Archives of Business Research – Vol. 10, No. 3
Publication Date: March 25, 2022
DOI:10.14738/abr.103.11904.
Samoilenko, S. (2022). Detection and Propagation of Metadata Updates within CIS environment: Methodology and Conceptual Solution. Archives of Business Research, 10(03). 92-105.

Detection and Propagation of Metadata Updates within CIS environment: Methodology and Conceptual Solution

Sergey Samoilenko
Department of CIS/CS, Averett University

ABSTRACT

In order to adapt to the demands of the modern hypercompetitive environment many organizations integrate their information sources and form alliances. Cooperative Information System (CIS) provides a flexible form of integration that requires, due to absence of the common integrated schema, heightened attention to the data quality management. Commonly, the associated with the CIS' data metadata assumed to be static and of a high quality. However, in a modern fast changing competitive environment this assumption must be relaxed, and the issue of metadata quality management should be addressed explicitly. While research concerning data quality in CIS environment is abundant, no work, to our knowledge, has addressed the issues of how changes to metadata could be detected and propagated within the CIS environment. In this paper we attempt to address this research problem by proposing, first, a metadata update management methodology and, second, a conceptual design of the technical solution allowing for the detection and propagation of metadata updates within CIS environment.

Keywords: Computer Information Systems, Metadata, Information Integration, Complex Dynamic Systems

INTRODUCTION

In order to adapt to the demands of the modern hypercompetitive environment many organizations integrate their information sources and form alliances. Information integration, according to [5], has the goal of providing an integrated and coherent view of data stored in multiple, possibly inhomogeneous information sources. One of the forms of the integration is that of Cooperative Information System (CIS), which is a large scale information system that interconnects various systems of different and autonomous organizations, geographically distributed and sharing common objectives [1].

The integration system of CIS is of a "loose" type, under which the source schemas of each member of CIS are visible to its peers and no unified common schema is constructed. Consequently, the loose integration, which is often called read-only integration, does not allow for any interference of the integration system with the local data sources. This absence of common schema makes the CIS environment flexible, but, clearly, difficult to integrate. It was suggested by [7, 8, 9, and 10] that the issues associated with information integration represent one of the core problems in CIS.

It has been argued in [26] that the typical problems that occur at integration level are the ones that are related to the data quality. The obvious results of this problem include the deterioration of the data quality within CIS environment and resultant lack of cooperation between the

organizations-members of CIS. The damage is furthered if such applications as data warehouses or decision support systems are built on top of the integrated environment. The effect of the poor reliability of the integrated data on such applications has been reported in [19, 20, and 21].

Most of the research in the area of heterogeneous database integration has been directed at the resolution of structural and semantic conflicts [28, 29, 30, 31], as well as associated with these conflicts issues of integrity constraints in data integration [32, 33, 34]. In general, this area of research is concerned with the question of how to create an integrated environment.

Another stream of research has been focused on the issues of data quality within the integrated environment. The common research question within this area is how to maintain and manage quality of the data once it was integrated. The metadata that is associated with the data usually assumed to be static and of a high quality (for example, see [24]). However, the dynamic nature of the modern business environment brought about such accelerated rate of changes, that the frequent updates of not only data, but changes in metadata as well are required now for the sound decision support [23]. We would like to propose, therefore, that within the fast changing modern environment neither the static nature of the metadata nor its quality cannot be assumed or taken for granted any longer; as a result, the above mentioned assumption must be relaxed, and the issue of metadata quality management should be addressed explicitly.

While research concerning data quality in CIS environment is abundant, no work, to our knowledge, has addressed the issues of how changes to metadata could be detected and propagated within the integrated environment. However, we would like to argue that the state of data quality in CIS environment is in part dependent on whether or not members of CIS have the current information regarding the metadata of the inquired into data sources. Moreover, some data may be stored redundantly by different members of CIS. In this case the quality of the data within CIS would directly depend not only on controlling the redundancy of the data values, but on controlling the redundancy of the metadata as well. Therefore, if the local metadata updates have taken place within CIS environment, the relevant changes must be communicated to those members and applications of the CIS, which store these data redundantly or use the data supported by the changed metadata.

Consequently, we would like to argue that the process of metadata quality management in CIS environment must include, as its integral part, a process for managing of the metadata updates. Keeping in mind the loosely integrated nature of CIS environment, we propose that such process would consist, conceptually, of two components, local and global. The first, local, component would address the issue of managing of the updates at the local sources, while the second, global, component would be dealing with the detection and propagation of the updates within CIS environment. We would argue, based on this perspective, that while metadata update management methodology could potentially cover both local and global components, the technical solution must be limited to the latter. In other words, it would be only concerned with the detection and propagation of the metadata updates. The scope of functionality of the technical solution is based on the assumption that while the metadata changes could be detected and propagated across CIS environment, the actual corresponding changes at any local source might be only suggested and never forced upon.

URL: http://dx.doi.org/10.14738/abr.103.11904

However, to our knowledge, there exists neither a technical solution nor a methodology that would allow addressing the issue of managing of metadata updates. Therefore, we would like to formally define our research problem as follows:

> *In order to manage, in a consistent manner, the metadata updates within CIS environment, there must exist, first, a methodology according to which the updates are managed and, second, a technical solution that allows for addressing the issues of the detection and propagation of the metadata updates within CIS environment.*

In this paper we attempt to address this research problem by proposing, first, a metadata update management methodology and, second, a conceptual design of the technical solution. Clearly, because we consider that the actual implementation and testing of the proposed solution is beyond the scope of this paper, our work only partially addresses the stated above research problem.

The organization of the paper as follows: first, we would like to justify our research question by calling attention to the types of problems that could be caused by the undetected changes in metadata within CIS. Second, we describe a framework according to which propagation of metadata could be managed within CIS environment. Finally, we are going to propose a solution allowing for the implementation of metadata propagation within CIS.

JUSTIFICATION OF THE RESEARCH PROBLEM

While there are multiple definitions of metadata available, the common message that 'metadata is the data about a data' underlies all of them. Metadata refers to data about the meaning, content, organization, or purpose of data [35]. It can be as simple as a relational schema representing few tables or as complicated as information describing the resources, standards, element sets and domain specific structure of information defined in namespace schema. Functionally, metadata should be rich enough to provide data descriptions that are semantically meaningful to the applications using the data. Metadata should also be comprehensive enough to identify and reconcile heterogeneities between data provider and data receiver.

There is no common taxonomy for metadata, as a result, various researchers have classified metadata differently [36]. They include Content Independent Metadata (e.g. location, modification-date of a document, type-of-sensor used in a photograph), Content Dependent Metadata (e.g. size of a file, number of rows in a spreadsheet), Content Descriptive Metadata (e.g. markup tags describing the contents of an xml file), Domain Independent Metadata (e.g. java/C++ algorithms, html/xml type definitions) and Domain Specific Metadata (e.g. database schema type used in a database).

However, despite the differences in classification, metadata can be viewed as conceptual description that abstracts out the underlying representational details and capture the information content. Once the relevant high-level contextual descriptions have been abstracted, retrieving the relevant data and display it to the user is a reverse process that is effected simply by identifying and matching data to the lookup criteria defined by the end user.

In most cases, (whether dealing with relational, object or object-relational databases), relationships that exists between database objects cannot be stored in the database, but are

mapped as extra information that is stored in the metadata. As contextual mapping of the database object descriptions and relationships can become extensive and complex, even under highly constrained domain specifications, it has to be assumed that each organization (and its database) has available to it, an ontology that corresponds to its specific domain.

Since organizations that participate in a CIS are brought together by common goals or interests, the domain constraints within a CIS can be considered as equals or subsets of an overall global ontology applicable to that environment. The availability of this commonality of an extensive global schema can be used as the basis to determine an effective way in which metadata changes at local sources can be identified and propagated within the boundaries of the global schema that is common to the CIS.

For the purposes of this paper we are going to use the following definition:

> *Metadata is an explicitly declared within database schema set of constraints according to which data values are obtained, stored, accessed, and displayed.*

In this section of the paper we would like to argue that the lack of attention to the metadata quality could potentially result in the following implications:

1. difficulty in cooperation between the organizations-members of CIS (due to inconsistency of the available metadata)
2. proliferation of the different 'versions' of the metadata supporting the same type of the data as well as the same data values
3. deterioration of the data and metadata quality within a single organization, as well as within CIS environment in general
4. applications that are built on top of the integrated environment might experience problems that are caused by the quality of the integrated data and metadata

Since metadata can be classified in different ways depending on what information and representational content is captured, in this paper we seek to identify a generalized approach for identifying metadata changes that could be applicable to all forms of metadata representations. In this regard, we use the well recognized TDQM_CIS methodological cycle as the basis for our conceptual framework.

The scope of this research paper does not allow us to present all possible scenarios depicting metadata changes. However, in order to provide a clear and concise explanation of the conceptual model presented in the paper, we attempt to use examples that might seem overly simple. We would argue, however, that the offered examples are representative of the most common occurrences of metadata changes, which quite often require intervention of a DBA. However, due to the nature of the framework, the same concepts can be applied to address other instances related to metadata changes.

As a first illustrative example, consider the case where one of the organizations members of CIS has changed the metadata related to *Customer*, i.e., change the length of email address from 50 to 75 characters. The company might have decided to extend the field size to accommodate longer email addresses based on its experience with the inability to handle email addresses longer than 50 characters. Possibly other members in the CIS may not have come across a similar situation to warrant changes to their schema. Other CIS participants can avoid the

URL: http://dx.doi.org/10.14738/abr.103.11904

potential metadata quality problem associated with longer customer emails by taking preemptive steps by correcting their existing schema definitions based on the information that was made available. Benefits from sharing such 'best practices' within the integrated environment thus ensure that the members are made aware of a data quality problem even before it strikes their location.

However, the schema changes could be made within a single organization without being detected and incorporated within CIS. The result of such turn of events shall be proliferation of the multiple versions of the same data existing in CIS, which, while having the same values, are structured differently (i.e., having different schema definitions). Clearly, this is detrimental to the consistency of information and quality of metadata. Moreover, the organizations that have not incorporated the changes, in the extreme cases, might not be able to access or display the data because of the changed field length.

As a second example let us consider a situation where an organization changes the length of *workphone* attribute for a *Customer* from varchar(10) to varchar(11). This is a practical problem currently facing institutions moving to cheaper voice-over-ip (VOIP) telephone numbers (calls to VOIP telephones mandate the inclusion of 1 followed by the area-code even within local dialing zones). If there are exist other organizations sharing the *workphone* attribute of a *Customer* class, their failure to administer corresponding changes to their schema definitions will result in inconsistency and potentially, in creation of duplicate records.

So is the case where a metadata change is made so that NULLs are no longer (NOT NULL) allowed for a particular field. Failure to implement corresponding changes to other organizations that are using the same data can cause inconsistent data and affect the overall quality of information commonly shared between the members.

THE PROPOSED FRAMEWORK FOR METADATA UPDATES PROPAGATION

The purpose of this part of the paper is to demonstrate, at the conceptual level, how the propagation of metadata updates could be handled in CIS environment. One of the possible approaches is the TDQM_CIS methodological cycle (shown below), which has been used in [24] as a framework for managing data quality. Although acknowledging the differences between data and metadata, we argue that the same framework could be used to manage quality of metadata, because, after all, metadata is the data about data, and the difference between the two exists only at the semantic level. We do not claim that the adapted methodological framework could provide a comprehensive solution to the issues of the metadata quality; however, we do propose that the framework allowing for managing data quality could be extended to manage the quality of metadata as well.

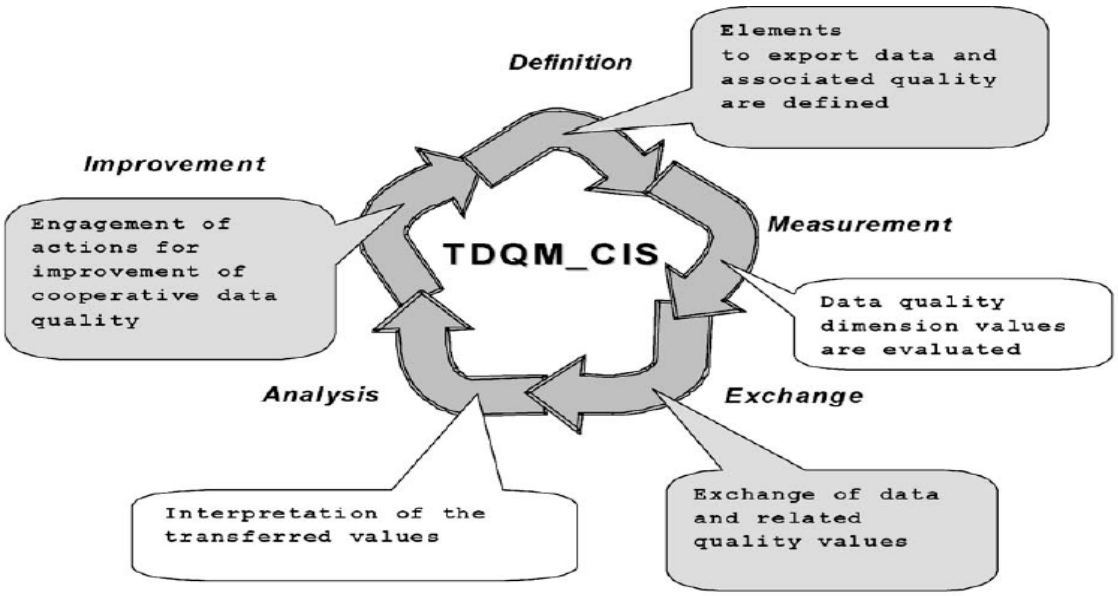


Figure 1. The phases of TDQM_CIS cycle (from [24])

At this point we would like to briefly outline, for illustrative purposes, how the five phases of TDQM_CIS could be utilized within CIS to manage propagation of metadata updates. The path through the TDQM_CIS cycle starts with DBA of local data source having modified the metadata. At the *Definition* phase the decision is made regarding what part of changes to metadata must be made 'visible' within CIS environment, i.e., what part of the changes must be kept "private" and which part is "public". For example, the management might decide to list all products offered by the company, their detailed descriptions and sale price, but may not want to reveal the manufacturing costs and expenses. The DBA can use metadata tags to defines sections of the metadata that are available to the public domain and those that cannot be exposed. Accessible parts of the metadata can be easily made visible to other members of the CIS using web service query and description languages.

At the *Measurement* phase the DBA evaluates the updates in the metadata against the CIS' criteria for metadata quality. The following aspects of the data quality, consistent with those identified in [27], could be used to evaluate the quality of metadata as well:

1. Accuracy: a measure of how accurately metadata describes data. For example, *varchar(20)* would be considered inaccurate depiction of the attribute *Phone_Number*, while *integer* or *number* would be considered accurate.
2. Completeness: a degree to which specific values are included in the description of the data. As an example, domain for the attribute *Student* consisting of five values would be considered more complete than the domain of three values.
3. Representation Consistency: a degree to which data values of the same type are described in the same manner. For example, if the data source describes the *SS#* and *StudentID#* , both being numeric values, as *varchar(20)* and *integer*, respectively, such data source is representation inconsistent.
4. Interpretability: it is related to the format in which data are specified and to the clarity of data definitions [25]

URL: http://dx.doi.org/10.14738/abr.103.11904

5. Relevance: it is a measure of the appropriateness of propagating the metadata update across CIS environment. For example, the metadata update could be eligible, yet not appropriate for propagation across CIS. It might happen in the case of the data being "public" but highly specific to a particular organization.

We do not claim that the presented above list of data quality aspects is exhaustive or sufficient. It is presented here for illustrative purposes only and in reality the dimensions of metadata quality would be determined by the needs and interests of a particular CIS. The purposes of *Measurement* phase is to determine whether the metadata updates fall under any of the predefined by CIS data quality dimensions. In the case if the changes to the local metadata are potentially relevant to the overall quality of metadata in CIS, *Exchange* phase takes place. But before we proceed further, we would like to point out that it is the Measurement phase that "determines" the scope of the metadata we are dealing with in this paper. Meaning*, in this paper we are dealing with the subset of the local metadata that could be considered eligible for propagation within CIS by the DBA of the local data source.*

The *Exchange* phase is the most difficult to implement. It is at this stage the propagation of the updates takes place, therefore, the target members of CIS must be identified by the local data source, updates must be send by the source and received by the target. The next part of the paper is dedicated to the implementation of the *Exchange* phase and shall cover the metadata propagation specifics in details.

During the *Analysis* phase the target organization receives the metadata updates. It is at this stage the decision is made whether or not the propagated updates should be incorporated within the metadata of the local source. Again, the decision could be made based on the comparison of the metadata quality values of the received update against the local quality values. This decision most likely would be made by the DBA of the recipient of the updates.

Finally, during the *Improvement* phase the actual changes to the local metadata are made by the DBA. The adapted framework, in the form of Total Metadata Quality Management in CIS (TMDQM_CIS) is depicted below in figure 2.

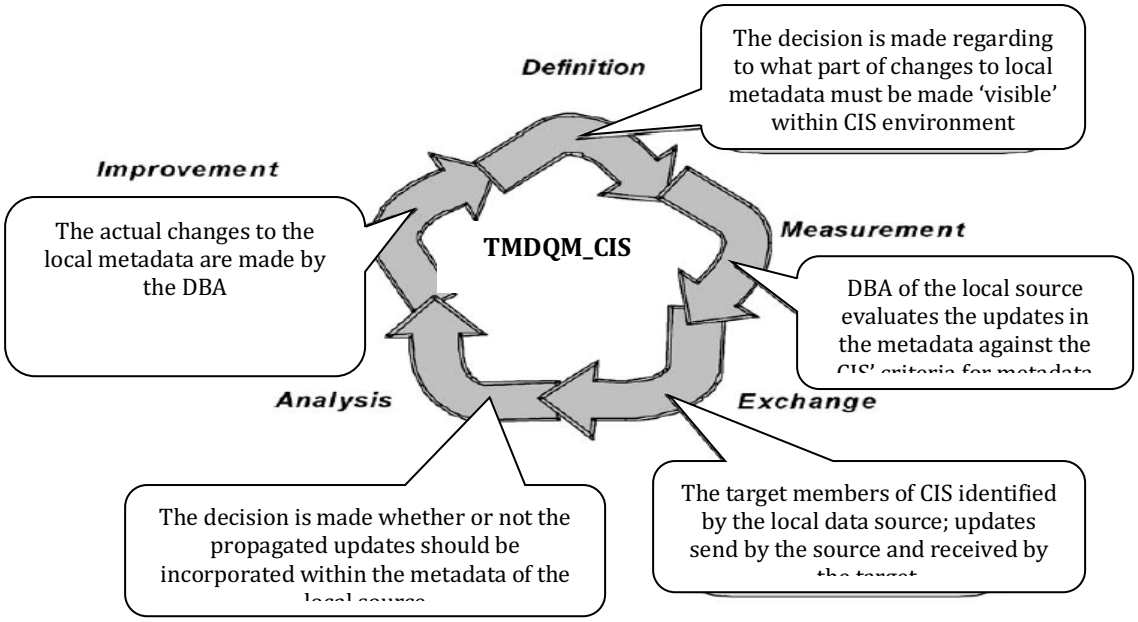


Figure 2. The phases of TMDQM_CIS cycle for metadata updates

The presented above methodology presents a solution to the first part of the research problem as it was defined in the Introduction section, namely, that:

> *"in order to manage, in consistent manner, the metadata updates within CIS environment, there must exist… a methodology according to which the updates are managed."*

SOLUTION ALLOWING FOR THE IMPLEMENTATION OF METADATA UPDATE PROPAGATION

Within the scope of this paper we would like to define the metadata update propagation as *a process of making a new metadata version visible and explicitly communicated to the relevant targets within CIS environment*. We identify three criteria according to which a solution to this problem could be evaluated.

The first criterion is that of the flexibility of CIS environment. The solution provided must not result in the decreased autonomy of the local sources and should not create any global supervising and controlling entity, i.e., global integration layer of some sort.

The second criterion is that of maximum autonomy and independence. This means that the provided solution must be as independent of human (read, DBA) input as possible, yet requiring a DBA supervision to ensure relevance and update consistency. This precludes the solution from becoming an extra burden in the process of the database management; it is also lessening the chance of the solution becoming affected by a human error.

Finally, the third criterion is that of expediency. It implies that the solution must be able to provide updates to the relevant targets as soon as possible after metadata update have taken place at the source. This shall contribute to the creation of the proactive, rather reactive metadata update management environment.

URL: http://dx.doi.org/10.14738/abr.103.11904

Keeping the three criteria in mind, we offer a solution in the form of the Metadata Update Management Module (MUMM), which resides at the local data source and loosely coupled with it. Conceptual representation of the proposed solution, demonstrating the interrelationships between CIS environment, MUMM, and the local data source depicted in Figure 3.

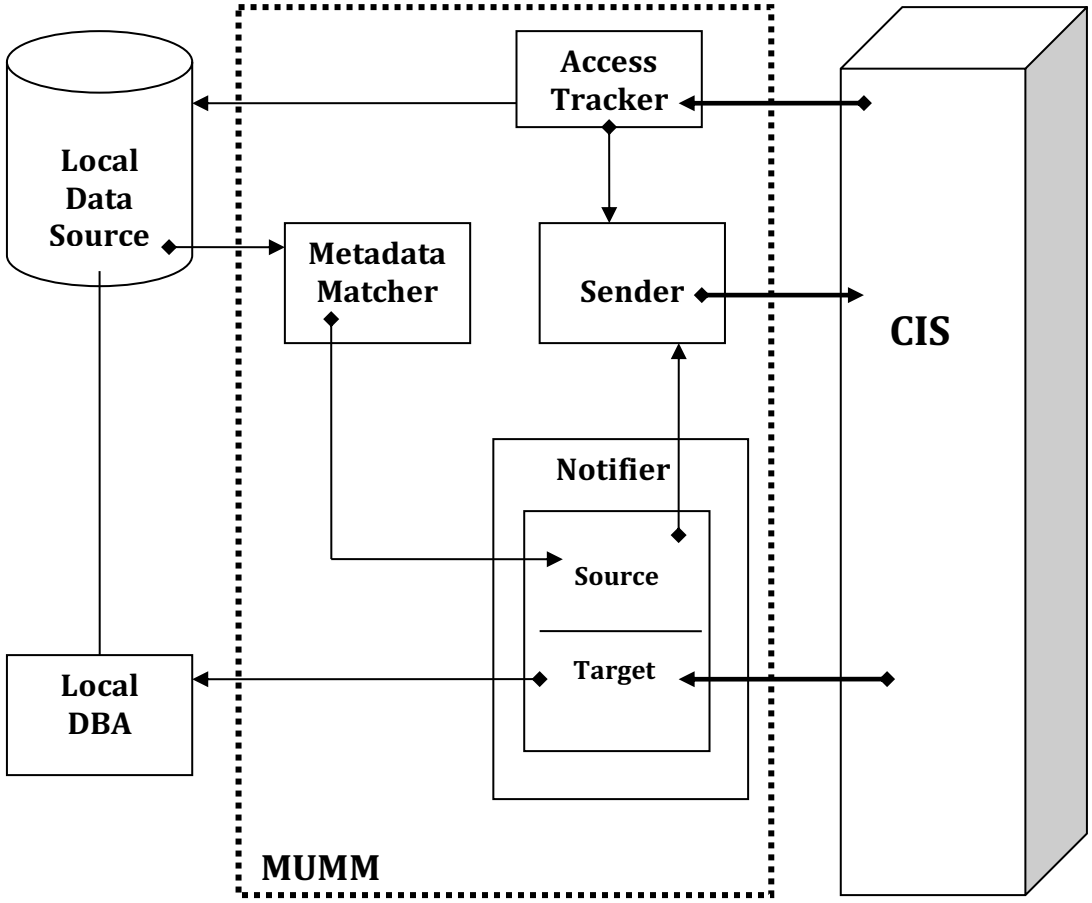


Figure 3. Conceptual Representation of the Proposed Solution

MUMM consists of the four components - Access Tracker, Metadata Matcher, Notifier, and Sender. We offer a brief overview of the components next.

Metadata Update Management Module: Overview of the Components

Access Tracker

The purpose of this component is to track the inquiries of the members of CIS into the local data source. It shall allow creating an 'address book' containing, for example, authentication identities, IP addresses and communication protocol details of the members of CIS that are interested in the data contained in the local data source. Once the address book is created, Access Tracker should also be able to maintain following information for each local source: total number of accesses, frequency of accesses (per time lag Δt, see below), as well as the date of the last access. The purpose of the total number of accesses is to determine the members of CIS that accessed the local data source the most. The frequency of accesses would suggest the list of members of CIS that are interested in or dependent on the data of the local data source the most.

Finally, the information about the date of last access could provide some hints on whether a particular organization is still an active member of CIS. This information could be used to create a 'hierarchy of priority', according to which (especially in the case of heavy network traffic) metadata updates could be propagated across the CIS environment. For example, 'high priority' CIS members would get the updates earlier than 'low priority' members.

Metadata Matcher

The purpose of metadata matcher is to actually identify the presence of the metadata updates at the local source. This is done by means of 'schema matching', which is a process of comparing the most current (today's) version of the metadata to the previous version (yesterday's). The time lag Δt , duration of which should be agreed upon by all members of CIS, represents the frequency of the checking for updates and is, in essence, a distance in time between two metadata versions (the current version and the previous one) that are matched to each other by this component. One of the possible ways of identifying the updates is through the conversion of the local schema into the Object-Oriented format and performing the 'intersect-like' operation on the two sets that represent two schemas. If the result of the intersection is not an empty set, it indicates that the metadata updates have taken place. In the case of Relational databases metadata updates could potentially be detected directly through the use of triggers. If changes in the metadata have been identified by metadata matcher, these changes then send to Notifier component.

Notifier

The purpose of this component is to hold in memory the identified updates. Notifier consists of two subcomponents, *Target Notification* and *Source Notification*. *Target notification* subcomponent holds the metadata updates received from other members of CIS. Once the input is detected, the message is send to DBA of the local data source (we assume that this component is coupled with organizational e-mail or some other communication medium), who then decides if the received updates are appropriate to his organizations. *Source notification* subcomponent holds the metadata updates that were identified at the local source (with which MUMM is coupled). Once this subcomponent has identified an input from Notifier, it then contacts the Sender. Overall, the implementation of this component could probably be similar to the implementation of simple messaging service.

Sender

The purpose of the Sender is to send the updates that are held in Notifier to the sources, the addresses and identities of which provided by Access Tracker. This component could be implemented as a trigger-based AutoFill-populated template with two required fields/areas, 'address' and 'message'. The address information is supplied by Access Tracker, while the message information by Source notification. If Source notification subcomponent' memory contains no information regarding the metadata updates, the required field 'message' would remain empty and the message would not be send. But once Source notification receives input, the required 'message' area would be populated and the message would be send automatically.

The Figure 4 (see below), provides an illustration how the propagation of the metadata updates, mediated by two MUMMs, takes place across the CIS environment.

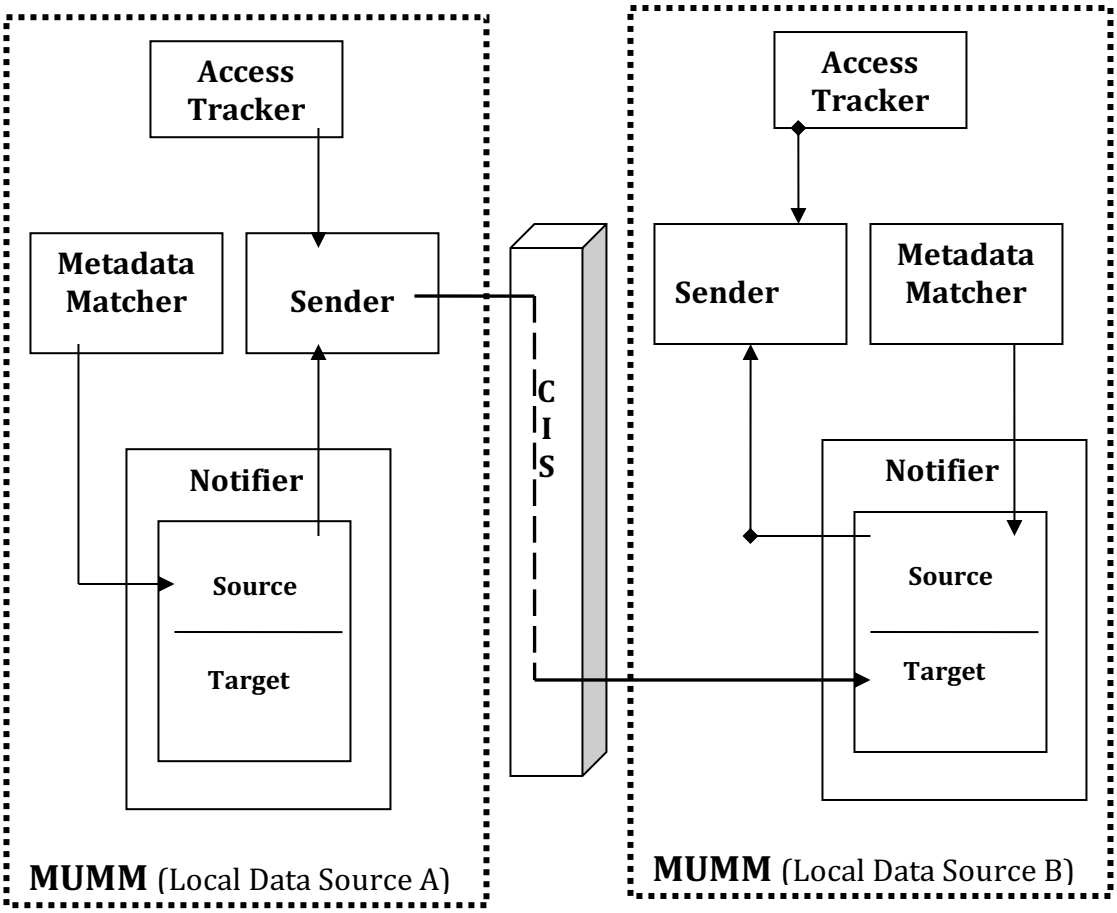


Figure 4. Interaction between two MUMMs across CIS environment

In the Introduction to this paper we have proposed that the functionality of the technical solution should include metadata updates detection and propagation. We would like to argue that the proposed above solution allows for both, metadata updates detection as well as metadata updates propagation. In particular, Metadata Matcher component is responsible for the detection of the changes at the local source, while Target Notifier component is responsible for the detection of the metadata updates that took place elsewhere. As for the metadata updates propagation, this functionality is implemented by using Access Tracker, Source Notifier, and Sender components. The conceptual design thus also presents a solution to the second part of the research problem as it was defined in the Introduction, namely, that there must exist a "*technical solution that allows for addressing the issues of the detection and propagation of the metadata updates within CIS environment.*"

CONCLUSION

The paper provides a conceptual approach on how metadata updates can be propagated to participating members of a CIS. The paper justifies the need for a system that allows easy dispensation of metadata changes to ensure higher quality of shared metadata on which distributed applications (for example, B2B based information exchange programs) or members rely on.

One of the strengths of the conceptual model presented is that the solution approach is based on existing TDQM_CIS model. TDQM is a familiar topic in the data quality realm of research and extended to CIS. Implementation and acceptance of the proposed model in this paper will hence be an easier task to accomplish. Another advantage is that the MUMM architecture can be modeled and developed using any programming language while maintaining independence from the underlying data source. Although there is reliance on DBA involvement to scrutinize the notification of change, this step only helps to ensure that relevant metadata changes are applied based on business requirements of the organization receiving the notification.

The paper only presents a conceptual model of the solution approach. A full implementation and testing will be required to identify and address issues that could arise in a real CIS arena. As mentioned in the 'justification of research problem' section, metadata helps to provide context descriptions and are classified along different lines based on difference in the context and domain where they are used. The conceptual model proposed in the paper will have a stronger foothold if additional working examples can be tested and analyzed to evaluate how the framework operates across scenarios from each individual classification described previously. Future research will focus on addressing this area and will help to provide an all-embracing solution to the problem definition. In addition, the proposed model is only semi-automated and the dependence on the DBA for change approval could affect the time lag Δt used by the metadata matcher, especially if change verification has to approved by the upper management. Overall, the paper makes a strong argument on how the solution presented in the paper can address the metadata update issues relevant in highly dynamic CIS environments that is prevalent in the current mode of conducting global business.

References

[1]. G. De Michelis, E. Dubois, M. Jarke, F. Matthes, J. Mylopoulos, M. Papazoglou, K. Pohl, J. Schmidt, C. Woo, E. Yu, "*Cooperative Information Systems: A Manifesto*" in: M. Papazoglou, G. Schlageter (Eds.), Cooperative Information Systems: Trends & Directions, Academic Press, New York, 1997.

[2]. Y. Wand, R. Wang, Anchoring data quality dimensions in ontological foundations, Commun. ACM 39 (11).

[3]. T. Redman, Data Quality for the Information Age, Artech House, 1996.

[4]. Scannapieco, M., Virgillito, A., Marchetti, C., Mecella, M., Baldoni, R. The DaQuinCIS architecture: a platform for exchanging and improving data quality in cooperative information systems. Information Systems 29 (2004) p.551-582.

[5]. D. Calvanese, G. De Giacomo, M. Lenzerini, D. Nardi, and R. Rosati. *Information integration: Conceptual modeling and reasoning support*. Proc. CoopIS'98, pp.280- 291, 1998.

[6]. R. D. Holowczak and W. S. Li. *A Survey on Attribute Correspondence and Heterogeneity Metadata Representation*. In Proc. MetaData Conference, http:/www.llnl.gov/liv com/metadata/md97.html, 1996.

[7]. R. Hull. Managing semantic heterogeneity in databases: A theoretical perspective. In *Proc. Of PODS-97*, 1997.

[8]. C. Knoblock and A. Levy, editors. *AAAI Symposium on Information Gathering from Heterogeneous, Distributed Environments*, number SS-95-08 in AAAI Spring Symposium Series. AAAI Press/The MIT Press, 1995.

[9]. J. Widom. Special issue on materialized views and data warehousing. *IEEE Bulletin on Data Engineering*, 18(2), 1995.

[10]. G. Wiederhold. Special issue: Intelligent integration of information. *J. of Intelligent Information Systems*, 6(2/3), 1996

[11]. C. Batini, M. Lenzerini, and S. B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys*, 18(4):323–364, 1986.

[12]. A. Hurson, M. Bright, and S. Pakzad, editors. *Multidatabase Systems: An Advanced Solution for Global Information Sharing*. IEEE Computer Society Press, 1994.

[13]. A. Sheth and J. Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Computing Surveys*, 22(3), 1991.

[14]. A. Gupta and I. S. Mumick. Maintenance of materialized views: Problems, techniques, and applications. *IEEE Bull. on Data Engineering*, 18(2):3–18, 1995.

[15]. W. H. Inmon. *Building the Data Warehouse*. JohnWiley & Sons, second edition, 1996.

[16]. A. Dogac et. al. METU Interoperable Database System. *ACM SIGMOD Record*, 24(3):56 -- 61, September 1995.

[17]. Holowczak, R. and Li, W., A Survey on Attribute Correspondence and Heterogeneity Metadata Representation. IEEE, 1996.

[18]. Kashyap, V. and Sheth, A. Semantic Heterogeneity In Global Information Systems. The Role of Metadata, Context and Ontologies. June, 1996. http://lsdis.cs.uga.edu/lib/download/KS97.pdf

[19]. R. Kimball: Dealing with Dirty Data. DBMS Magazine 9:10, September 1996, Miller Freeman, Inc, 1996.

[20]. J. Bischoff, T. Alexander: Data Warehouses: Practical Advice from the Experts. Prentice-Hall, 1997.

[21]. M. Jarke, Y. Vassiliou: Data Warehouse Quality Design: A Review of the DWQ Project. Invited Paper, Proc. of the 2nd Conference on Information Quality. Massachusetts Institute of Technology, Cambridge, 1997.

[22]. M. Schönhoff, M., Strässler , and K., Dittrich, "Version Propagation in Federated Database Systems" in Proceedings of the International Database Engineering & Applications Symposium, 2001, pp.189 - 198.

[23]. A. Sen, "Metadata management: past, present and future." in *Decision Support Systems,* 1043, 2003, pp. 1-23.

[24]. M. Scannapieco, Virgillito, et al. "The DaQuinCIS architecture: a platform for exchanging and improving data quality in cooperative information systems." In *Information Systems,* Vol. 29, 2004, pp. 551-582.

[25]. R.Y. Wang, and D.M. Strong, "Beyond Accuracy: What Data Quality Means to Data Consumer," in *Journal of Management Information Systems*, 1996, 12, 5-34.

[26]. M. Gertz, "Managing Data Quality and Integrity in Federated Databases" in Proceedings of the IFIP TC11 Working Group 11.5, Second Working Conference on Integrity and Internal Control in Information Systems: Bridging Business Requirements and Research Results, 1998, pp. 211-230.

[27]. C. Cappiello, C. Francalanci, B. Pernici, P. Plebani, M. Scannapieco, "Data quality assurance in cooperative information systems: a multi-dimension quality certificate," in: Proceedings of the ICDT'03 International Workshop on Data Quality in Cooperative Information Systems (DQCIS'03), Siena, Italy, 2003.

[28]. W. Kim, J. Seo:, " Classifying Schematic and Data Heterogeneity in Multidatabase Systems" in *IEEE Computer* ,24:12 (December 1991), pp. 12-18.

[29]. S. Spaccapietra, C. Parent, Y. Dupont, "Model Independent Assertions for Integration of Heterogeneous Schemas" in *VLDB Journal* ,1:1, pp. 81-126, 1992.

[30]. A. Sheth, V. Kashyap, "So Far (Schematically) Yet. So Near (Semantically)" In D. Hsiao, E. Neuhold, R. Sacks-Davis (eds.), Interoperable Database Systems (DS-5), North-Holland, Amsterdam, The Netherlands, 1993.

[31]. W. Kim, I. Choi, S. Gala, M. Scheevel ," On Resolving Schematic Heterogeneity in Multidatabase Systems" In (Kim 1995), pp. 521-550.

[32]. M.P. Reddy, B.E. Prasad, A. Gupta, " Formulating Global Integrity Constraints During Derivation of Global Schema" in *Data & Knowledge Engineering* ,16:3 (1995), pp. 241-268.

[33]. M. W. Vermeer, P. M. Apers, " The Role of Integrity Constraints in Database Interoperation" In T. Vijayaraman, A. Buchmann, C. Mohan, N. Sarda (eds.), VLDB'96, Proceedings of the 22th International Conference on Very Large Data Bases, Sept. 3-6, 1996, Bombay, pp.425-435, Morgan Kaufmann Publishers, 1996.

[34]. S. Conrad, M. Höding, G. Saake, I. Schmitt, C. Türker, “Schema Integration with Integrity Constraints” In C. Small, P. Douglas, R. Johnson, P. King, N. Martin (eds.), Advances in Databases, 15th British National Conf. on Databases, BNCOD 15, pp. 200-214, Lecture Notes in Computer Science 1271, Springer-Verlag, Berlin, 1997.

[35]. M. Siegel, S. Madnick, “A metadata approach to resolving semantics conflicts,” Proceedings of the 17th International Conferene on Very Large Databases, 1991.

[36]. V. Kashyap, A. Sheth, “Semantic Heterogeneity in Global Information Systems: The Role of Metadata, Context and Ontologies,” June, 1996.