Journal of Database Management, 15(1), 87-103, Jan-Mar 2004 87

RESEARCH NOTE

Process-Embedded Data Integrity

Yang W. Lee, Northeastern University, USA
Leo Pipino, University of Massachusetts, Lowell, USA
Diane M. Strong, Worcester Polytechnic Institute, USA
Richard Y. Wang, Massachusetts Institute of Technology, USA

ABSTRACT

Despite the established theory and the history of the practical use of integrity rules, data quality problems, which should be solvable using data integrity rules, persist in organizations. One effective mechanism to solve this problem is to embed data integrity in a continuous data quality improvement process. The result is an iterative data quality improvement process as data integrity rules are defined, violations of these rules are measured and analyzed, and then the rules are redefined to reflect the dynamic and global context of business process changes. Using action research, we study a global manufacturing company that applied these ideas for improving data quality as it built a global data warehouse. This research merges data integrity theory with management theories about quality improvement using a data quality lens, and it demonstrates the usefulness of the combined theory for data quality improvement.

Keywords: data quality; data integrity; action research; data warehouse

INTRODUCTION

Data integrity rules are the recommended approach for enforcing data quality in operational databases. These rules are well-grounded in relational database theory (Codd, 1970, 1990) and are widely used. Despite established theory and history of the practical use of integrity rules, data quality problems persist in organizations (Becker, 1998; Brodie, 1980; Preston, 2001; Price, 1994; Tayi & Ballou, 1999; Segev, 1996; Huang et al. 1999).

The failure to link integrity rules to organizational changes is among the reasons that data quality problems plague organizations. In essence, conventional practice is to view the application of data integrity as a one-time static process applied when data enters the database. We pro-

pose that the application of data integrity be viewed as a dynamic, continuous process, embedded in an overall data quality improvement process. We develop such an embedded data integrity process and demonstrate its usefulness to a real organization.

Similar issues are apparent in data warehouses. A common difficulty with data warehousing is the poor quality of data in operational databases, the sources of data for data warehouses (Ballou & Tayi, 1989; Celko & McDonald, 1995). If poor-quality data enter the data warehouse, they can impede global access to information and knowledge (Aho, 1996). Similarly, software tools used to achieve data quality (Brown, 1997; CRG, 1997, 1998) should be applied within a coherent and systematic improvement process, but are typically applied in an ad hoc fashion.

A key challenge, then, is to understand how data integrity principles can be embedded in an ongoing continuous data-quality improvement process. While the literature provides principles of data quality improvement, data integrity, and software tools, we can only fully understand their appropriate use through the dynamic instantiation in an organization.

PROCESS-EMBEDDED DATA INTEGRITY

Total Data Quality Management (TDQM) Cycle

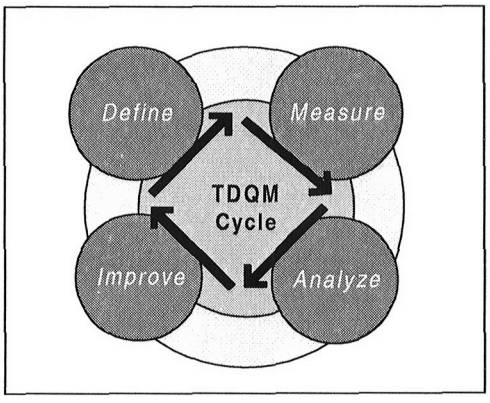
Total Quality Management (TQM), a practical approach for improving quality (Deming, 1986; Juran & Godfrey, 1999), explicitly links quality to a continuous improvement process. The Total Data Quality Management (TDQM) Cycle (Madnick & Wang, 1992), an adaptation of TQM principles to the context of data, consists of defining, measuring, analyzing, and improving data quality through multiple, continuous improvement cycles (Figure 1).

First, we *define* data quality as data that are fit for use (Strong et al., 1997), following the quality definition in the TQM literature. To data consumers, fitness for use means data that are accurate, believable, objective, relevant, timely, reputable, value-added, appropriate in amount, concisely represented, consistently represented, complete, interpretable, accessible, understandable, and secure (Wang & Strong, 1996). Typically, in a data quality improvement project, a subset of these dimensions is chosen.

Second, we *measure* data quality along the chosen dimensions using metrics (Pipino et al., 2002). Consider accuracy, which means that stored data values do not differ from "true" data values. Since determining true values is costly, an appropriate accuracy metric is a value in an appropriate range. Iteration between the definition and measurement steps serves to develop metrics that are measurable at a reasonable cost and useful for improvement activities.

Third, *analyze* is interpreting the measures and deciding whether and how to improve the quality of data. Measures of

Figure 1: The TDQM Cycle



data quality, e.g., percentage of values outside the range, provide the foundation for investigating problematic values and determining root causes of out-of-specification values. Decisions can then be made as to whether the quality definition was inappropriate or corrective quality improvement actions are needed.

Fourth, *improvement* is actions to change data values directly or, more appropriately, change the processes that generate data. If processes are not changed, they will continue to produce poor quality data.

After the fourth step, the TDQM cycle repeats as improvement activities produce results and as the definition and metrics for data quality are refined. "Continuous improvement involves incremental improvement when appropriate and discontinuous improvement when required" (Juran & Godferey, 1999: 25). It implies strategic planning and judicious and proactive scanning of changes in environment, emerging technology, market, and customers' needs. For data quality, data integrity rules must be reviewed, redefined, and implemented, producing a continuous improvement strategy based on an iterative process.

Data Integrity and its Problems

Data integrity is defined as entity, referential, domain, column, and user-defined integrity rules by Codd (1970, 1990) (Table 1). Within a database or data warehouse, data quality dimensions are operationalized as data integrity rules, automatically enforced by relational databases. While data integrity rules cannot capture some data

Table 1: Data Integrity Constraints Revisited (adapted from Date, 1990; Rob. & Coronel, 2000)

Integrity Category	**Constraints**	**Objectives**
Entity integrity	Entity integrity rules state that all entries are unique, and no part of a primary key may be null.	This rule guarantees that each entity will have a unique identity.
Referential integrity	A foreign key may have either an entry that matches the primary key value in a table to which it is related or a null entry – as long as it is not a part of its table's primary key.	Referential integrity guarantees that every non-null foreign key value must reference an existing primary key value. The enforcement of the referential integrity rule makes it impossible to delete a row in one table whose primary key has matching foreign key values in another table.
Domain integrity	All values of an attribute must be drawn from a specified domain.	The enforcement of the domain integrity rule makes it impossible to enter a value not from the specified domain. This, however, does not ensure that the entries are correct values.
Column integrity	All values of an attribute must be drawn from a specified range within the domain.	The enforcement of the column integrity rule makes it impossible to enter a value not from the specified range of values. Like domain integrity, this does not ensure that the entries have correct values.
User-defined integrity	Captures business rules that restrict values beyond the restrictions from the above integrity rules. User-defined integrity rules may cross multiple columns, possibly in different tables.	The enforcement of user-defined integrity makes it impossible to enter values that do not conform to the rules of a particular business.

quality concerns, e.g., believability and interpretability, they can capture many, producing significant improvements for data consumers.

These constraint-based rules in databases help ensure that data represents the real-world states (Wand & Wang, 1996). The real-world state, however, is dynamic, not static, changing over time. In addition, real-world states have become increasingly global as businesses expand from domestic to global operations.

The dynamic nature of data integrity is illustrated by a commercial bank building a relational database from its customer flat file. Within this project, data integrity rules were tested, e.g., the column integrity rule on gender "all values must be M or F" produced about half the records in violation. An analysis found two consistent patterns: many missing values and "C" values. Root cause investigation with users and data entry people yielded a discovery that C is valid, meaning a corporation. Over time, the business situation required a new category and users added "C" in a de facto fashion, acceptable given the lack of controls in their flat file environment. As a result, the bank redefined the gender integrity rule.

A second example illustrates the global nature of data integrity, i.e., acceptable local data integrity rules may not be sufficient from a global perspective. When building a global customer data warehouse from its divisional databases, this same bank tested entity integrity on customers. Locally, entity integrity on each database was good. When the same data were combined, violations occurred because the same customer had two different IDs, or two different customers had the same ID. A root cause analysis found that locally assigned customer IDs, when combined for global use, created violations.

Embedding Data Integrity Rules in the TDQM Cycle

To ensure data integrity rules reflect the dynamic, global nature of real-world states, organizations need a process that guides the mapping of changing real-world states into redefined data integrity rules. Our solution is embedding data integrity rules into the TDQM process, producing the following steps for improving data quality. An organization first *defines* what data quality means for their data and context (Strong et al., 1997), producing data integrity rules. Next an organization *measures* the quality of data against these integrity rules. Measurement may involve simple metrics, e.g., the percentage of violations, or more elaborate metrics, e.g., the difference between the data and the defined data quality standard. Third, the underlying cause of violations are *analyzed*, producing a plan to *improve* the quality of data to conform to data integrity rules. In addition, data integrity rules are *redefined* when violations are actually valid data. This redefinition operation is of the utmost importance for the continuous improvement of data quality and makes the process more than simply an iterative process.

The result is process-embedded data integrity, in which the use of data integrity tools are explicitly linked to organizational processes for improving data quality in the context of global and dynamic data changes. It serves as a potent means of making the organization's data integrity rules more visible and identifiable. It promotes reflection on the rules and facilitates their communication throughout the organization. The consequence is support for the dynamic and global nature of organizational data.

RESEARCH METHOD

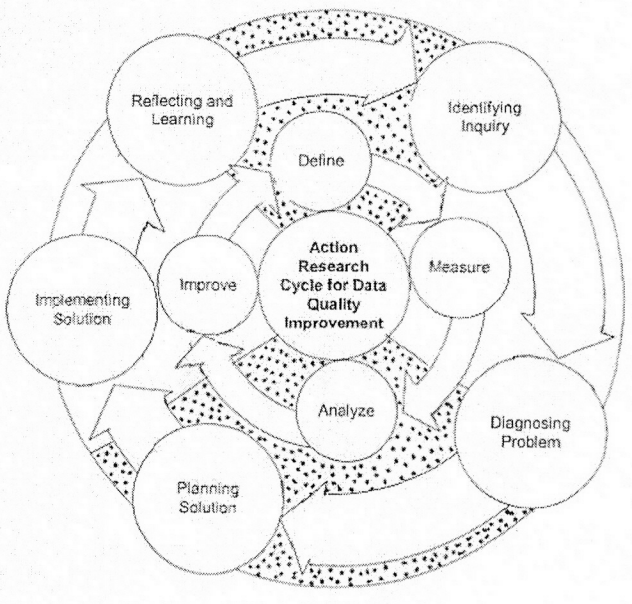
We use action research to study the applicability of process-embedded data integrity in a global manufacturing company, Glocom. The essence of action research lies in its objectives of both advancing theories in research as well as facilitating organizational changes (Baskerville, 2001; Schein, 1987; Argyris & Schön, 1978). Action research differs from consultancy in its aim to contribute to both theory and practice. It differs from the case method in its objective to intervene, not simply to observe. Action research is empirical, yet interpretive; experimental, yet multivariate; observational, yet interventionist (Baskerville & Wood-Harper, 1996). Action research methods, therefore, place IS researchers in a "helping-role" within organizations (Baskerville & Wood-Harper, 1998), requiring longer-term involvement with the situation and subjects (Mumford, 2001). Figure 2 shows how IS researchers, using an action research cycle, contribute to improving data quality in organizations. In short, action research merges research and practice, which in turn produces significantly relevant IS research findings (Keen, 1991).

Research Setting

Glocom, a consumer goods manufacturing company with subsidiaries in 52 countries, produces health, cleansing, and household products. Its corporate culture is friendly, international, and non-confrontational. Glocom's information technology infrastructure is advanced, as it typically adopts new IS practices and technology early. It has up-to-date database systems and has been experimenting with data quality improvement projects for several years.

Our research focused on the data quality project for the global sales data warehouse, which started in 1998 and lasted for two years. Continuous improvement of data quality is on-going. We focused on its sales and materials procure-

Figure 2: Action Research Cycle for Data Quality Improvement



ment because we were interested in changes in database practice due to business policy changes, that is, from a multinational business to a global business policy.

Research Process

We followed the guidelines for conducting action research, which involves two key principles designed to increase the likelihood of advancing both practice and theory (Baskerville, 2001; Mumford, 2001; Susman, 1983; Elliot, 1982; Kemmis, 1982; Brown, 1982; Winter, 1989). One is to study the intended change in a real organizational setting.

The other is to follow an iterative research process. Our research process involves five primary activities, which dovetail with the TDQM data quality improvement cycle as shown in Figure 2. *Identifying inquiry* involves defining what data quality and data integrity means for Glocom. *Diagnosing problems* includes measuring and analyzing data integrity. *Planning solutions* involves analyzing root causes and devising solutions. *Implementing solutions* entails specific improvements. *Reflecting and learning* involves thinking about the meaning of data integrity violations. It also involves specifying learning by members of Glocom to raise the competency for understanding business processes in different contexts, e.g., how current processes are expected to differ from future processes and how global business practices are expected to differ from local ones.

We worked with Glocom from the start of its data warehouse data quality improvement project. The collaborative team included data analysts, data architects, IS managers, and programmers from Glocom, and researchers from several universities. Interventions by researchers were grounded in the principles of database integrity and data quality, with an emphasis on embedding data integrity analysis into a process. Specifically, the action researchers intervened by suggesting the technology and processes to be adapted, facilitating the use of certain data quality and data integrity improvement methods, collecting research data for analysis, and offering feedback to the participants periodically. The Glocom participants were responsible for designing and implementing a high-quality data warehouse to support global analysis and decision-making. The Chief Data Architect was the focal point for communications between the research team and Glocom practitioners.

Data Collection and Analysis

We observed Glocom's practice paying special attention to how it defined and redefined data integrity in the data warehousing project. We offered training sessions on data quality and discussed ways to improve data integrity. Documents collected included periodic data quality assessment and auditing results. Glocom used a data integrity software package, *Integrity Analyzer* (CRG, 1998), to audit data integrity in its databases. The *Integrity Analyzer* embeds the four steps of the data quality improvement process (define, measure, analyze, improve) to analyze violations of data integrity rules within an overall data quality improvement process. See the Appendix for further information on the *Integrity Analyzer*.

We mitigated two typical risks of conducting action research (Baskerville, 2001). To mitigate the possible timeframe conflict between practice and research, our research timeline was dictated by meaningful change and improvement at Glocom. To mitigate the subjectivity that authors can

bring to action research due to their involvement in the change process, the two authors not directly involved in the intervention process served as objective observers of the action research.

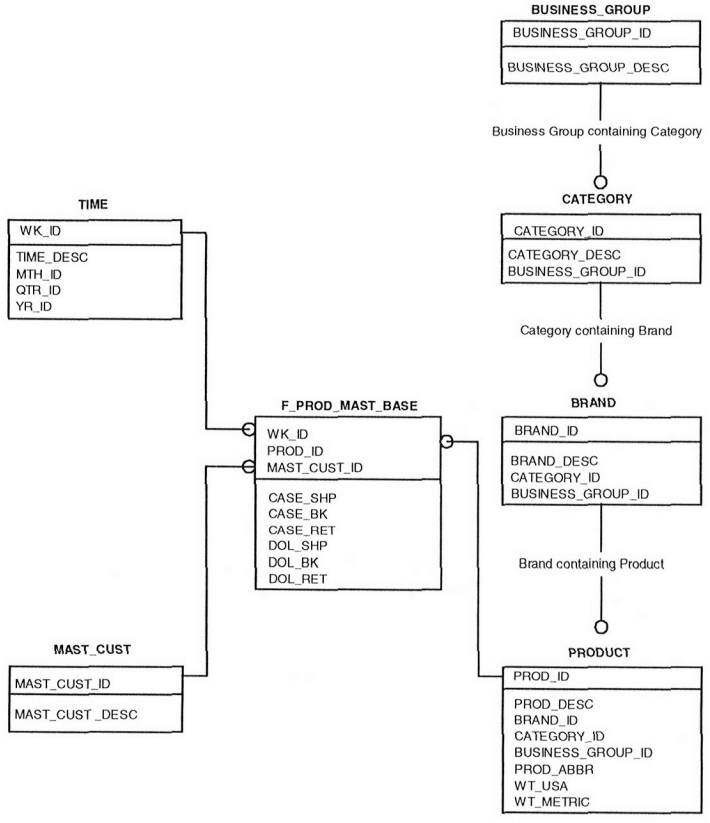
PROCESS-EMBEDDED DATA INTEGRITY AT GLOCOM

Glocom's operating units in 52 countries managed their sales data independently, but Glocom needed consolidated global shipping and booking data for understanding sales patterns for global procurement. Before loading its new global sales data warehouse, Glocom's two million global shipping and booking records were examined for data quality. Data quality problems were fixed in the source transaction files. Furthermore, improvement processes, including better data collection, more coordination across previously independent units, and new or revised data integrity constraints, were initiated to fix underlying problems.

The data warehouse subset used for data quality analysis, Figure 3, forms a standard data warehouse star schema design. It includes one fact table, F_PROD_MAST_BASE, of aggregate quantities and dollar values for product shipments, bookings of product for future shipment, and product returns, and three dimension tables, TIME, MAST_CUST, and PRODUCT.

Each example offered of Glocom's evaluation of integrity constraints reflects one or more data quality dimensions. For example, in checking column integrity the analysts were assessing the dimensions of

Figure 3: Star Schema for the Sample Data Warehouse at Glocom



accuracy and *interpretability*; in checking entity integrity, the dimensions of *accuracy* and *completeness*; and for user-defined integrity, the dimension of *consistency* across data elements.

Process-Embedded Column Integrity Cycles

Glocom's first iterations of the TDQM cycle focused on column integrity. Integrity violations were expected since these column integrity rules had not been consistently applied in the legacy systems. We present one example of their many column integrity cycles in terms of the five action research activities (Table 2).

Identifying Inquiry. The column integrity rule for cases booked, non-negative values, was tested using the *Integrity Analyzer* (IA) software to measure violations, producing the sample results in Figure 4.

Figure 4: Sample Column Integrity Results

	CASE_BK	DOL_BK	AST_CUST_ID	PROD_ID	WK_ID
▶	-17	0	69000	90095	199841
	-7	0	69000	90096	199841
	-6	0	155450	1943	199738
	-1	0	348000	90097	199643
	-12	0	465600	90095	199633
	-49	0	541800	90095	199710

Table 2: A Process-embedded Column Integrity Example

	Actions
Identifying Problem	• For shipment data, "case_bk" column, acceptable values should not be negative. (***Define***) • Check shipment data, "case_bk" column, for values that are less than zero. (***Measure***)
Diagnosing Problem	• Review of violation records shows negative values for number of cases booked (See Figure 4). (***Analyze***) • The data quality manager communicated with sales managers to determine the validity of negative bookings.(***Analyze***)
Planning Solution	• The sales managers confirmed that the negative values represent cancelled bookings. (***Analyze***) • The data quality manager and the sales managers jointly decided, "this is confusing". (***Analyze***) • The data quality manager with approval from the sales managers decided to add a new field for cancelled bookings. (***Analyze***)
Implementing Solution	• A new field for cancelled bookings, 'case cnl" column was added to the shipment data. (***Improve***) • The cancelled bookings were removed from the "case bk" column and added to the new field. (***Improve***)
Reflecting and Learning	• The data quality manager learned to use the process-embedded data integrity tool to support problem solving. • The sales managers learned to understand their processes for exceptional cases and communicate this explicitly with the data quality manager. • Use of an explicit iterative process for column integrity improvement facilitated communications between the data quality manager and the sales managers.

The inquiry produced the discovery of negative values in the CASE_BK (cases booked) field.

Diagnosing Problem. The analyst examining these violations took them to the data quality manager for interpretation. From experience, the data quality manager surmised a possible legitimate meaning for the negative values, cancellations, that is, negative bookings. He discussed and confirmed this interpretation with sales managers.

Planning Solution. Using this diagnosis, the analyst and managers searched for a solution that would capture cancellations, but not interfere with the standard interpretation of cases booked. They decided to create a new field for cancellations and remove these data from the "cases booked" field.

Implementing Solution. This change was reflected at all relevant data levels, i.e., the operational database as well as the data warehouse.

Reflecting and Learning. Using an explicit iterative process for column integrity improvement facilitated communications between the data quality manager and the sales managers, which facilitated further column integrity improvement cycles. The data quality manager also learned to use the process-embedded data integrity tool to add new constraints and modify constraints as new requirements developed. For example, to detect new values created by sales, he programmed additional user-defined rules into the tool. Over time the sales managers learned the processes for exceptional cases and communicated directly with the data quality manager.

This example illustrates a common problem in legacy data over time: fields are used for purposes other than originally intended. The dynamic nature of data drives this behavior. In this example, column integrity for the cases booked remained as initially defined, and a new field was added to represent cancelled bookings. In other cases, the column integrity definition may be adjusted to fit the revised field use. With automatically enforced integrity in relational databases, if a salesperson needed to enter data not within range, such as negative bookings, he/she should follow a standard procedure of requesting a database change. Organizations should be aware, however, that integrity constraints can be misused in relational DBMSs, e.g., by recording 'cancellation' in a comment field and using a positive value. Thus, regular monitoring of column integrity is still necessary.

Process-Embedded Entity Integrity Cycles

After completing column integrity improvement cycles for each column, the analyst focused on primary key columns to verify entity integrity.

Identifying Inquiry. In F_PROD_MAST_BASE, the analyst verified that the three fields, WK_ID, PROD_ID, and MAST_CUST_ID, together formed a unique key. In most cases, entity integrity analysis showed no violations (Table 3). An entity integrity analysis on the 66 tables in the overall global data warehouse produced seven tables (10.6 %) with violations.

Diagnosing Problem. In the overall global data warehouse, an analysis revealed multiple key values for what appeared to be the same products and same customers, i.e., customers who purchased from more than one operating unit. This was caused by the integration of independently assigned customer IDs and product IDs.

Planning Solution In parallel with data warehouse development, Glocom was standardizing global product and customer IDs. All entity integrity violations were re-

Table 3: Sample Entity Integrity Results

Tables	Primary Key(s)	Records	Not Unique	Number Null
BUSINESS_GROUP	BUSINESS_GROUP_ID	3	0	0
CATEGORY	CATEGORY_ID	22	0	0
BRAND	BRAND_ID	102	0	0
PRODUCT	PROD_ID	1825	0	0
TIME	WK_ID	208	0	0
MAST_CUST	MAST_CUST_ID	75	0	0
F_PROD_MAST_BASE	WK_ID, PROD_ID, MAST_CUST_ID	21,352,587	0	0

ferred to the manager working with operating units towards standardized product and customer IDs.

Implementing Solution For some tables, one or more fields were added to the initial primary key to achieve record uniqueness. For others, some values for keys were changed.

Reflecting and Learning. The continuous process of embedded data integrity supported by the tool explicitly guided the analyst in taking a global view of data. This facilitated further cognition of the need for coordinated global assignment of IDs. As with column integrity, continuously monitoring the global assignment process is necessary.

This example illustrates the importance of a global view in data integrity. Entity integrity was excellent (0% violations) when operating unit data were analyzed independently, but poor (10% violations) when multiple units were analyzed together.

The analyst also checked referential integrity and tracked compliance over time. There were no remaining referential integrity violations after correcting column and entity integrity violations.

Process-Embedded User-Defined Integrity Cycles

Beyond conventional column, entity, and referential integrity rules, data in databases must conform to business rules, called user-defined integrity rules (Codd, 1990).

Identifying Inquiry. One business rule at Glocom is that if the number of cases booked is greater than zero, then the dollars booked should also be greater than zero. The software tool showed 839 fact table records in violation of this user-defined integrity rule (Figure 5).

Diagnosing Problem. The analyst discovered that these violations represent product samples, i.e., products shipped, but not billed, to customers.

Planning Solution. The analyst was deciding how best to store such information so that legitimate samples are not treated as integrity violations and do not

Figure 5: Sample User-defined Integrity Violation Results

	Condition	Date	Time	# Met
▶	f_prod_mast_base case_bk >0 and dol_bk=0	8/13/98	12:39:29 PM	839

lead to incorrect decisions. The sales managers, however, said that such data represented standard business practice and was not confusing.

Implementing Solution. This user-defined integrity rule was dropped since it did not reflect current business practice.

Reflecting and Learning. In this example, knowledge of current special situations was useful for guiding data integrity analysis and interpreting results. This made the concept of understanding the uses and interpretations of data values more visible to the analyst. He also learned to communicate earlier with the sales manager before attempting to solve an apparent problem.

Identifying Inquiry. Another user-defined integrity rule illustrates the need for a dynamic view of data integrity. No product should be shipped to a major customer that filed for bankruptcy.

Diagnosing Problem. The values in F_PROD_MAST_BASE should be zero for CASE_SHP and DOL_SHP to this customer (cases and dollars shipped). Since non-zero CASE_SHP and DOL_SHP are technically valid, Glocom does not have a data integrity rule for this, but relies on domain experts to handle it. In this situation the process worked; the data quality manager verified that no shipments were made to this customer since bankruptcy.

Planning Solution. They decided there were sufficient available solutions, e.g., a customer credit hold, once a domain expert identifies the problem.

Implementing Solution. There was no need to change user-defined data integrity rules.

Reflecting and Learning. The analyst learned that knowledge of past problems held by domain experts was useful for guiding data integrity analysis and interpreting results. The use of software tools like the IA which store problem history exemplifies process-embedded tool use. As these examples illustrate, domain experts are often needed to determine whether data values make sense in changing global business situations.

Glocom recognized that good data integrity practice is part of good data quality practice. Integrity improves many dimensions of data quality, and process-embedded data integrity ensures that organizational data reflect changes in global business. As a result, assessing and assuring data integrity now is part of Glocom's institutionalized data quality initiatives.

LESSONS LEARNED FROM GLOCOM

Glocom subscribed to the view that data should represent the way it does business: globally and dynamically. It took its data warehousing project as an opportunity to improve and learn about its data quality and integrity. It used the process-embedded data integrity approach as a foundation for communicating with data users, reviewing data integrity rules, and utilizing its data practice history. Glocom employed the *Integrity Analyzer* to discover new problems and to check conformance to defined data integrity rules. Its routine repetitive uses became institutionalized after its data warehouse project because it realized the need to continuously monitor data integrity constraints.

Glocom redefines its data integrity rules when needed to synchronize them with its changing global business, a key factor in its continuous improvement effort. Glocom is moving from its two percent data integrity error rate toward its goal of six-sigma. Its use of the process-embedded data integrity approach and an associated software tool was instrumental in this improvement.

IMPLICATIONS FOR RESEARCH AND PRACTICE

To provide solutions to data quality problems, researchers, practitioners, and tool vendors must work together to develop processes and tools for managing data and metadata in dynamic and global environments.

Database researchers must ensure that solutions to data quality problems are integrated into an overall data improvement process (Storey & Wang, 1998; Wang et al., 1995). Some research reflects the dynamic and global nature of data, e.g., the Evolution and Change in Data Management Workshop (Roddick et al., 2000), database schema evolution (Liu et al., 1994), and information integration research (Jhingran et al., 2002; Batini et al., 1986; Ram & Ramesh, 1997; Och et al,, 2000). These solutions could address integrity problems, but some human knowledge or additional constraints are still needed.

Practitioners must proactively update data integrity rules as data fields evolve and businesses change. They should incrementally integrate data and build data warehouses through well-defined processes (Trisolini et al., 1999). Overall, practitioners should manage information as a product, monitor the changing business environment, and initiate improvement projects to ensure the information product meets data consumers' needs (Wang et al., 1998).

Vendors are beginning to provide process-embedded tools (Allen, 1996; Brown, 1997; Sogini, 2003; Boldwin, 2000; Garvey, 1999; Anthes, 2002; Meehan, 2002). Used properly, a process-embedded data integrity tool, such as *Integrity Analyzer*, reveals violations of data integrity rules, promotes communication among analysts, managers, and domain experts, and facilitates the redefinition of data integrity. Tools that automatically "clean" data may improve quality only temporarily because they do not promote data consumers' understanding of their data. Visibility and communication of problems are key to the continuous cycle of re-defining, re-measuring, and re-analyzing.

CONCLUSION

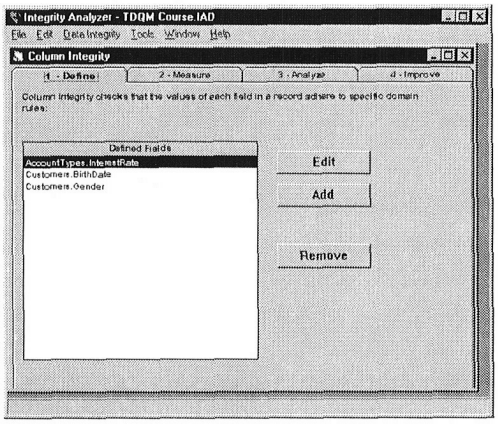
The major theoretical advance in this research lies in redefining data integrity theory to link it to changing business processes, by merging computer science with management theories. Data integrity theory as defined by Codd works well in a static, local database environment. To work effectively in a dynamic, global environment, data integrity principles must be embedded in a continuous improvement process as we propose. With such an embedded process, Glocom ensured that data integrity rules reflected business changes, resulting in data warehouse data truly fit for use for global business decisions in a dynamic environment.

APPENDIX: OVERVIEW OF THE INTEGRITY ANALYZER (IA)

Core Functionality

The IA's core functionality is a realization of the TDQM Cycle applied to data integrity (CRG, 1998). To access the core functionality, use the pull-down menu labeled Data Integrity from the starting IA screen. The menu items are Entity, Referential, Column, and User-Defined Integrity. Upon selection of an integrity type, the IA generates a screen with tabs labeled *Define*, *Measure*, *Analyze*, and *Improve* (Figure A1), which are the four components of the TDQM Cycle.

Figure A1: IA with Column Integrity Selected from Data Integrity Menu



For each integrity type, the *Define* tab provides menus and list boxes for specifying the integrity constraint. The *Measure* tab selects a defined constraint to be measured and measures the number of violations. The *Analyze* tab displays constraint violation statistics. The *Improve* tab reports the record instances that are violations. When the IA is connected in real time, users can correct the records highlighted, provided write access is permitted. The user can select among define, measure, analyze, and improve in any sequence, provided that appropriate definitions and measurements are performed before analysis and improvement, and can choose the integrity functions in any order desired.

IA Utilities

The Tools pull-down menu provides three utilities–frequency check, data set builder, and table browser–to facilitate the analysis of data integrity problems. Although integrity constraints cover these problems, an IA user may not know the exact constraint to specify. The utilities facilitate searching for problems in an exploratory manner, but do not change database values (Table A1).

Managing a DQ Project

The IA facilitates the management of a data integrity improvement project by storing measurement results over time, and graphically reporting these multiple measurements to support the comparison of new and old DQ measurements. This functionality is implemented in three parts:

1) A project file stores measures across instantiations of the IA.
2) For each measurement, the IA stores the results with a date and time stamp and the identifier of the constraint being measured in the project file.
3) For each analysis, the user can select, by date and time, any or all stored measurements. If multiple time periods are selected, the IA displays the changes in

Table A1: IA Utilities

	Frequency Check	Data Set Builder	Table Browser
Define	Specify which column to check	Select fields to include in a data set	Not Applicable
Measure	For columns specified, compute the frequency of occurrence of each value	Not Applicable	Not Applicable
Analyze	Examine the frequency statistics	Examine the selected data	Examine the data in the database tables
Improve	Not Applicable	Not Applicable	Not Applicable

data integrity violations over time.

The project file also stores information about the structure of the source database being assessed, including the names of tables and their attributes. For example, when the IA first attaches to a database, the base domain of each field, e.g., text or numeric, is defined. If the structure of the database changes, not unusual during data integrity improvement, the Update Table function in the File pull-down menu updates this metadata.

REFERENCES

Aho, A. V. (1996). Accessing Information from Globally Distributed Knowledge Repositories. Paper presented at the *Symposium on Principles of Database Systems (PODS)*, Montreal Quebec, Canada.

Allen, S. (1996). Name and Address Data Quality, Paper presented at the 1996 *Conference on Information Quality*, Cambridge, MA, 242-255.

Anthes, G. (2002, October 14). Bridging Data Islands. *Computerworld,* 23-24.

Argyris, C., & Schön, D. A. (1978). *Organizational Learning: A Theory of Action Perspective*. Reading, MA: Addison-Wesley Publishing Co.

Ballou, D. P., & Tayi, G. K. (1989). Methodology for Allocating Resources for Data Quality Enhancement. *Communications of the ACM, 32*(3), 320-329.

Baskerville, R. (2001). Conducting Action Research: High Risk and High Reward in Theory and Practice, *Qualitative Research in IS: Issues and Trends* (pp. 192-217). Hershey: Idea Group Publishing.

Baskerville, R.& Wood-Harper, A.T. (1996). A critical perspective on action research as a method for information systems research. *Journal of Information Technology, 11*, 235-246.

Baskerville, R.& Wood-Harper, A.T. (1998). Diversity in information systems action research methods. *European Journal of Information Systems, 7*, 90-107.

Batini, C., Lenzirini, M., & Navathe, S. (1986). A comparative analysis of methodologies for database schema integration. *ACM Computing Survey, 18*(4), 323-364.

Becker, S. (1998). A Practical Perspective on Data Quality Issues. *Journal of Database Management, 35*(Winter), 35-37.

Boldwin, C. (2000, November 27). Data Integration Software Speeds Web. *Network World,* 59.

Brodie, M. L. (1980). Data Quality in Information Systems. *Information and Management*(3), 245-258.

Brown, L. (1982). Action-research: Notes on the national seminar, *Classroom Action-Research Network Bulletin* (Vol. 5). Norwich: University of East Anglia.

Brown, S. M. (1997, October). Preparing Data for the Data Warehouse. Paper presented at the 1997 *Conference on Information Quality*, Cambridge, MA, 291-298.

Celko, J., & McDonald, J. (1995). Don't Warehouse Dirty Data. *Datamation,* October, 15th.

Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM, 13*(6), 377-387.

Codd, E. F. (1990). *The Relational Model for Database Management: Version 2*. Reading, MA: Addison-Wesley.

CRG. (1997). *Information Quality Assessment (IQA) Software Tool*. Cambridge, MA: Cambridge Research Group.

CRG. (1998). *Integrity Analyzer: Getting Started*. Cambridge, MA: Cambridge Research Group.

Date, C. J. (1990). *An Introduction*

to Database Systems (5th ed.). Reading: Addison-Wesley.

Deming, E. W. (1986). *Out of the Crisis*. Cambridge, MA: Center for Advanced Engineering Study, MIT.

Elliot, J. (1982). Action -research: A framework for self-evaluation in schools, *Teacher-Pupil Interaction and the Quality of Learning* (Mimeo ed., Vol. 1). London: London Schools Council.

Garvey, M. (1999, August 9). Easy Data Integration. *Information Week,* p. 142.

Huang, K., Lee, Y., & Wang, R. (1999). *Quality Information and Knowledge*. Upper Saddle River: N.J.: Prentice Hall.

Jhingran, A., Mattos, N., & Pirahesh, H. (2002). Information Integration: A Research Agenda. *IBM Systems Journal,* Vol. 41(No. 4).

Juran, J., & Godferey, A. B. (1999). *Juran's Quality Handbook* (Fifth Edition). New York: McGraw-Hill.

Keen, P. (1991). Relevance and rigor in information systems research. In R. Hirschheim, Klein, H., Nissen, H.E. (Ed.), *Information Systems Research: Contemporary Approaches and Emergent Traditions* (pp. 27-50). Amsterdam: North-Holland.

Kemmis, S. (1982). *The Action-Research Planner* (2nd ed.). Victoria, Deakin University Press.

Liu, C. T., Chrysanthis, P. K., & Chang, S. K. (1994). Database Schema Evolution through the Specification and Maintenance of Changes on Entities and Relationships. *Proceedings of the 13th International Conference on Entity-Relationship Approach (ER '94).*

Madnick, S., & Wang, R. Y. (1992). *Introduction to Total Data Quality Management (TDQM) Research Program* (TDQM-92-01): Total Data Quality Management Program, MIT Sloan School of Management.

Meehan, M. (2002). Data's Tower of Babel. *Computerworld,* (April 15), 40-41.

Mumford, E. (2001). Action Research: Helping Organizations to Change, *Quality Research in IS: Issues and Trends,* Hershey: Idea Group Publishing, 46-77.

Och, C., King, R., & Osborne, R. (2000, March 19-21). Integrating Heterogeneous Data Sources using the COIL Mediator Definition Language. Paper presented at the *Symposium on Applied Computing*, Como, Italy.

Pipino, L., Lee, Y., & Wang, R. (2002). Data Quality Assessment. *Communications of ACM,* (April), 211-218.

Preston, R. (2001). Don't Overlook Data Integrity in E-Biz Planning. *Internetweek, Editor's Note,*(June 25), 7.

Price, H. (1994). How Clean is your Data? *Journal of Database Management, 5*(1), 36-39.

Ram, S., & Ramesh, V. (1997). Integrity constraint Integration in Heterogeneous Databases: An Enhanced Methodology for Schema Integration. *Information Systems,* 22(8), 423-446.

Rob, P., & Coronel, C. (2000). *Database Systems: Design, Implementation and Management* (4th ed.). Cambridge, MA: Course Technology, A Division of Thomson Learning.

Roddick, J. F., Al-Jadir, L., Bertossi, L., Dumas, M., Estrella, F., Gregersen, H., Hornsby, K., Lufter, J., Mandreoli, F., Mannisto, T., Mayol, E., & Wedemeijer, L. (2000). Evolution and Change in Data Management-Issues and Directions. *SIGMOD Record,* 29(1), 21-25.

Schein, E. (1987). *The Clinical Perspective in Fieldwork*. Newbury Park: Sage.

Segev, A. (1996). On Information

Quality and the WWW Impact. Paper presented at the *1996 Conference on Information Quality*, Cambridge, MA, 16-23.

Sogini, M. (2003). IBM Plans Software to Link Data from Multiple Sources. *Computerworld,*(January 13), 4.

Storey, V. C., & Wang, R. Y. (1998). An Analysis of Quality Requirements in Database Design. Paper presented at the *1998 Conference on Information Quality*, Massachusetts Institute of Technology, 64-87.

Strong, D. M., Lee, Y. W., & Wang, R. Y. (1997). Data Quality in Context. *Communications of the ACM,* 40(5), 103-110.

Susman, G. (1983). Action research: a sociotechnical systems perspective. In G. Morgan (Ed.), *Beyond Method: Strategies for Social Research* (pp. 95-113). Newbury Park: Sage.

Tayi, G. K., & Ballou, D. (1999). Examining Data Quality. *Communications of the ACM, 41*(2), 54-57.

Trisolini, S., Lenzerini, M., & Nardi, D. (1999). *Data Integration and Warehousing in Telecom Italia.* Paper presented at the SIGMOD (Special Interest Group on Management of Data), Pyiladephia, PA.

Wand, Y., & Wang, R. Y. (1996). Anchoring Data Quality Dimensions in Ontological Foundations. *Communications of the ACM, 39*(11), 86-95.

Wang, R. Y., Lee, Y. L., Pipino, L., & Strong, D. M. (1998). Manage Your Information as a Product. *Sloan Management Review, 39*(4), 95-105.

Wang, R. Y., Storey, V. C., & Firth, C. P. (1995). A Framework for Analysis of Data Quality Research. *IEEE Transactions on Knowledge and Data Engineering, 7*(4), 623-640.

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

Winter, R. (1989). *Learning From Experience*. London: The Falmer Press.

Yang W. Lee is an Assistant Professor and Joseph G. Reisman Research Professor in the College of Business Administration at Northeastern University. Dr. Lee's publications have appeared in leading journals such as Communications of the ACM, Journal of MIS, Sloan Management Review, Information & Management, and IEEE Computer. She co-authored Quality Information and Knowledge (Prentice Hall, 1999), Data Quality (Kluwer Acadmic Publishers, 2000) and Journey to Data Quality (MIT Press, forthcoming). Her research interests include data quality, IT-mediated institutional learning, and systems integration. She was a visiting professor at MIT Sloan School of Management, where she taught e-systems integration and conducted research on data quality. She was also a Conference Co-chair for International Conference on Information Quality (ICIQ) in Cambridge. She received her Ph.D. from MIT. She can be reached at y.lee@neu.edu.

Leo Pipino is Emeritus Professor of MIS at the University of Massachusetts Lowell. Dr. Pipino is also an affiliate at the MIT Information Quality Program. Dr. Pipino's research interests include issues in data and information quality including data quality metrics for enterprise services and web services, applications of neural

networks, data mining and data warehouses. He has published in such journals as Journal of MIS, Communications of ACM, Sloan Management Review, Decision Support Systems, and Expert Systems with Applications.

Diane M. Strong is an Associate Professor in the Management Department at Worcester Polytechnic Institute and Director of the MIS program. She received her Ph.D. in Information Systems from Carnegie Mellon University. Dr. Strong's research centers on data and information quality and on the organizational impacts of MIS application systems, especially ERP systems. Her publications have appeared in leading journals such as Communications of the ACM, ACM Transactions on Information Systems, Journal of Systems and Software, Journal of Management Information Systems, and Information & Management. She is a member of AIS, ACM, and INFORMS, is serving on AIS Council, and was Program Co-Chair for AMCIS in Boston. She can be reached at dstrong@wpi.edu.

Richard Wang is Director of MIT Information Quality Program and Co-Director for the Total Data Quality Management Program at the Massachusetts Institute of Technology. He has served as a professor at MIT for a decade, and the faculty of the University of Arizona, Tucson, Boston University, and a Visiting Professor at the University of California, Berkeley. Wang has put the term Information Quality on the intellectual map with myriad publications and conferences. In 1996, Prof. Wang organized the premier International Conference on Information Quality, which he has served as the general conference chair, and currently Chairman of the Board. He also co-founded the Workshop on Information Technologies and Systems (WITS) in 1991. Wang's books on information quality include Quality Information and Knowledge (Prentice Hall, 1999) and Data Quality (Kluwer Academic, 2001) and Journey to Data Quality (MIT Press, forthcoming). Dr. Wang received his Ph.D. degree from MIT.