A Proposed Framework for the Analysis of Source Data in a Data Warehouse

M. Pamela Neely
Marist College
Pamela.neely@marist.edu

Abstract

This paper introduces a framework for the analysis of data quality in source databases, prior to migration to the data warehouse. Additionally, as part of the framework, a tool for the collection of meta-data is proposed. The use of the framework, in conjunction with the tool, will allow the developer of a data warehouse to allocate the scarce resources available for data cleansing. This is accomplished by identifying the data fields that yield the greatest benefit to the warehouse and focusing cleansing efforts on those fields. Additionally, it is proposed that when the meta-data tool is completed, it is possible to assign the task of specific data field identification to novices on the data warehouse development team.

Introduction

In a study by Wixom and Watson (2001), examining the factors that affect data warehouse success, it was concluded that the quality of data in a data warehouse is a critical factor in the success of the warehouse. Wixom and Watson's research supports the previous data warehousing literature, showing that high quality data creates value for the organization. However, although they pose the question, "…can a data warehouse even exist without data quality? (Wixom and Watson 2001, pg. 35)", their research does not show how data quality is achieved.

In this paper, I propose a framework, the Data Quality Analysis Framework (DQAF), for the analysis of source data, prior to migration to a data warehouse. The use of this framework, and a related relational database tool for the collection of meta-data (data about the data), can begin to address the question of how to achieve data quality in a data warehouse. A data warehouse is a dynamic system, growing and changing as user needs grow and change. Data quality is an ongoing concern within the data warehouse and the framework provides a platform for the analysis of source databases throughout the life of the system.

Background

Integrated data repositories, also known as data warehouses, are regularly used to support management decision-making (Goodhue and Wybo 1992) and data mining activities (Forgionne and Rubenstein-Montano 1999). These integrated repositories consist of data from many source databases, which have been designed to support on-line-transaction-processing (OLTP) systems

for day-to-day activities. The data is brought together in one structure to support on-line-analytical-processing (OLAP) systems, which includes multi-dimensional views of data for decision-making, and data mining.

It is essential to both management decision-making and data mining that the data in these repositories are of high quality. Many of the dimensions of data quality, as defined by Wang and Strong (1996), are important in a data-warehousing context. Additionally, fitness for use (Tayi and Ballou 1998) is key. The data must be *accurate* to result in correct decisions. Furthermore, data that is used for secondary purposes, as is the case in a data warehouse, will be judged differently from data that is used for primary purposes, i.e. the transaction processing system. Thus, the *context* of the data becomes a critical determinant in the decision as to the quality of the data. The degree of c*ompleteness* for primary use may be much greater than the degree of completeness required for the data warehouse. Each of these data quality dimensions will contribute to the overall fitness for use as defined by the users of the data warehouse.

The flow of data from source to warehouse is depicted in Figure 1. The data can be examined at multiple points in this flow. Research in source database quality (Storey and Wang 1998) shows us that examining the data at the source is possible. However, much of this research focuses on database design and the necessary steps to take in creating a database that will create an environment where data quality issues will be addressed. For example, validation rules, codes, and date parameters can all be implemented to help ensure quality data in the source. However. at the time of integration it is too late for considering many of these issues. Additionally, many of the challenges associated with integrating multiple data sources are not considered at the source, such as consistent field names across data sources. If data were coming from multiple organizations, then it would have been impossible to address these issues when the databases were created. Thus, although the data can be examined at its source, it is generally not a realistic option to change the source and examining the data elsewhere should be explored.

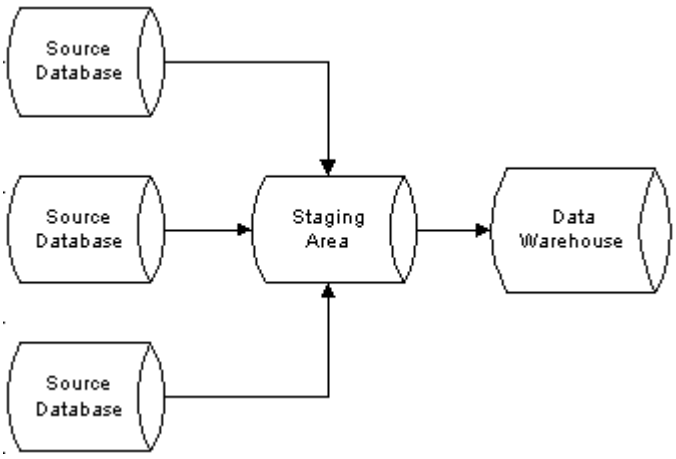

Figure 1- Data Flow from Source to Warehouse

Ideally, the data will be examined in the staging area, prior to migration to the warehouse. Changes (e.g. field names and types) can be made in the staging area that will not affect the source. This allows the developer of the warehouse to analyze data across sources, and determine

exactly what data is needed in the repository. Additionally, the data warehouse developer is in a unique position to evaluate data from a variety of sources and will be able to recommend the best data sources for the warehouse. Finally, the developer can provide feedback to individuals in charge of the source databases regarding the quality of their data.

Framework Development

The current study involved development of a preliminary framework and related tool for collection of meta-data. The framework was then populated using the results of a series of semi-structured interviews of data warehouse developers and users. In a parallel process, the portion of the framework related to the meta-data tool was tested in two pilot tests. The results of the interview analysis and pilot tests were then used to modify the preliminary framework and create a new framework.

Preliminary Framework

The research began with the development of a preliminary framework, the Data Quality Audit Process (DQAP). It was built on concepts used in information systems (IS) auditing, database design, and data quality, as well as the financial statement audit framework. This preliminary framework, as shown in Figure 2, consisted of three parts: Planning the Data Quality Audit, Executing the Data Quality Audit Program, and Reporting the Findings.

Figure 2- Data Quality Audit Process (DQAP)

The preliminary framework formed the basis for a questionnaire, designed to interview data warehouse developers and users. The results of these interviews were then used to refine and create a new framework.

Embedded in the DQAP is a tool for assimilating meta-data. This tool, known as an Enhanced Data Dictionary (EDD), is an extension of the data dictionary typically found in database documentation. As seen in Table 1, there are columns unique to a data warehouse. For example, the column labeled "Skip?" is used to alert the developer that this data field will not be used in the warehouse and thus, no efforts should be made to determine the quality of the data. The EDD is designed to capture the meta-data that is available from a variety of resources into one document.

Field Number	Column Heading	Field Description	Type/ Attribute/Size	Value	Skip?	Field Type
1	Client ID	Unique identifier for each client (Client ID) (internal)	Alpha-numeric /formatted text/15		N	Key
2	Client Name	Client Name	Alpha-numeric /free text/30		N	Text
3	SOURCE_CD	Site	Alpha-numeric/ Coded/15	Fairfield Highgate Other	N	Code
4	Agency ID	Agency ID (this field was used for SSN, then changed to PA#- still also tracking PA# in Identifiers table)	Alpha-numeric/Free Text/15		Y	Text

Table 1- Enhanced Data Dictionary (EDD)

It was recognized that implementation of the framework was potentially a multi-year project. Thus, only a portion of the framework was involved in testing. In a parallel process to the interviews, two pilot tests were conducted to determine the value of the EDD and the role it played in the overall framework. The results of these pilot tests further refined the framework.

In the next sections the interview process as well as the pilot test procedure will be discussed.

Interviews

A series of ten interviews was conducted with individuals responsible for the development and use of a data warehouse. These professionals came from a variety of

backgrounds- industrial, banking, telecommunications, healthcare and government. The questions asked during the semi-structured interview were developed from the DQAP. They addressed the concepts of planning the audit, executing the audit and reporting the findings. Questions were constructed to populate the DQAP, and thus closely followed the nodes of the DQAP. For example, the following questions were asked to elicit data regarding obtaining information about an organization's data quality policies:

- Do you utilize a data dictionary in your process? If so, what purpose does the data dictionary serve in your analysis?
- Who defines your business rules? Are they codified? Who codifies them?

Each interview each lasted approximately one hour. They were taped and transcribed, then analyzed using a qualitative software tool. Key findings from the interviews included:

- Data quality (DQ) was a primary concern for all of the developers
- DQ was considered early in the project, and continued to be an ongoing concern as development progressed
- Data Warehouse developers were not auditors, and did not follow an auditing methodology
- The development of the warehouse generally followed a systems development life cycle (SDLC) approach
- Analysis of DQ had no standard approach. Tools were used when possible, and they ranged from spreadsheets, to programming, to data quality tools
- A data dictionary was unavailable for most of the source data, although attempts were frequently made to collect the meta-data typically found in a data dictionary
- Many of the developers mentioned the criticality of knowing the data suppliers and having a place to go when problems arose

Pilot Tests

In a parallel process, 2 pilot tests were conducted to test the effectiveness of a portion of the DQAP, specifically, the design and use of the EDD. The pilot tests were conducted in two undergraduate classes, one focused on data quality and the other a data management class. The students were instructed to construct a portion of the EDD related to one data source, using the source documents that were available. These documents included a list of field headings and descriptions, as well as a codebook. The students were then asked to analyze a completed EDD for several data sources related to the same warehouse project. Their analysis was designed to highlight "across data source" anomalies such as homonyms and synonyms as well as incompatible codes and field types. Finally the students were asked to look at actual data from the data sources and complete the EDD using this visual inspection of the data.

Data Quality Analysis Framework (DQAF)

As a result of the interviews and the pilot tests, a proposed framework, the Data Quality Analysis Framework (DQAF) was developed (see Figure 3). This framework differs considerably from the DQAP, principally because of the focus of the developers. They think in terms of systems development, not auditing. Although they found the questions regarding audit

professionals and reporting audit findings intriguing, they did not see them as relevant in the process of data warehouse development. Overall, they felt that the audit considerations should be a concern for another group. Their goal was not to give an opinion on the data, but rather to ensure that it was acceptable to meet the needs of the data warehouse.

The DQAF is designed to fit into the analysis phase of the Systems Development Life Cycle (SDLC), a structured process for developing information systems. It is an iterative process and attempts to incorporate best practices that were elicited from the interviews and to address the deficiencies found in the current practice. The framework is continuing to evolve as the research progresses.

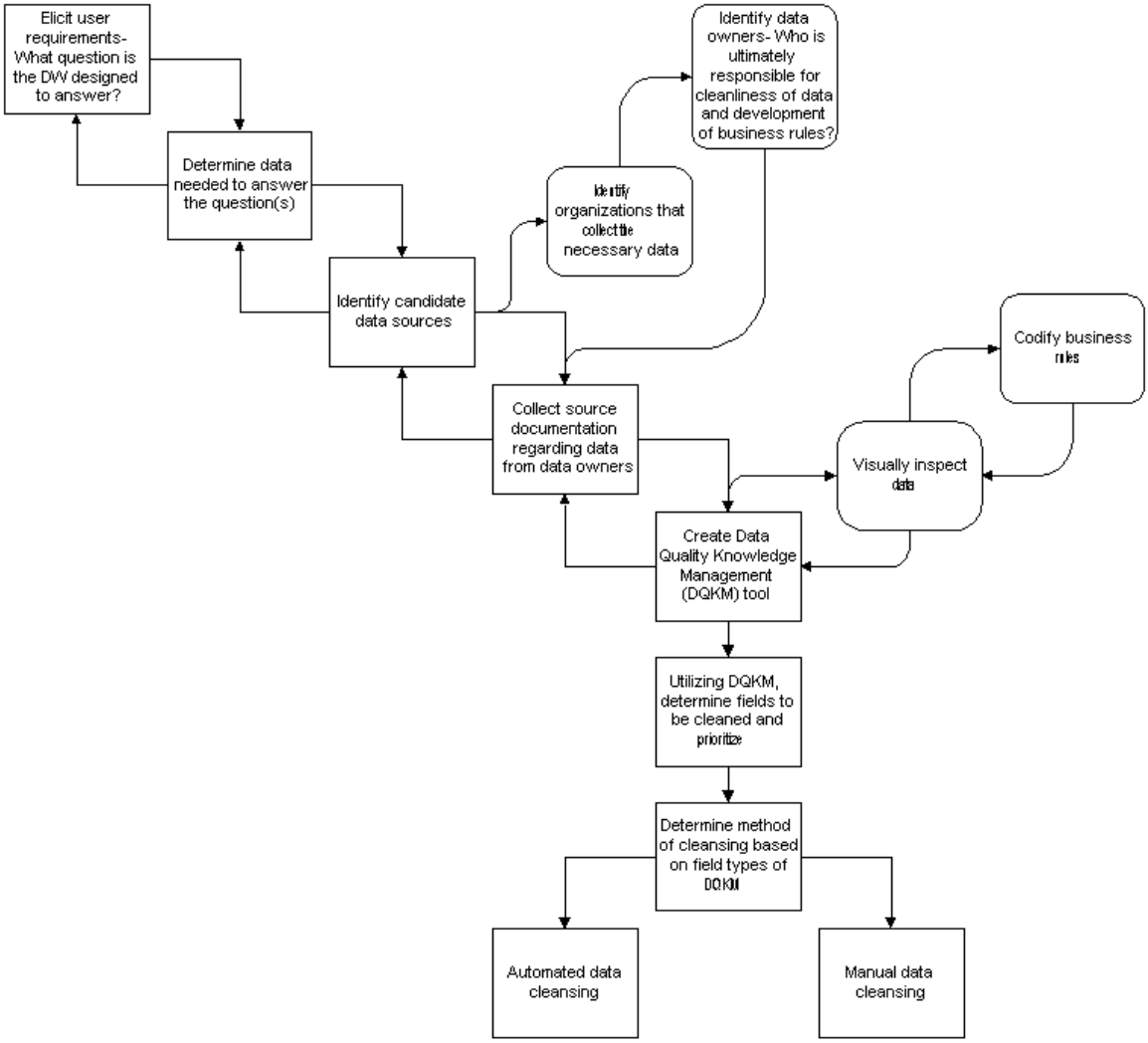

Figure 3- Data Quality Analysis Framework (DQAF)

As noted previously, the Data Quality Analysis Framework (DQAF) is an iterative process. The interviews clearly indicated that not only was data quality considered early in the

development process, but also considered continually throughout the process. Thus, the DQAF shows many loops that resemble the waterfall approach of the SDLC. In the next section I will discuss the various components of the DQAF.

Elicit User Requirements

A key activity in the analysis phase of the SDLC is gathering user requirements. In a data warehousing project the object of this phase is to determine what questions the data warehouse is designed to answer. Thus, this is where the data quality analysis logically begins.

Determine Data Needed to Answer the Questions

In determining the questions that the warehouse should answer, it is critical to know what data will be needed to answer the questions. As an example, consider a system designed for a governmental agency that provides services to the homeless population. They want to know if the services that are provided have an affect on the homeless population. What is the recidivism rate for a given population given a specific mix of services? In order to answer this question, data is needed on length of stay (how long the individual stays in a shelter and how often they return to the shelter once they leave), what services are provided to the individual, and the demographics of the population. Identification of these data is the next step in the analysis of the quality of data.

Identify Candidate Data Sources

Once the needed data is known, the identification of available data sources begins. In a typical data-warehousing situation, the needed data may be available in multiple locations. Identifying candidate data sources encourages the warehouse developer to look “outside the box” and consider sources that may not at first appear to be relevant. This data may vary in quality. However, in the early phases of the framework, it is important to identify all of the available sources. A goal of this framework is to build a repository of meta-data that can be used to facilitate knowledge management. Capturing all of the available sources will yield a richer repository.

Identify Organizations

Identification of the organizations will be a natural extension of the previous step. In the example used earlier of the homeless system, data will be found in the agency as well as homeless shelter providers. In other situations, the data is intra-organizational. In this case the identification of organizations will be identification of departments or subdivisions. This phase should be customized to fit the warehouse being developed.

Identify Data Owners

A critical finding in the interviews was that the data providers must be involved in the process of building the warehouse. As the data was analyzed, exception reports were generated. Automated tools could identify discrepancies between what should be and what was, but it took

human intervention to determine how the data should be changed. For example, an automated tool could determine that an individual left a shelter without ever entering it by comparing the fields associated with admit and depart dates. However, human intervention was necessary to determine if the individual was ever in the system or not.

Identification of the data owners has multiple benefits. First, it involves the people who know the data best to be involved in the process, thus aiding the process of data cleansing. Secondly, the process of building a data warehouse involves integrating data from disparate sources. Previously, these data suppliers would not have had contact with each other. Thus, identification of data owners, and incorporation of this data into the meta-data repository, will allow a better view of the "big picture" of an organization or data-warehousing project.

Collect Source Documentation

Once the data owners are identified, they can then be asked to provide source documentation that will enable the developer to build the repository of meta-data. Source documentation can include, but is not limited to, data dictionaries, codebooks, lists of field names, and other documentation generated by the data suppliers. This source documentation provides the foundation for thorough understanding of the data available for the warehouse. In turn, a complete familiarity with the data will help the developer make better decisions regarding the appropriate actions to take as far as cleansing the data.

Create Data Quality Knowledge Management Tool

A critical component and focus of the DQAF is the Data Quality Knowledge Management (DQKM) tool. This tool, embodied in a relational database, is designed to be a repository for the meta-data associated with a data-warehousing project. The DQKM is designed to collect traditional meta-data, such as field size and type, as well as information about the source data organizations and the individuals who know and understand the source data. The DQKM grew out of the Enhanced Data Dictionary (EDD) that was developed with the DQAP. The EDD went beyond the traditional data dictionary and collected meta-data that is suitable for evaluating data to be included in a data warehouse. It provided information allowing the developer to look "across" databases and determine if there are homonyms (fields that have the same name but store different values, i.e. a field named counselor in two databases- one of the databases refers to the counselor for the patient, another database refers to the counselor that is in charge of the unit) and synonyms (fields that have different names but collect the same values, i.e. SSN and student number).

However, as the amount of meta-data to be collected grew, it became obvious that a flat file method of storing the meta-data was inefficient. Data is collected regarding organizations, data suppliers, business rules, context of use, quality dimensions and appropriate field values, as well as the standard data dictionary components of field size and type. See Figure 4 for the relational schema.

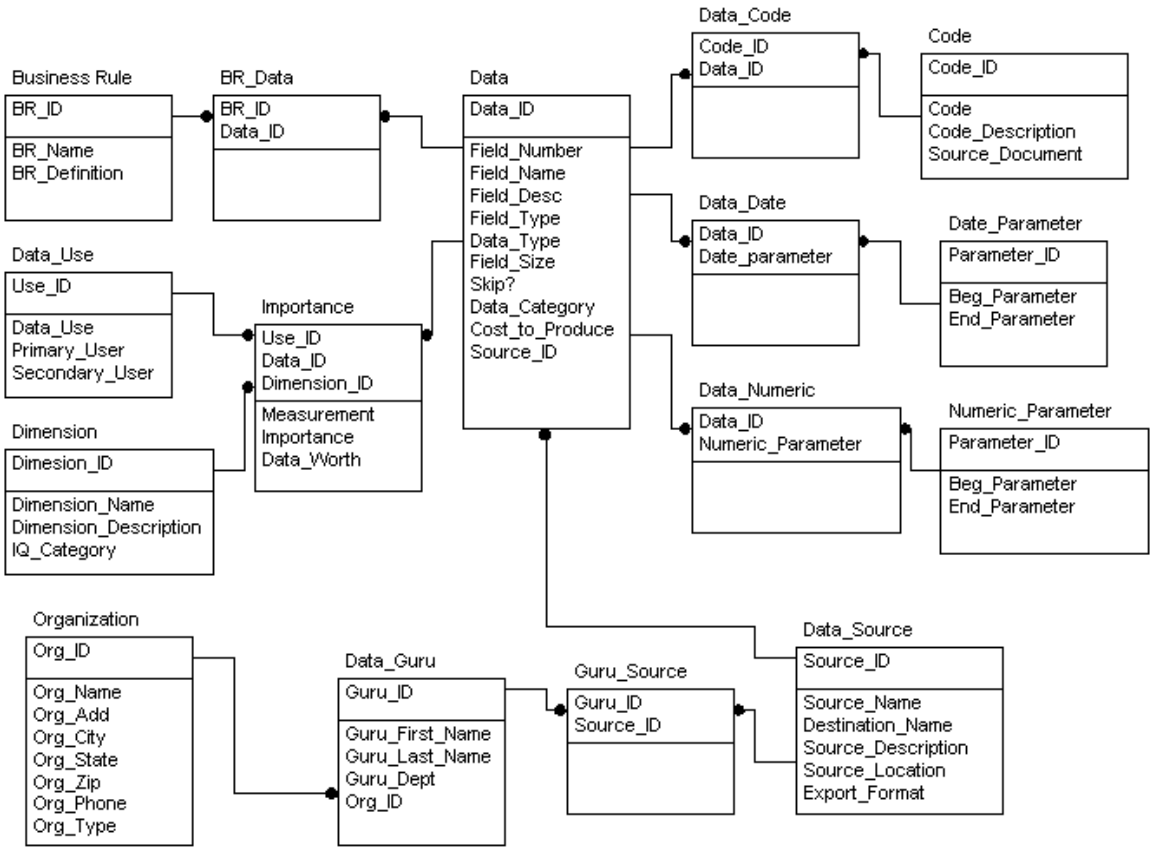

Figure 4- Schema for DQKM

The DQKM requires an enormous amount of effort to construct. Collecting and entering all of the meta-data is very time consuming. My previous experience with construction of the DQKM involved 40 person hours for roughly 5000 records. However, population of the DQKM is easily divided among a number of individuals, each with a different expertise. Thus, the task can be accomplished in a short time frame, as needed. Additionally, making decisions regarding the importance of data requires a great deal of expertise and judgment. The importance table fields are coded based on the data quality dimension being considered, as well as the use of the data. Thus, a particular field may be adequate for use 1 regarding accuracy, but not for use 2. Or a field may be adequate on completeness for use 3 but not on timeliness.

However, the result of all of this effort is a tool to facilitate knowledge networking. The meta-data that has been captured can be used in making decisions regarding the specific data fields on which to spend time and money for cleaning. A given warehouse development team may be composed of individuals who are experts as well as individuals who are novices. Once the DQKM has been constructed, the novices can make the decisions regarding cleansing, as this will be a SQL query. The knowledge is in the tool and can be extracted by novices. For example, a query on the data category will generate the available fields to answer a specific question. If the novices are provided with criteria for deciding the specific fields on which to expend cleansing resources, they can query the database and generate a listing of fields that meet the criteria. Thus, the decision as to exactly which fields should be cleaned has been automated.

Because the DQKM captures so much meta-data, it is also a repository that can be used to provide information when individuals who know the data leave an organization. Additionally, the wealth of information collected in the DQKM allows for a "bigger picture" look of the interrelationships among the data elements. Insights provided by this look at the data may provide new avenues to explore for competitive advantage. For example, querying the database on data category will determine what fields are available detailing a specific category such as gender or ethnicity. These fields will come from a variety of sources and analysis of these sources could alert management to redundancy in data collection efforts. Decisions could then be made as to which sources can be eliminated, thus reducing overall costs.

Visually Inspect Data

The DQKM is an evolving tool. Once the meta-data has been entered from source documents a visual inspection of the data in the sources will enable a greater understanding and clarification of the data. For example, in the absence of a data dictionary, field size and type can only be determined by a visual inspection of the data. The richest DQKM will be built utilizing all of the available resources.

Codify Business Rules

An essential element in determining what data is needed to populate a data warehouse and answer the necessary questions is a definition of the business rules. For example, in the homeless system, how is length of stay defined? Is it admit date to depart date for one shelter? Or is it the length of stay in the system, even if they move from shelter to shelter? Definition of the business rules, and the data needed to support them, is necessary for a well-defined data warehouse and will enrich the DQKM.

Utilize DQKM to Determine Which Fields to Clean

Once the DQKM has been fully populated it can be queried to determine where to focus the data cleansing efforts. After defining standards for cleansing, an analyst can query the database with the criteria set. For example, if it is determined that for use 1, accuracy scores greater than 65% should be cleaned further, a simple query will then determine what fields fall into that category. After determining which fields are candidates for cleansing, then the fields should be prioritized. Thus, a novice, using pre-defined criteria, can analyze and prioritize data fields that will make the best use of scarce resources and offer the greatest benefit to the project.

Determine Cleansing Method

Once it has been determined which fields should be cleaned and they have been prioritized, it is necessary to determine if the fields should be cleansed manually or automatically. Data cleansing tools are appropriate for fields that can be compared to another source, such as a list of codes, range of dates, or postal address lists (Neely 1998). By querying the database, it can be determined if these comparative values are available and what the values are. Conversely, if no values are available for comparison then the data must be cleansed

manually and appropriate steps should be taken to ensure that the data is returned to the data supplier to verify the data.

Further Research

A development team for a data warehouse will consist of both experts and novices. The framework will be used by the team as a whole in the development of the warehouse. However, experts and novices will perform different tasks in the accomplishment of the goal of determining where to focus cleansing efforts.

The framework and tool described in this paper are being tested in a three-phase approach. The first phase consisted of the pilot tests conducted using the DQAP framework and EDD. This phase provided data to construct the DQAF and DQKM. Phase two was conducted after construction of the framework and tool. This phase involved testing the ability of novices to determine which data fields should be cleaned given a completed DQKM. Phase three will be a modification of the second phase, and will involve prioritizing the data fields for cleansing in addition to the identification of them. The goal of the second and third phases is to show that novices can use the DQKM to make decisions regarding the cleansing of data.

Future research will involve more extensive testing of the DQKM. At this point only the portion related to importance has been tested. Testing regarding the ability to determine whether data should be cleaned manually or automatically needs to be done, as well as testing the framework in it's entirety.

References

Forgionne, G. and B. Rubenstein-Montano (1999). "Post Data Mining Analysis for Decision Support through Econometrics." Information, Knowledge and Systems Management **1**(2): 145-157.

Goodhue, D. L. and M. D. Wybo (1992). "The Impact of Data Integration on the Costs and Benefits of Information Systems." MIS Quarterly **16**(3): 293-311.

Neely, M. P. (1998). Data Quality Tools for Data Warehousing- A Small Sample Survey. The Conference on Information Quality, Cambridge, MA.

Storey, V. C. and R. Y. Wang (1998). Modeling Quality Requirements in Conceptual Database Design. 1998 Conference on Information Quality, Cambridge, MA.

Tayi, G. K. and D. P. Ballou (1998). "Examining Data Quality." Communications of the ACM **41**(2): 54-57.

Wang, R. Y. and D. M. Strong (1996). "Beyond Accuracy: What Data Quality Means to Data Consumers." Journal of Management Information Systems (JMIS) **12**(4): 5-34.

Wixom, B. H. and H. J. Watson (2001). "An Empirical Investigation of the Factors Affecting Data Warehousing Success." MIS Quarterly **25**(1): 17-38.