear distinction should be made between approaches originating from practice on the one hand and research activities on the other.

Practice Approaches

In practice, the usual approach has been to execute short–term activities aiming at acute problems, e. g. writing a C routine to eliminate a casually detected inconsistency. The main disadvantages of such a proprietary approach are obvious, namely low reusability and high costs. In the last few years, a large number of so–called ETL (Extraction, Transformation, Loading) tools hit the market [Eng99], claiming to simplify the process of populating a data warehouse significantly. But although some of these tools provide sophisticated graphical interfaces and comprehensive libraries of transformation functions, they are insufficient for DQM for the following reasons:

- Their functionality is usually tailored to simple data migration tasks, like standardisation of addresses and conversion of measures.
- They do not support DQM explicitly, lacking both a data quality model and a process model for DQM.

Nevertheless, an ETL tool can be particularly useful as a single module of an overall DQMS (see Sect. 5).

Research Approaches

In data quality research, there are two well–known projects, namely MIT's *Total Data Quality Management (TDQM)* [Wan98] and the ESPRIT project *Foundations of Data Warehouse Quality (DWQ)* [JJQV98]. These projects – both finished meanwhile – can be characterised as follows:

- *TDQM* claims to establish a theoretical foundation of data quality. Based on the enterprise philosophy of Total Quality Management [Jur99], a so–called TDQM cycle is defined consisting of the four phases quality planning, measurement, analysis and improvement. Additionally, TDQM identifies a set of so–called quality dimensions (e. g. accuracy, completeness, timeliness) and proposes some simple metrics for data quality measurement. Finally, TDQM provides an approach to enrich the relational data model with data quality information by introducing meta relations. As a conclusion, TDQM offers some interesting ideas that form a suitable basis for further data quality research.
- *DWQ* is tailored to the data warehousing world. Not only the quality of warehouse data is considered, but also the quality of the warehouse architecture itself. There is a clear distinction between conceptual, logical, and

physical data models, the semantics of which are explicitly described as metadata. Integration of source schemas is supported by so–called inter-schema knowledge networks which specify relationships between schemas. Although DWQ follows a holistic approach, it disregards the need for a formal foundation of data quality concepts. Consequently, the quite sophisticated meta model cannot be reasonably used in practice since no suitable metrics for data quality measurement have been defined.

Apart from TDQM and DWQ, there are several minor research activities (e. g. [BT99], [Jar89], and [KKPP98]), each concentrating on some special aspects of data quality.

4 Research Methodology

In our work, we concentrate on data conflicts and data integration. We assume that schema integration [SL90] has already been done and that there is one global database schema (called staging area in [Kim98]) where data from different sources can be stored (see Fig. 1). These data possibly still contain data deficiencies like inconsistencies, redundancy, and incompleteness. Before they can be transferred into a data warehouse, they have to undergo a quality control, comparable to quality control in manufacturing.

Our procedure to tackle the research problems described in Sect. 2 consists of the following subtasks:

- *Definition of a formal data quality framework:* Domain dependent and subjective aspects play a prominent role in DQM. For this reason, we believe that it is not possible to set up a fully equipped data quality concept which is immediately applicable in practice. Instead, we decided to develop a framework for DQM that can be instantiated according to domain specific and/or subjective requirements. The framework should provide the following features:
  - It should cover all relevant aspects of DQM in data warehousing environments, forming a holistic model.
  - It should be based on a formal model, offering strong guidelines and minimising ambiguities.
  - It should take the aspect of subjectivity into account, an important characteristic of DQM.
  - It should be adaptable to domain specific aspects in order to realise a DQM within an existing data warehouse system, thus assuring pragmatism and fitness for practice applications.
- *Definition of a process model for DQM:* Based on the framework resulting from the previous task, we define a process model that specifies which measures are to be carried out in which order under which conditions.

- *Design of a metadata model for DQM:* In order to enable interoperability and a high degree of automation, we decided to integrate an extensive metadata support into DQM. In this subtask, the data quality framework is being mapped to an appropriate metadata model, making use of an existing metadata standard.

- *Design of a DQMS:* We identify software modules that are able to realise the just defined process model, specify their internal stuctures and workflows, and set up the flow of data (especially metadata) and control information between them.

- *Prototypical implementation of selected software modules:* Those software modules that cannot be realised by off–the–shelf products are implemented prototypically to demonstrate their basic functionality.

- *Evaluation by means of a real–world application:* To prove the soundness of our concepts and implementation, we apply the DQMS to a real–world application suffering from data quality problems.

According to this research methodology, we will now sketch the current state of our work, which is done in the scope of a research project named CLIQ (Data Cleansing with Intelligent Quality Management):

5 Basic Ideas and Preliminary Results

The following sections describe the current state of the CLIQ project (January 2000).

Definition of a formal data quality framework

The preliminary framework consists of the following components (see Fig. 2):

- A set of *data objects* under consideration. Each data object is defined by a database query (using the `SELECT` statement of SQL).[1] This query–based approach (including the selection, projection, join[2], union, intersection and subtraction facilities of SQL) enables a uniform processing of data objects at different levels of *granularity* (attribute value, partial or complete database record, set of partial or complete database records, and joins of database records from different relations).

- A *domain model* representing the domain to which the framework is to be tailored. It contains the following components:

  - Sets of *business terms* and *business rules* that describe the semantics of the domain.

[1] Preliminarily, the relational data model is supported exclusively.
[2] Not reasonable if the data comprises referential integrity deficiencies.

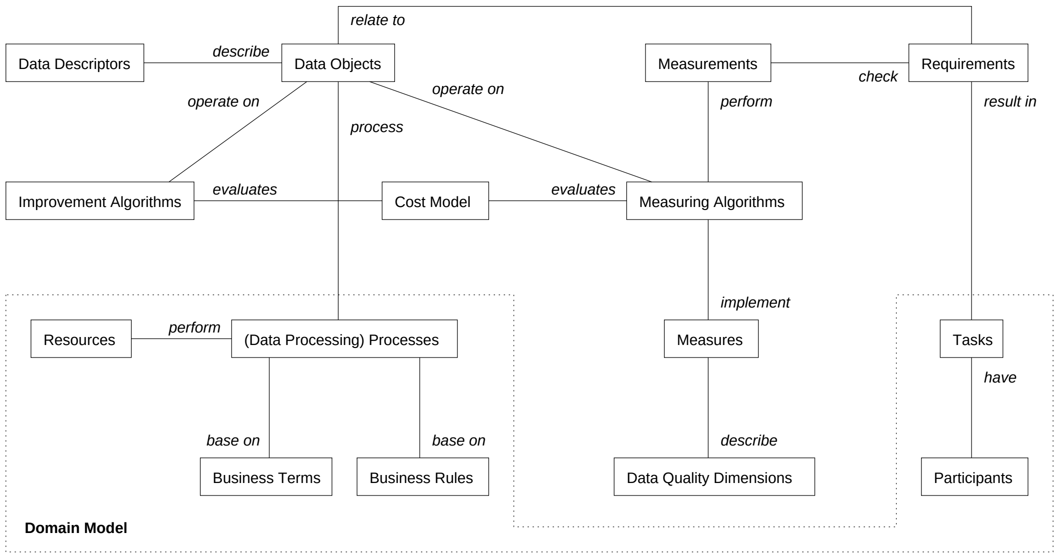

Figure 2: Overview of our data quality framework

- A set of *(data processing) processes* that may influence the quality of processed data objects. Processes can be concatenated to *process chains*. Furthermore, *checkpoints* can be defined at various process steps.
- A set of *resources* (human or software agents) performing processes, e. g. data entry clerks performing the data acquisition process or ETL tools performing the data transformation process within a data warehouse system.
- A set of *participants* (users of data objects in any way), grouped by participant *profiles* (e. g. management, analysts, application developers, or database administrators).
- A set of participant specific *tasks* that are to be carried out on data objects to reach certain *goals*. Tasks and goals represent the subjective influence of participants. They are reflected in quality requirements (see below).

• A set of *data descriptors* describing data objects, including both conventional data dictionary information and statements concerning the semantics of data (metric units etc.).

• A set of *data quality dimensions* which we freely adapted from TDQM (see Sect. 3). Since we consider this set to be canonical, it is predefined in the framework (see Fig. 3).

Data quality dimensions may interrelate. Each relationship is associated with a *type* (part of–relationship or influence) and – in case of influence – with a *weight* which correlates with the direction and strength of the influence. This weight is scaled to an interval $[-1; 1]$ whereby -1 denotes the maximal negative (conflicting) influence, 1 the maximal positive (synergetic) influence,

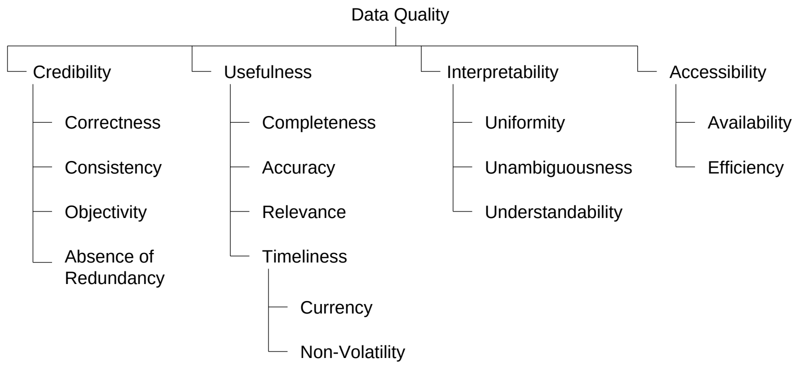

Figure 3: Dimensions of data quality (derived from [Wan98])

and 0 the independence of dimensions. In case of part of–relationships, a weight indicates the share of a part in the whole (interval $[0;1]$). Due to domain dependence, these weights are to be specified during instantiation of the framework.

The data quality dimensions and their interrelations represent our understanding of the concept ”data quality”. It can be formally described by an *empirical relation system* $ERS = (E, R)$ consisting of a set E of *entities* (dimensions, particularly) and a set R of *relations* between entities. The ERS defines a number of statements about data quality that we consider correct and relevant. In order to make data quality measurable, we have to transform the ERS into a *numerical relation system* $NRS = (N, P)$ consisting of a *number system* N (e. g. $\mathbb{N}$ or $\mathbb{R}$) and a set P of *relations* over N [FP97].

- A set of *measures* for data quality dimensions. Each measure M is a mapping from the empirical relation system ERS to the corresponding numerical relation system NRS. Entities in E are mapped to numbers in N, and relations in R are mapped to relations in P. We insist that M preserves all empirical relations in NRS (homomorphism), i. e. that M captures the semantics of data quality appropriately. A measure M is tailored to a (perhaps single element) subset of granularity levels (of data objects).

- A set of participant specific *requirements*, defining nominal values for the quality of certain data objects along certain dimensions (quality planning). Requirements can be *weighted* according to their relative (subjective) importance.

- A set of *measuring algorithms* for data quality measures, including parameter lists and return value types. Measuring algorithms are assigned to one of two categories, dependent on whether they perform a *direct* or *indirect* measurement. Whilst direct measurement of an dimension A involves the dimension A itself, indirect measurement makes use of one or more dimensions $B_1, ..., B_n$ different from A.

- A set of *measurements* resulting from the application of measuring algorithms to data objects.

- A set of *improvement algorithms* for data objects, classified into *prospective* and *reactive* algorithms.

- A *cost model* where (prospective or reactive) DQM activities are associated with costs and benefits. This cost model represents the economic component of quality planning.

The subtask of developing methods for data quality measurement turned out to be quite difficult, since there is very little preparatory work to be found in literature, except for some universally applicable fundamentals of measurement theory [FP97]. For this reason, we decided to examine approaches from various other disciplines, in detail:

- Quality of conceptual data models [KLS95] [MS94]
- Software quality [FP97]
- Quality of networking services [Sti96]
- Facets of interestingness in data mining [Mue99]
- Quality management in manufacturing [Jur99]

For each approach, we extract the ideas and concepts which we consider relevant for our context, adapt them appropriately, extend them by data quality specific aspects and integrate all these into a holistic, formal model that meets our requirements. This task has not been finished yet.

Definition of a process model for DQM

Some basic decisions concerning this task, which is currently in process, have already been made: Similar to TDQM (see Sect. 3), we adapt a cyclic model from the manufacturing domain. Prospective and reactive DQM are explicitly supported by two dedicated submodels of the overall process model.

In detail, prospective DQM is assisted by a process cycle which we derived from statistical process control (SPC), a technique well–established in manufacturing for several decades. The idea of our SPC derivative is to draw samples of data, check these samples, and hence draw conclusions about the entire data set using statistical methods [Hin00].

Reactive DQM is supported, apart from conventional data scrubbing methods, by a newly developed process model for *data auditing* [HW00] which was derived from the knowledge discovery in databases (KDD) field [Fay96]. Data auditing makes use of data mining methods (e. g. decision trees, neural networks, and rule induction) in order to detect possible data inconsistencies and predict missing values.

Design of a metadata model for DQM

In [Hin99], we defined a classification of metadata for DQM, comprising so–called data descriptors (e. g. specifications of data acquisition characteristics), domain knowledge (e. g. business rules and domain specific ontologies), and DQM specific information (e. g. quality plans, measurement results, and process logs). These types of metadata have already been taken into account in framework design. Consequently, this task is reduced to mapping the data quality framework onto appropriate metadata structures.

Framework Element	OIM Submodel	OIM Model
Data Descriptors	Database Schema Model	Database & Warehousing
Improvement Algorithms	Data Transformations Model	Database & Warehousing
Participants (Domain Model)	Organisational Elements	Business Engineering
Resources (Domain Model)	Organisational Elements	Business Engineering
Tasks/Goals (Domain Model)	Business Goals	Business Engineering
(Data Processing) Processes (Domain Model)	Business Processes	Business Engineering
Business Rules (Domain Model)	Business Rules	Business Engineering
Business Terms (Domain Model)	Knowledge Descriptions	Knowledge Management

Table 1: Correspondences between OIM and our framework

We decided to rest our metadata model upon some broadly agreed standard in order to minimise the development effort and to maximise interoperability. An evaluation of several standards (CWMI, MDAPI, MDIS, OIM, RDF [DBTG00]) yielded that the Open Information Model (OIM) [MDC00] is the one which best suits our requirements for the following reasons:

- For many basic concepts of our framework there are corresponding OIM data structures (see Tab. 1).
- OIM is easily extensible by defining new classes or extending predefined ones.
- OIM is being supported by major software vendors and other companies, e. g. Microsoft, IBM, Informix, and SAS.
- A free implementation of OIM is available (ships with Microsoft SQL Server 7, for example).

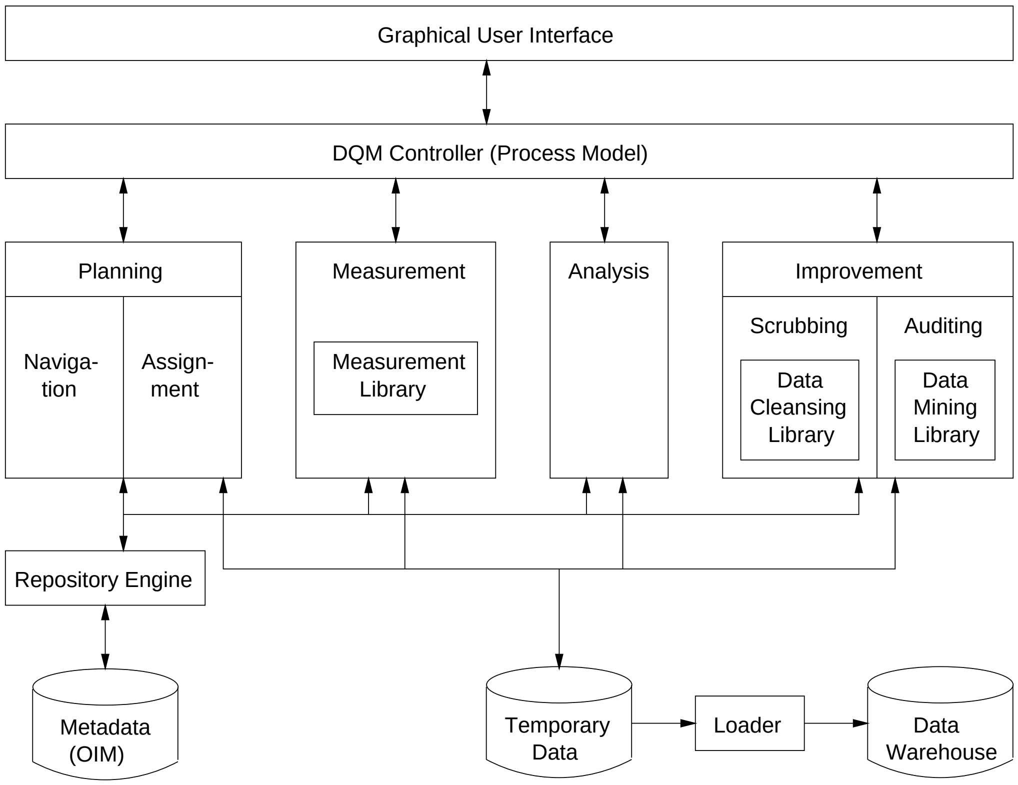

Figure 4: DQMS Architecture

We currently extend the information models of OIM by the DQM specific aspects of our data quality framework.

Design of a DQMS

The architecture of the DQMS being developed within CLIQ is depicted in Fig. 4. The data whose quality is to be assured are temporarily stored at a special location which can be viewed as the data staging area of a data warehouse system [Kim98]. A loader is responsible for transferring quality assured data from the temporary store to a data warehouse.

The single phases of the DQM process are reflected by corresponding software modules. The cooperation of these modules within the process model is managed by a DQM controller interacting with the user via a graphical user interface. Each quality management module accesses metadata by means of a central repository.

The DQMS provides a high degree of automation. Manual interferences are reduced to system configuration, specification of quality requirements, and solving data conflicts that cannot be handled automatically.

Prototypical implementation of selected software modules

Up to now, a module for user specific data quality planning and measuring (the measuring methods will be plugged in later) and a data auditing system have been implemented [Sac99] [HW00]. On the part of commercial software tools, the Microsoft Repository [Ber97], the rule processor Ilog Rules [Ilo00], and the data mining class library MLC++ [KSD96] are being integrated into the DQMS. Furthermore, the ETL tool Integrity [Val00] with its data migration and scrubbing algorithms is to be integrated as well.

Evaluation by means of a real–world application

Both the concepts of the DQMS and their implementation will be evaluated by means of a real–world application, namely the epidemiological cancer registry of Lower–Saxony. Like many other organisations, this cancer registry has to cope with serious data quality problems.

6 Rating

In the scope of CLIQ we are developing solutions to problems and challenges related to data quality issues. The primary objective of CLIQ is to design and implement a software tool for data quality management based on a formal data quality framework and a well–defined process model. These can be tailored to various application domains in a flexible way. The extensive use of metadata provides interoperability and a maximum degree of automation of the quality management process. We believe that this formal and concurrently holistic approach represents a novelty in current data quality research.

Among the related topics not treated within the scope of CLIQ are schema integration and reengineering of metadata.

References

[Ber97] Bernstein, P. A. et al.: The Microsoft Repository. *Proc. of the 23rd Intl. Conf. VLDB, Athens, Greece*, 1997.

[BT99] Ballou, D. P., Tayi, G. K.: Enhancing Data Quality in Data Warehouse Environments. *Communications of the ACM*, **42** (1): 73–78, 1999.

[DBTG00] Database Technology Research Group of Zurich University: *http:// www.ifi.unizh.ch/ dbtg/Projects/SMART*, 2000.

[Eng99] English, L. P.: *Improving Data Warehouse and Business Information Quality*. Wiley, New York, 1999.

[Fay96] Fayyad, U. M.: Data Mining and Knowledge Discovery: Making Sense out of Data. *IEEE*, **11** (5), 1996.

[FP97] Fenton, N. E., Pfleeger, S. L.: *Software Metrics: A Rigorous and Practical Approach*. 2nd Ed., Intl. Thomson Computer Press, London, 1997.

[Hin99] Hinrichs, H.: Metadata–based Quality Management of Warehouse Data. *Proc. of the 19th Conf. DATASEM'99, Brno, Czech Republic*, pp. 239–248, Masaryk University, Brno, 1999.

[Hin00] Hinrichs, H.: Statistical Quality Control of Warehouse Data. *Proc. of the 4th IEEE Intl. Baltic Workshop on DB and IS, Vilnius, Lithuania*, 2000.

[HW00] Hinrichs, H., Wilkens, T.: Metadata–Based Data Auditing. *Proc. of the 2nd Intl. Conf. Data Mining 2000, Cambridge, UK* (to appear), WIT Press, Southampton, 2000.

[Ilo00] Ilog: *http://www.ilog.com/products/rules*, 2000.

[Inm92] Inmon, W. H.: *Building the Data Warehouse*. Wiley, New York, 1992.

[Jar89] Jaro, M. A.: Advances in Record Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. *Journal of the American Statistical Association*, **84**: 414–420, 1989.

[JJQV98] Jarke, M., Jeusfeld, M. A., Quix, C., Vassiliadis, P.: Architecture and Quality in Data Warehouses. *Proc. of the 10th Intl. Conf. CAiSE*98, Pisa, Italy*, pp. 93–113, Springer, Berlin, 1998.

[Jur99] Juran, J. M. (Ed.): *Juran's Quality Handbook*. 5th Ed., McGraw–Hill, 1999.

[Kim98] Kimball, R.: *The Data Warehouse Lifecycle Toolkit*. Wiley, New York, 1998.

[KKPP98] Kaplan, D., Krishnan, R., Padman, R., Peters, J.: Assessing Data Quality in Accounting Information Systems. *Communications of the ACM*, **41** (2): 72–78, 1998.

[KLS95] Krogstie, J., Lindland, O. I., Sindre, G.: Towards a Deeper Understanding of Quality in Requirements Engineering. *Proc. of the 7th Intl. Conf. CAiSE*95*, Springer, 1995.

[KSD96] Kohavi, R., Sommerfield, D., Dougherty, J.: Data Mining using MLC++ – A Machine Learning Library in C++. *Tools with AI 1996*, pp. 234–245, 1996.

[MDC00] Meta Data Coalition: *http://www.MDCinfo.com*, 2000.

[MS94] Moody, D. L., Shanks, G. G.: What Makes a Good Data Model? Evaluating the Quality of Entity Relationship Models. *Proc. of the Intl. Conf. ER'94, Manchester*, LNCS 881, Springer, 1994.

[Mue99] Mueller, M.: Interessantheit bei der Entdeckung von Wissen in Datenbanken (in German). *Kuenstliche Intelligenz*, 3/99, pp. 40–42, arenDTaP, Bremen, 1999.

[Sac99] Sachtleber, M.: *Eine generische Bibliothek von Datenqualitaetsmessverfahren fuer Data Warehouses* (in German). Diploma Thesis, University of Oldenburg, Germany, 1999.

[She31] Shewhart, W. A.: *Economic Control of Quality of Manufactured Product*. D. Van Nostrand, New York, 1931.

[SL90] Sheth, A. P., Larson, J. A.: Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, **22** (3): 183–236, 1990.

[Sti96] Stiller, B.: *Quality of Service: Dienstguete in Hochleistungsnetzen* (in German). Intl. Thomson Publishing, Bonn, 1996.

[Val00] Vality Technology Inc.: *http://www.vality.com*, 2000.

[Wan98] Wang, R. Y.: Total Data Quality Management. *Communications of the ACM*, **41** (2): 58–65, 1998.