Multi-Source Model and Architecture for Quality Negotiation and Integration of Biological Data

Laure Berti-Équille

Multi-Source Model and Architecture for Quality Negotiation and Integration of Biological Data

Laure Berti-Equille
IRISA, Campus Universitaire de Beaulieu,
35042 Rennes cedex, France

20th April 2001

1 Introduction

Maintaining a certain level of quality of data and data sources is challenging in distributed multiple source environment. In practice, assessing data quality in database systems is mainly conducted by professional assessors with more and more cost-competitive auditing practices. Well-known approaches from industrial quality management and software quality assessment have been adapted for data quality and came up with an extension of metadata management [23, 14, 30, 32, 29]. Classically, the database literature refers to data quality management as ensuring : 1) syntactic correctness (e.g. constraints enforcement, that prevent "garbage data" from being entered into the database) and 2) semantic correctness (i.e. data in the database truthfully reflect the real world situation). This traditional approach of data quality management has lead to techniques such as integrity constraints, concurrency control and schema integration for distributed and heterogeneous systems. Techniques such as data tracking, data cleaning and data quality controlling are costly in practice and time-consuming to adapt and configure efficiently in a specific application domain. Although critical to data quality for a single database, these techniques fail to address some issues that are important in a distributed and heterogeneous environment. As recent studies show, applications built on top of data warehouses often experience several problems due to the reliability and the quality of integrated data (see Chapter 7 in [13]). The main reason is that the local databases participating in providing their data contain incorrect, inaccurate, outdated or poor quality data. The quality of integrated data then becomes even worse unless suitable methods and techniques are employed during the multi-source environment design. Despite the amount of work focuses on semantic heterogeneity among data and metadata [12], the quality of integrated data has not been addressed so far. Only few research projects addressed the issues of multi-source data quality control and the management of enriched metadata in which specific analysis and optimization techniques are embedded [7, 18, 3]. Data quality mainly has been and still is an important research topic independent of data base integration. Another body of research literature recently focuses on improving aspects of quality of service in information and database systems [1, 24], or on the particular problems of trade-off between freshness of data, system performance and precision by minimizing data transfer through incremental view maintenance [19].

In this extended abstract, we focus on data quality in a multi-source environment. We claim that data conflicts often occur due to the difference of local data quality and its variation over time. Considering quality dimensions implies that there is not a unique resolution of data conflicts for integration, several solutions are possible and should be transparent to users : for

example, if the values of two objects referring to the same real world concept differ and if one value is known to be more accurate and the other one is known to be more up-to-date, a conflict needs to be solved considering aspects of data accuracy versus data freshness. Note that we assume that accurate data does not necessarily imply up-to-date data and vice-versa. So, for the resolution of this conflict, a unique data integration rule is not sufficient and not flexible enough. Therefore, in the context of a standard wrapper-mediator architecture, we suggest to differ data conflict conciliation using quality requirements for the query results. Rather than data integration, we propose data recommendation based on data and source quality. Quality dimensions are formalized by a quality contract which is negotiated by the mediator. We propose a negotiation algorithm included into the query and mediation processing. The negotiation allows to customize multi-source integration according to end-users' requirements. This technique embeds implicitly quality control into the multi-source data extraction. Our application context is biological databanks.

2 Related Works

Classically, the methods and techniques described in this paper are influenced by two main areas : data cleaning and data quality as metadata management. The aim of this section is to confront the research propositions of these areas to some specificities of the biological databanks.

Mediation and cleaning of multi-source data

Data cleaning objectives are to detect matching records from several input extensional data structures (relational tables, object classes, DTDs), to find out and eliminate duplicate records in the integration process. As mentioned in [6], the main drawback of data cleaning methods is that, besides being a knowledge and time intensive task (since it implies several passes on the data), finding the suitable key for putting together similar records is very difficult. As another drawback, the support for cleaning rules offered in [17] allows matching rules to be applied only to pairs of neighbour records in the same file. Although, the problem of the possible conflict in data values was recognized, few specific solutions were offered [26, 16]. Another problem arising from the proliferation of independent sources, some of them with overlapping information, is the inconsistency of information content, and hence there is a need for methods to resolve such inconsistencies in global answers. Inconsistencies result in multiple candidate answers; the dual problem also exists, that is a global query might have no answer at all. The usual approaches to reconciling the heterogeneities in data values are (1) to prefer the values from a more reliable database, (2) attach tags with data source identifications to data items and rely on the reputation of the data source [33], (3) store reliability measures of data sources from which a data item originated along with the data item itself [26]. These approaches also suffer from several drawbacks. First, it is not clear how to determine which of the sources is more reliable and how to measure reliability of a source. Second, even if the reliability of the data sources is somehow provided, it is implicitly assumed that the reliability remains the same for all data items from a particular data source. However, the reliability of the data items may be significantly different in the different parts of the source. And third, storing the reliability information or the data source tags along with the data items requires significant modifications in the conventional query processing mechanisms and increases data storage requirements.

Multi-source data quality and meta-data management

There are a number of research investigating issues related to models and methodologies for data quality improvement [34, 32, 29], specifications and metrics for data quality dimensions [14]. Currently, data quality audits are the only practical means for determining quality of data in databases by using the appropriate statistical techniques. Since databases model a portion of the

real world which constantly evolves, the data quality estimates become outdated as time passes. Therefore, the estimation process should be repeated periodically depending on the dynamics of the real world. The statistical aspects of data quality have been the primary focus with statistical methods of imputation (i.e., inferring missing data from statistical patterns of available data), predicting accuracy of the estimates based on the given data, data edits (automating detection and handling of outliers in data). The use of machine learning techniques for data validation and correction is considered in [27]. For example, [28] describes a prototype system for checking correctness of the existing data (called data validation and cleanup). Utilization of statistical techniques for improving correctness of databases and introduction of a new kind of integrity constraints were proposed in [11]. The constraints are derived from a database instance using the conventional statistical techniques (e.g., sampling and regression), and every update of the database is validated against these constraints. If an update does not comply with them, then a user is alerted and prompted to check correctness of the update. Despite the fact that there is much research growing the importance impact of data quality for end-users and that many techniques have been proposed to improve and maintain quality of local databases, very few projects try to use quality metadata for multivalued attributes in distributed and quality-heterogeneous environment (DWQ [3], HiQiQ [18], [7]). The use of metadata for data quality evaluation and improvement was advocated in [25]. The author argued that information producers should perform Verification, Validation, and Certification (VV&C) of their data. The data quality metadata should also be supplied along with a database. The metadata help in the process of estimating and maintaining the quality of data. A great amount of effort has been invested in the development of metadata standard vocabularies for the exchange of information across different applications domains (such as geographic information systems [8] or digital libraries (Dublin Core [35]). We are not aware of any kind of project that tries to specify metadata for biological data and, furthermore, data quality has not been addressed so far in the biological domain and in the current standardization works.

3 Example of Biological Information Retrieval from Distributed Databanks

Searching across distributed, disparate biological databases is increasingly difficult and time-consuming for biomedical researchers. Bioinformatics is coming to the forefront to address the problem of drawing effectively and efficiently information from a growing collection of 511 multiple and distributed databanks [1].). For example, suppose a biological researcher working with a protein and wanting to know what are the protein functions, its cellular location, what gene it is and its DNA sequence, whether the rRNA sequence is known, how the gene is transcribed into mRNA, how the mRNA is translated into that protein... With the currently available biological banks and tools the researcher has to search the relevant databases one by one and then to locate the information items of interest within the return results.

Example 1

Our mediator is designed to provide information about proteins. A protein (Pr) related to a species (Sp) has a sequence (Sq) with an identification number (Id) and a set of bibliographic references (Rf). The mediator of our example will query three existing sources S_1 (EMBL[2]), S_2

[1] For more details, see the Public Catalog of Databases : DBcat, http://www.infobiogen.fr/services/dbcat

[2] EMBL (European Molecular Biology Laboratory), http://www.ebi.ac.uk/embl/

(GeneCards[3]) and S_3 (SWISS-PROT[4]). The raw values are automatically extracted by scripts using the DTD of sources. In order to define an appropriate data integration rules for these sources, obviously data conflicts have to be resolved due to the different values recorded for the same concept. Traditional data integration approaches suggest a conflict resolution method that either chooses one value over the other (or that computes the average in case of the numerical values). A global query for the protein sequence would retrieve one data value from the source according to the specified data integration rule. Note that the same source (S_1) may propose several values for the same concept. Now assume the following scenarii :

Scenario 1 : S_1 updates their data every night, S_2 monthly and S_3 weekly. In this case, the global query time may determine from which data source the most up-to-date data are retrieved.

Scenario 2 : S_3 is the server of an institute which does its own sequencing for human species. Sequence data are highly accurate for this species. S_2 data may have sometimes parsing errors and may come from other sites.

Scenario 3 : S_1 and S_2 cover more biological domains than S_3. S_1 is one of the main genetic databanks and S_3 is one of the main databanks on proteins. Information items of S_1 are usually less complete and accurate than those of S_3 for the protein domain.

The above scenarii briefly show that the way of how and when data is populated into the local sources plays an important role for integrating local data for global queries. They also describe source dependencies and data quality dimensions such as freshness, accuracy, completeness or coverage. Up-date data does neither imply most accurate data nor most complete data. Actually, a global user might be interested in most complete data, and another one might be rather interested in most accurate data. In both cases, it should be possible for these users to specify tolerance thresholds for data and source quality (or, at least, to give technical means to estimate the quality of query results from the different sources). We propose quality contract specification for enhancing and using dynamically source and data quality requirements in the query processing. This technique enables to differ data integration to end-user according to flexible quality criteria. Our approach is a standard mediator-wrapper architecture including new functionalities such as quality source contract negotiation included into mediation. The negotiation algorithm we propose selects the multi-source data as query results considering the quality of sources. Inter-linked objects are detected before combining the results. A implicit assumption underlying this research is that we incorporate a set of controls on the top of data sources to enhance the global system's reliability and its data quality ; the final aim is to maintain a high probability of preventing, detecting and eliminating data non-quality for data integration in an application domain where data quality has not been addressed.

4 A Description of the Multi-Source Architecture

We argue that database techniques such as having an expressive internal data model and query language, together with a meta-information repository and meta-information analysis techniques constitutes a necessary foundation for a mediator system. In order to alleviate the problem of information overload and confusion when results of a query are presented, the classical solution is to rank the results according to consistent relevance assessments. Our approach is to include quality specifications for sources and send them with the query. Such a functionality for retrieval and integration of information must be supported by an easily extensible, scalable and customizable architecture for addressing a wide range of specific applications such as the biological data

[3]GeneCards, http://bioinformatics.weizmann.ac.il/cards/

[4]SWISS-PROT, http://www.expasy.ch/sprot/sprot-top.html

domain. In this perspective, we propose a multi-source architecture (see Figure 1). From the application layer, the user can submit a global query to the mediator which conjointly sends the query and a quality contract type to the sources' wrappers. The wrappers send the corresponding local query to their respective source. Information sources may be cross-referenced, structured or not and with or without a meta-information repository. They respond to their wrapper with the query result and a contract instance. At the mediation layer, the mediator computes 1) a conformance score for each source corresponding to the constraint satisfaction with respect to the contract specification, and 2) a conflict score for the query result. The mediator negotiates with the best sources and combines the query results for data recommendation, and sends it back to the user.

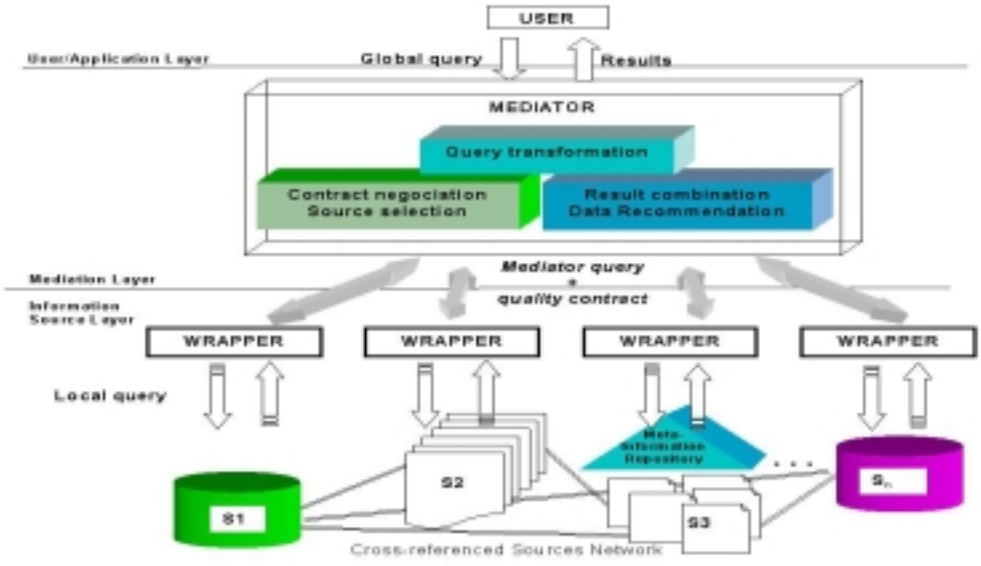

Figure 1: Multi-source Architecture

5 The Multi-Source Data Model

The main features of the multi-source data model are summarized by a reference model, which will be used in the rest of the paper for the discussion. The reference model should not be interpreted as a new model, rather it is similar to the *core model* Object Exchange Model OEM described for the TSIMMIS project [2]. We distinguish the mono-source objects level and the multi-source objects level. A concept of the real world is described by n mono-source objects extracted from n sources and also by one multi-source object.

Definition 1: A mono-source object

A mono-source object is a quintuple $< oid, label, type, value, sid >$ *where oid is the mono-source object's identifier, label is a character string, type is either complex or some identifier denoting an atomic type (like integer, string, gif-image...) ; when type is atomic and its value is an atomic value of that type. Otherwise, the type is complex and its the value is a set of oids (identifiers of so called subobjects). sid is the data source's identifier.*

A mono-source object may be considered as a tree structure and we define that it has the first

depth level (noted $level(o) = 1$).
A source S contains a set of mono-source objects O_s composed with objects o_i, ... , o. S accepts queries over O_s. A query over S simply specifies values for some of the objects of O_s. Thus, a query Q is an assignment of values v_1, ... ,v to the objects o_i, ... , o of O_s.

A subobject o' has the same structure than its root object o, but its depth level is $level(o') = level(o) + 1$.

Definition 3 : A multi-source object
A multi-source object is a quintuple $< ofid, list(label), type, value, list(sid) >$ where $ofid$ is the multi-source object's identifier. Its value is composed with a set of pairs $(value, sid)$ extracted from the mono-source level which describe the same concept of the real world.
The multi-source object has a list of labels considered as synonyms (concatenation of the different labels of mono-source objects referring the same real object). The value of a multi-source object is composed by aggregating several values of mono-sources objects under the conditions that : i) those values refer to the same concept of the real world ii) the data conflicts raised between more or less contradictory values are negligible (i.e. under a given threshold) and still allow data aggregation into the value of the multi-source object. The list of sources' identifiers is also mentioned in the multi-source object.

Example 2
The multi-source object aggregating all the available values for a query asking for the species (**Sp**) and the sequence (**Sq**) of YWHAZ protein is built as follows : In order to provide

```
<of,["YWHAZ"],set(ofid),{of1,of2,of3},[S1, S2, S3]>          * list of labels
     <of1,["DE","GeneCard for","Protein name"],set((string,sid)),
          {("Human phospholipase A2 mRNA, complete cds",S1),
            ("YWHAZ ",S2),("14-3-3 PROTEIN ZETA/DELTA",S3)},
          [S1, S2, S3]>                                    * list of source's IDs
     <of2,["OS","FT /organism=","from"],set((string,sid)),
          {("Homo Sapiens(Human)",S1),("Homo Sapiens",S1),  * 4 pairs (value, sid)
           ("Homo Sapiens(Human);Bos taurus(Bovine)",S3)},
          [S1,S1,S3]>
     <of3,["FT /translation=","SQ Sequence","Representative Sequence","Sequence information:"],
          set((string,sid)),
          {("MDKNELVQK...",S1),
           ("2834 BP; 831 A; 541 C; 550 G; 912 T; 0 other; gcccactccc accgccag...",S1),
           ("M86400 Sequence Information,Sequences GenBank/EMBL/DDBJ,RefSeq, assembly ",S2),
           ("Length: 245 AA MDKNELVQK...",S3)},
          [S1,S1,S2,S3]>
```

adequate level of quality, the mediator needs to include capabilities such as negotiation, monitoring and adaptation. These capabilities all require the expected and the provided quality levels to be explicitly specified. Quality dimensions can be specified statically at the time of source integration or dynamically at deployment or runtime. We characterize quality of data and quality of source along named dimensions (mentioned non-exhaustively in Table 1). Specifying abstractly quality dimensions with a name and a domain value gives a flexible approach for deciding which dimension should be provided and implemented for a given application. The transparency of user's global query processing is weakened by the fact that the user can specify quality contracts for the query results. Based on the matching of user's quality requirements and the quality for each local source, negotiation and mediation are generated dynamically by the global query processor of the mediator to ensure the retrieval of high quality data from the multiple local sources.

For our particular application, these quality dimensions may be specified in a contract type which represents quality dimensions, specifies the name, the domain and possibly user-defined

Quality Contract Type	Definition
Availability	Time and way the source is accessible based on technical equipment and statistics
Freshness	how up-to-date the information is
Accessibility	Estimation of waiting time for user seaching time and for request/response processing (including the time consumption per-query of the wrapper for translating, negociating ...)
Security	Estimation of the number of corrupted data
Coverage	Estimation of the number of data for a specific information domain
Accuracy	Estimation of the number of data free-of-error
Completeness	Estimation of the number of missing data or null values
User satisfaction	User grade based on presentation of data results and ease of understanding and using

Table 1: General Definitions of Quality Contract Types

ordering for each dimension. We can specify examples of source contract types such as : Availability, Freshness, Coverage and Completeness. Figure 2 also presents the instance of contract type [6]. A contract is an instance of a contract type that represents a set of quality dimensions specifications for the source. A contract aggregates a number of constraints. Conformance corresponds to constraint satisfaction (as a boolean function noted CS). We define a conformance score for each source S_j as a weighted function F on the i specified contract types and their k dimensions.

$\forall S_j, contractType_i, dimensionName_k,$

$Conformance(S_j) = F(w_i, w_k, CS(ContractType_i, dimensionName_k, S_j))$

Each contract type may has also particular weights w_i for computing the conformance score of each source S_j (e.g. the contract type on `Freshness` is more important than the one on `Completeness`). Each dimensions of a contract type has particular weights w_k indicating the relative importance of the kth dimension for the contract (e.g. `dataAge` is more important than `updateFrequency`). The conformance scores range from 0 to 1.

```
type Availability = contract {
serverFailure : enum{halt,initialState,rolledBack};
numberOfFailures :decreasing number failures/month;
reliability : increasing number;};
type Freshness = contract {
dataAge : number year,month,day;
lastUpdate : number day(s);
UpdateFrequency : number updates/month;};
type Completeness = contract {
NbOfObject : increasing number/QueryElement;};
```

```
S1_Availability = Availability contract {
serverFailure == initialState;
numberOfFailures <= 0.2 failures/month;
reliability == 0.999; };
S1_ Freshness = Freshness contract {
dataAge == 8 years, 11 months, 3 days;
lastUpdate == 52 days;
updateFrequency == 25 updates/day;};
S1_Completeness = Completeness contract {
NbOfObject : 3/queryElement;}
```

Figure 2: Example of quality source contract types and instances

Each object o in the result for query Q is ranked according to a conformance score $Conformance(S, Q, o)$ of the source S and a conflict score $Conflict(S, Q, o)$ which are computed by the mediator. Query results are then ranked according to $Conformance(S, Q, o)$ and $Conflict(S, Q, o)$ scores

Definition 5

The conflict score $Conflict(S, Q, o)$ of source S for query Q corresponds to the distance between o the object of S and q the queried object of Q. The distance $Dis(o, q)$ is the sum of data conflict importances defined as follows :

[6] The values used to compute the conformance score are extracted from the current release of EMBL, Genecards and SWISS-PROT available on-line

$Dis(o,q) = \sum_{j=1}^{n} I_j$ with I_j the jth data conflict importance such as :

$$I_j = \begin{cases} 0 & \text{if there is no conflict} \\ 0.1 & \text{if the conflict is weak} \\ 1 & \text{if the conflict is strong} \end{cases}$$

The values of conflict score range from 0 to 1.

6 Conclusion

In order to facilitate the multi-source data integration in the context of distributed biological databanks, we propose a technique based on the notions of quality contract and source negotiation. Our approach is based on a standard wrapper-mediator architecture. A quality contract with a source allows to specify quality dimensions necessary to the mediator for data selection between several distributed applications. The selectivity of data sources is dynamically computed by the algorithm of contract negotiation we propose to associate with the global query processing and before data acquisition. Our article proposes to evaluate the importance of data conflicts and to use the quality conformance of sources : the integration of the data is carried out according to the quality of the data required by users. The approach is original and flexible with respect to data mediation and conciliation because it includes source quality of service specifications into the query processing. From the biological application point-of-view, we first introduce the notion of multi-source data quality and data recommendation for biologists and our objective is that the current standardization efforts take into account the quality of biological data and promote operational techniques and tools to evaluate and improve it.

References

[1] A. Bouch, A. Kuchinsky, N. Bhatti. Quality is in the eye of the beholder: meeting users' requirements for Internet quality of service. In *Proc. of CHI'2000*, pages 297–304, 2000.

[2] S. Chawathe, H. Garcia-Molina, J. Hammer et al. The TSIMMIS project: Integration of heterogeneous information sources. *IPSJ*, pages 7–18, October 1994.

[3] D. Clavanese, G. De Giacomo, M. Lenzerini et al. Data integration in datawarehousing. Technical report, DWQ-UNIROMA-001, 1997.

[4] K. Claypool and E. Rundensteiner. Flexible database transformations : the SERF approach. *IEEE Data Engineering Bulletin*, 22(1):19–24, March 1999.

[5] W. Cohen. Integration of heterogeneous databases without common domains using queries based on textual similarity. In *Proc. of ACM SIGMOD Conf. on Management of Data*, 1998.

[6] H. Galhardas, D. Florescu, D. Shasha, E. Simon, C. Saita. Declarative data cleaning : Language, model, and algorithms. Technical Report RR-4149, INRIA, 2001.

[7] M. Gertz and I. Schmitt. Data integration techniques based on data quality aspects. In *Proc. of FDBS Workshop*, pages 1–19, 1998.

[8] M. Goodchild and R. Jeansoulin. *Data quality in geographic information : from error to uncertainty*. Hermès, 1998.

[9] O. Günther and A. Voisard. *Metadata in geographic and environmental data management*, pages 1–31. McGraw Hill, 1997.

[10] M. Hernandez and S. Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *J. of Data Mining and Knowledge Discovery*, 2(1):9–37, 1998.

[11] W. Hou and Z. Zhang. Enhancing database correctness : a statistical approach. In *Proc. of ACM SIGMOD Conf. on Management of Data*, 1995.

[12] R. Hull. Managing semantic heterogeneity in databases: a theoretical prospective. In *Proc. of PODS'97*, pages 51–61, 1997.

[13] M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis. *Fundamentals of Data Warehouses.* Springer, 1998. ISBN 3-540-65365-1.

[14] S. H. Kan. *Metrics and models in software quality engineering.* Addison-Wesley, 1995. ISBN 0-201-63339-6.

[15] V. Kashyap and A. Sheth. So far (schematically) yet so near (semantically). In *Proc. of the IFIP Database Semantics Conference on Interoperable Database Systems*, 1992.

[16] E.P. Lim, J. Srivastava, S. Shekhar. Resolving attribute incompatibility in database integration : An evidential reasoning approach. In *Proc. of the 10th Intl. Conference on Data Engineering (ICDE'94)*, 1994.

[17] A. Monge and C. Elkan. An efficient domain-independent algorithm for detecting approximately duplicate database records. In *Workshop on Research Issues on Data Mining and Knowledge Discovery*, 1997.

[18] F. Naumann and U. Leser. Quality-driven integration of heterogeneous information systems. In *Proc. of VLDB'99*, pages 447–458, 1999.

[19] C. Olston and J. Widom. Offering a precision-performance tradeoff for aggregation queries over replicated data. In *Proc. of VLDB'00*, pages 144–155, 2000.

[20] Y. Papakonstantinou, S. Abiteboul, H. Garcia-Molina. Object fusion in mediator systems. In *Proc. of VLDB'96*, 1996.

[21] Y. Papakonstantinou, H. Garcia-Molina, J. Ullman. MEDMAKER: A mediation system based on declarative specifications. In *Proc. of ICDE'96*, pages 132–141, 1996.

[22] M. Reddy and R. Wang. Estimating data accuracy in a federated database environment. In *Proc. of CISMOD'95*, pages 115–134, 1995.

[23] T.C. Redman. *Data quality for the information age.* Artech House, 1996. ISBN 0-89006-8836.

[24] C. Rolker and R. Kramer. Quality of service transferred to information retrieval: The adaptive information retrieval system. In *Proc. of CIKM'99*, pages 399–404, 1999.

[25] J. Rothenberg. Metadata to support data quality and longevity. In *Proc. of IEEE Metadata Conf.*, 1996.

[26] F. Sadri. Reliability of answers to queries in relational databases. *IEEE TKDE*, 3(2):245–252, 1991.

[27] J. Schlimmer. Learning determinations and checking databases. In *Proc. of the AAAI-91 Workshop on KDD*, 1991.

[28] A. Sheth, C. Wood, V. Kashyap. Q-data : Using deductive database technology to improve data quality. In *Proc. of ILPS'93*, pages 23–56, 1993.

[29] D. Strong, Y. Lee, R. Wang. Data quality in context. *Comm. of the ACM*, 40(5):103–110, 1997.

[30] G. Tayi and D. Ballou. Examining data quality. *Comm. of the ACM*, 41(2):54–57, 1998.

[31] J. Walpole, L. Liu, D. Maier, C. Pu, C. Krasic. Quality of service semantics for multimedia database systems. In *Proc. of the Conf. on Database Semantics*, pages 393–412, 1999.

[32] R. Wang. A product perspective on Total Data Quality Management. *Comm. of the ACM*, 41(2):58–65, 1998.

[33] R. Wang and S. Madnick. A polygen model for heterogeneous database systems : the source tagging perspective. In *Proc. of VLDB'90*, pages 519–538, 1990.

[34] R. Wang, V. Storey, C. Firth. A framework for analysis of data quality research. *IEEE TKDE*, 7(4):623–638, 1995.

[35] S. Weibel and C. Lagoze. An element set to support resource discovery : The state of the dublin core. *Int. J. on Digital Libraries*, 1(2):176–186, 1997.

[36] M. Xu and S. Gauch. Associated biological information retrieval from distributed databases. In *Proc. of CIKM'98*, pages 193–200, 1998.

[37] G. Zhou, R. Hull, R. King. Generating data integration mediators that use materialization. *J. of Intelligent Information Systems*, 6(2/3):199–221, 1996.