International Journal of Operations Research

International Journal of Operations Research Vol. 2, No. 2, 1−8 (2005)

An Object-oriented Quality Framework with Optimization Models for Managing Data Quality in Data Warehouse Applications

Chung-Yang Chen[1,*] , Yu-Liang Chi[2], and Philip Wolfe[3]

[1]Deptartment of Information Management, Chang Gung University, 259 Wen-Hwa 1st Road, Kwei-Shan, Tao-Yuan, 333, Taiwan, R.O.C.

[2] Deptartment of Management Information Systems, Chung-Yuan Christian University, Chung-Li 320, Taiwan

[3]Department of Industrial Engineering, Arizona State University, Tempe, AZ 85287-5906, USA

Received December 2004; *Revised March 2005*; *Accepted May 2005*

Abstract—Data quality is an important issue, especially in large-scale data applications such as data warehousing (DW). The validity (a super quality type specialized by accuracy, completeness, consistency, and currency) of data in fact has corresponding impacts on ad-hoc decisions. To ensure quality, improvement actions such as edit check, imputation, and audit et al. are applied. Yet these utilize and consume resources and time, particularly for large sets of data which get more critical as achieving zero-defects. In this paper, an object-oriented and multi-dimensional quality framework is suggested in order to comprehensively realize data quality. Two simple mixed binary integer programming optimization models based on the quality framework are presented to study the cost issues and investment allocation according to different quality aspects in DW. An example is then given to illustrate the managerial use of the models.

Keywords—Data quality, Object concept, Data warehousing, Crashing costs, Mixed binary integer programming

1. INTRODUCTION

1.1. Data quality

Data are important in our daily lives. Individuals and businesses use data to make decisions. The volume of data grows tremendously when businesses analyze and discover business patterns based on all kinds of historical data. Therefore, the quality of data affects the quality of decisions, which further impact the success of business or organization.

1.2. Data quality in data warehouse applications

The emergence of data warehouses (DW) greatly helps make ad-hoc business decisions and identify potential business patterns according to information or data captured or derived from data warehouses. The data or derived information is contained in a view or a fact table with multi-dimensional tables/views such as star schema, snowflake, data cube, fact table, or data mart in a materialized or virtual way. Figure 1 illustrates a simplified DW application process.

Poor quality of the views in Figure 1 become critical when crucial business decisions are made based on the views. Data quality assurance is then crucial in order to ensure the quality of the decision. Research papers and methods, e.g. Redman (1996), TQdM (English,1999), and TDQM (Wang et al., 2001) et al. provide a comprehensive overview that covers regarding lifecycle of data quality assurance. Moreover, the cost due to poor data quality in DW applications has been illustrated by Chen (2002).

1.3. Research problems

1.3.1. Generalization and specialization problems

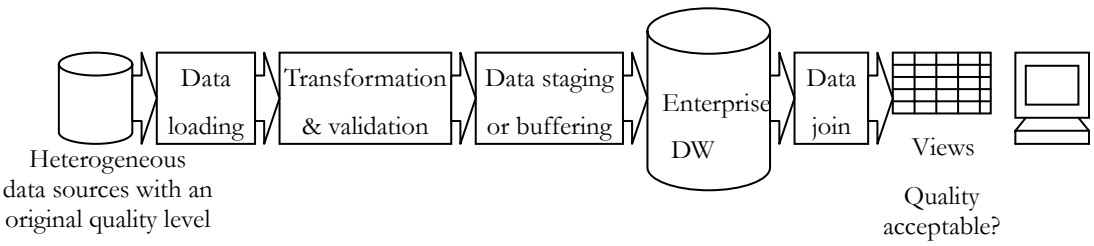


Figure 1. Simplified Data Warehouse Application Process (Chen, 2002).

* Corresponding author's email: cychen@mail.cgu.edu.tw

In the area of information/systems quality, researchers have been working on methods to assess and control data quality. One of the accomplishments is a number of quality dimensions identified. Some quality dimensions can be found and are, in fact, an abstract/super type of some others. Thus they are difficult to be realized or operationally defined. This refers to a generalization problem among quality dimensions.

Imprecise data are regarded as incorrect or invalid; yet an invalid data value does not necessarily mean that the data value is inaccurate. It might be just outdated but the value is always accurate. This refers to a specialization problem. This type of problems become obvious where the data quality dimensions are based on user's perception as well, e.g., the quality dimension "validity" of same data may have different meanings to different users.

Hence, a data quality problem may not be described by a single dimension solely. In addition, dimensions used to measure data quality ought to be operable. Thus there is a need of using multiple and specialized dimensions to view data quality problems in order to gain a more comprehensive view on the quality of data.

1.3.2. Data quality management problems

Improving data quality consumes business resources, e.g. money and time et al., and this is more critical when managing the quality of data warehouses. Furthermore, achieving a perfect data quality status is uneasy for large data sets, and this is particularly true when the resources are not free. Therefore, there exists a tradeoff between data quality improvement and limited business resources: improvement action gets more difficult and difficult when approaching zero-defects.

For example, an estimated accuracy value 0.9995 implies approximately 5 physical or logical errors per 10,000 data items. A management question then arises: is it really cost-effective to find and correct these errors? The efforts spent to increase the same 1% accuracy level on different original levels e.g., 50% and 99%, are certainly different. Hence data quality improvement needs to consider available business resources and the diminishing returns.

However, there is no research for studying and optimizing this tradeoff

2. LITERATURE SURVEY

2.1. Data quality

Poor decisional problems due to poor data quality have been evinced and quantified (Redman, 1996; English, 1999; Chen, 2002). Two famous examples are the two tremendous tragedies: the explosion of the space shuttle Challenger and the shooting down of an Iranian Airbus by the USS Vincennes. They have been proven that poor data quality is one of the reasons that should be blamed (Fisher and Kingma, 2001). Wang, Shanks et al. presented in their articles the impacts and affects on businesses due to data quality problems (Wand and Wang, 1996; Shanks, 1998),. Srinivasan, InduShobha, Strong, et al. used simulations and analysis to demonstrate the impacts of data quality on decision-making (Strong, 1997; InduShobha, 1999; Srinivasan, 1999).

As more and more damages and failures resulting from poor data quality are reported, businesses and organizations should pay more attention to the data they generate or use

2.2. Data quality problems in DW

Data quality problems in DW result from having errors on data or information. Such data errors can be classified as two types: syntactic and semantic errors. As Ballou et al. state, data warehousing efforts have to address several potential problems such as data from different sources may exhibit semantic differences and syntactic inconsistencies (Ballou et al., 1999).

Syntactic inconsistencies, or syntactic errors, mean that data's physical properties are invalid. Semantic errors occur when the data value is syntactically correct but is not the true value. For example, as a simple data join example shown in Figure 2, each data item in the tuple is syntactically valid before they arrive in the star schema, but the entire information of the employee is semantically incorrect because it does not make too much sense that a

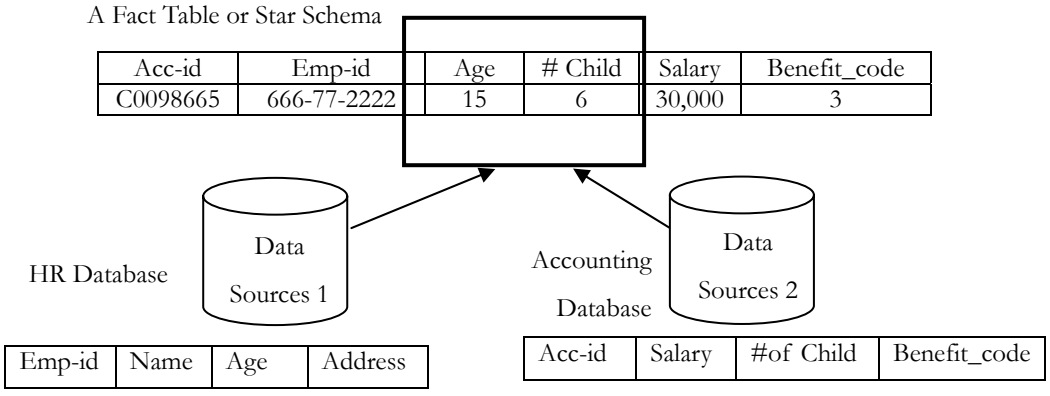


Figure 2. A simple semantic error example when data are joined.

person of age 15 has 6 children.

A syntactic error can be detected and corrected by automatic edit checks. Yet semantic error is difficult to be detected in that ad-hoc logical rules are different and difficult to model from application to application. Thus manual inspection or audit is sometimes used for this type of errors. Therefore, improving two types of errors involves different levels of efforts and techniques

2.3. Optimization models for managing data quality

Regarding optimization model for managing data quality, there are several found in the literature. They fall into two categories. The first category is operational optimization for data editing task, e.g. Ragsdale and McKeown (1996), Garfinkel et al. (1986) and Schaffer (1987). The second category is managerial and strategic concern of time and resource investment. References of Hamlen (1980) and Ballou and Tayi (1989) fall into this category.

Models in the first category address optimal data editing methods to identify most likely locations and fields of erroneous records, or to impute the missing value. Imputation of missing data considers discrete or continuous domain values. In addition, these models are used in organizations to deal with numeric and fact data, such as statistical agencies, in order to gain better statistical analysis results.

There exists a tradeoff between cost of poor data quality and cost of quality improvement. There is little research regarding optimizing data quality improvement. An earlier operational research model by Hamlen in the area of accountings and internal control design minimizes system cost subject to a given level of quality desired in the system's output (Hamlen, 1980). The paper presented an optimization model for the design and evaluation of an internal control system. Several (binary x_i) validation processes are available for against the j type of errors with the error reduction rate e_{ij} respectively. Giving management goal of error rates ε_j and cost c_i for control i, the combination of these control procedures is optimized in order to minimize costs of poor quality.

In addition, Ballou and Tayi presented an OR model as the methodology for allocating control procedures over single data quality enhancement task (Ballou and Tayi, 1989). The model uses the integer programming technique to determine which control procedure (i.e. the tool or method to perform data quality enhancement) is applied to which data set to gain the maximum savings due to data quality losses. The model considers limited tools and effectiveness for each enhancement. They also assume that control procedures enhance data quality by reading data one by one and making necessary corrections as well.

Ballou and Tayi's model is useful for managing syntactic problems. The model may not be practical for semantic data errors, particularly when the task is to check and correct data for a larger amount of records.

3. METHODOLOGY

3.1. An object oriented, multi-dimensional data quality framework

The first step to improve data quality is to understand and realize the data quality problems in DW. Object concept can be applied to address the generalization/specialization problems mentioned earlier. Object concept is not only about class and its object, but also mechanisms such as inheritance mechanisms, polymorphism, overriding and dynamic method binding using super call, and interface et al. In order to comprehensively realize data quality, this paper suggests four operational definitions to specialize a generic and abstract type of data quality dimension, validity or correctness. This is inheritance. The four operable quality dimensions are accuracy, completeness, consistency, and currency. Figure 3 describes the object relationships using UML, an object oriented representation tool.

As the figure implies, the four specialized dimensions can share common property abstraction but more diversifying characteristics. The common abstract properties are described in their super type, correctness or validity. Hence correctness or validity (i.e. correct data value) does not provide too much specific information about how correct or how valid the data is. While the four specialized dimensions, e.g. accurate data or current data, would have more specific information meaning that the data is correct in quantity, or correct in its representation.

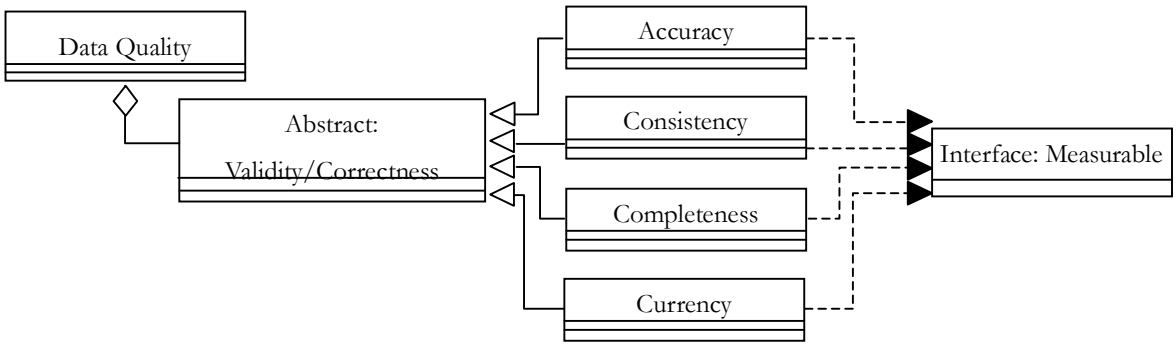


Figure 3. Object-oriented representation of generic, abstract quality dimensions and their sub-types, which are operable.

Polymorphism and behavior overriding that diversifies operable characteristics for quality dimensions can be available. All the common measurement behaviors, e.g. ad hoc measurement metric, value conversion, et al. ought to be described in type Measurable and are overridden in the four specialized dimensions. Due to the property of polymorphism, accuracy, completeness, consistency, or currency can all refer to data validity or correctness problem.

Therefore, validity or correctness has no operational definition due to its abstraction, and this explains why validity or correctness is not easy to be quantified or perceptional (e.g. degree of satisfaction) survey is always used to quantify data correctness problems. Accuracy is operationally defined as a percentage of data items not having syntactic data errors in a data set. Completeness in data warehousing refers to the degree of not having missing or redundant records for a record/tuple. Currency refers to whether the data value reflects the current representation/status of the tuple/record/object. Consistency is the degree of the data values that are semantically identical, although they have different representations.

3.2 An investment allocation model based on the quality framework

Two managerial models for managing multiple data quality improvements based on the proposed multi-dimensional quality framework are constructed. The objective of both models is to maximize the saving of loss due to poor data quality through the multi-dimensional improvements. Four binary integer decision variables, y_i (i = 1, 2, 3, and 4), are defined as switching variables regarding whether or not to perform the improvement. Four decision variables x_i are defined as the target quality values for accuracy, completeness, consistency, and currency respectively.

Mixed binary integer programming is applicable because the decision variables in the proposed model include binary integers (the necessity of performing a data improvement) and decimal numbers (the target data quality levels). Model construction is stated as follows.

3.2.1. Setup costs constraints

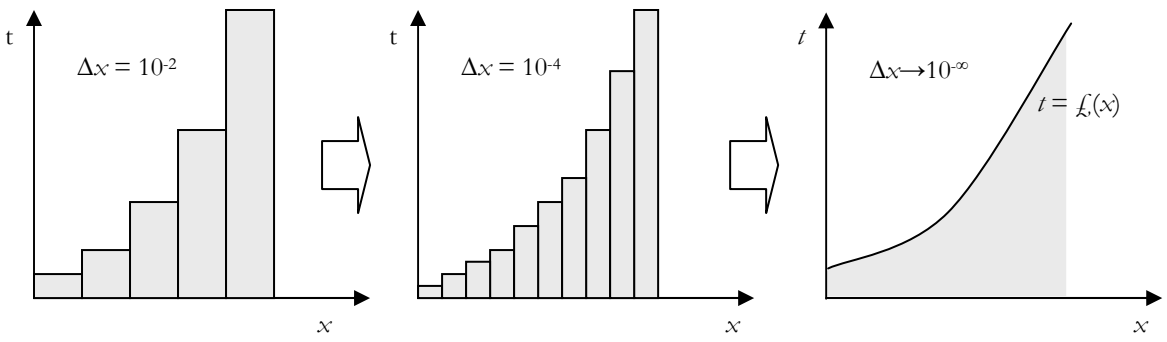
Two cost items, (1) setup cost and (2) incremental detection and correction cost, are considered. Setup cost incurs when a data quality improvement is performed. The setup cost occurs regardless the effectiveness of the improvement as long as the action is taken (y_i = 1). Let k_i denote the setup cost for improvement i; then the total setup cost for can be expressed as:

$$\text{Total setup cost} = \sum_{i=1}^{4} k_i * y_i$$

3.2.2. Detection & correction cost

The detection and correction (D&C) cost is defined as the cost spent for data editing, i.e., finding and correcting expected data quality errors. In general, more investments for data editing are expected in order to achieve next quality levels. Thus, when the quality level is already high, it becomes more difficult to detect and correct the next error. Such characteristics of nonlinear diminish returns on improvement actions can be illustrated in Figure 4.

In Figure 4, Δx is defined by this research as the resolution of the improvement progress. If the size of a data set is n, then according to the diagram, the unit expected time required to detect and correct errors for improving the quality level from x to $x+\Delta x$ is t, which equals $f_t(x)$ when it is a continuous case ($\Delta x = 10^{-\infty}$). The total expected time spent would be $n*\Delta x*t$, which is $n*\Delta x*f_t(x)$ in the continuous case. Therefore, the D&C cost can be written as (The variable β is the target quality level) (OH refers to the unit overhead cost of a data quality improvement):



t: Expected time to detect and correct next errors to next quality levels
x: The next quality level
$f_t(x)$: A quadratic or higher order nonlinear function

Figure 4. Expected time (t) for detecting and correcting next errors in order to achieving next data quality level ($x+\Delta x$).

$\sum_{(x=\alpha,\beta)} n * \Delta x * \pounds_i(x) * OH$ for the discrete case

$\int_{(x=\alpha,\beta)} n * \pounds_i(x) * OH$ for the continuous case

3.2.3. Crashing costs

Crashing costs are applied when investments are being spent to control the time of each data editing. Assume that before improvement, the data quality value for dimension i is a, the relationship between the crashing cost and the target data quality value can be described using a nonlinear function $f_i(x_i)$, which is illustrated as Figure 5 (a_i and b_i are constants).

3.2.4. Time cnstraints

3.2.4.1. Without crashing costs

Suppose the time allowed for data quality improvements is D, the time constraint for expected accumulated D&C time can be expressed as follows:

$$\int_{(x=ai,xi)} n * f_i(x_i) \le D$$

Parameter n is the size of the view, schema, or fact table; a_i is the current data quality value for dimension i; x_i is the target data quality value for dimension i; and i = 1, 2, 3, and 4 for the accuracy, completeness, consistency, and currency improvement.

3.2.4.2. With cashing costs

In the case when crashing cost is applied, the unit correction time is controlled and constant. Thus the system spends the following amount of time for improving, for example, accuracy from a_1 to x_1:

$t_1 n(x_1 - a_1)$,

where: t_1 is the unit correction time for accuracy; n is the total amount of data items; x_1 is the target accuracy value; and a_1 is the original accuracy value.

3.2.3. Switching constraints between decision variables y_i and x_i

When an improvement task i is performed, setup cost is incurred. It is possible that even the setup cost is invested; the target quality value eventually remains unchanged. Therefore, the model should address the interdependencies between decision variables y_i and x_i :

Rule 1. If the target data quality value x_i remains unchanged, then the system should not perform the improvement at all, i.e. $y_i = 0$, the setup cost is 0.

Rule 2. If the target data quality value x_i is changed, i.e. higher, then y_i must be 1.

A switching constraint is then facilitated to describe the above relationship between the target value x_i and the setup cost incurred as the improvement is performed, i.e. variable y_i = 1. Therefore, the switching constraint is formulated as:

$0 \le y_i - (x_i - a_i) < 1$

$y_i = 1$ or 0; $0 \le x_i \le 1$; $0 \le a_i \le 1$,

For example, when $x_i > a_i$, then y_i must be 1; when $x_i = a_i$, y_i might be 0 or 1. This constraint ensures y_i to be 0 when $x_i = a_i$ since it has a negative coefficient in the objective function.

3.2.4. Model formulation

Two models for two situations are then formulated as follows:

Case I: Resource allocation with variant unit detection & correction time

$$\text{Max } Z = \sum_i (c_i n x_i - c_i n a_i) \quad (1)$$

subject to:

$$\sum_i [k_i y_i + \int_{(X=\alpha,xi)} n * \pounds_i(xi) * OH_i] \le R, \quad (2)$$

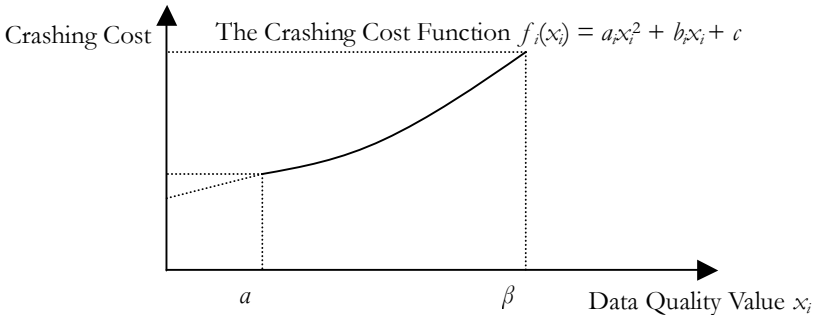


Figure 5. A nonlinear crashing cost function when $\Delta x \to 10^{-\infty}$.

$$\sum_i \int_{(X=\alpha i, xi)} n * \mathcal{L}_i(xi) \le D, \quad (3)$$

$$0 \le y_i - (x_i - a_i) < 1, \quad (4)$$

$$a_i \le x_i \le 1, \quad (5)$$

$$y_i = 0 \text{ or } 1. \quad (6)$$

Case II: Resource allocation with fixed unit detection and correction time and crashing costs

$$\text{Max } Z = \sum_i (c_i n x_i - c_i n a_i) \quad (1)$$

subject to:

$$\sum_i [k_i y_i + \int_{(X=\alpha,\beta)} N * \mathcal{L}_i(x_i) + t_i n (x_i - \alpha_i) * OH_i] \le R, \quad (2)$$

$$\sum_i tin (xi - \alpha i) \le D, \quad (3)$$

$$0 \le y_i - (x_i - \alpha_i) < 1, \quad (4)$$

$$a_i \le x_i \le 1, \quad (5)$$

$$y_i = 0 \text{ or } 1. \quad (6)$$

where:

i	quality dimension, where $i = 1$ for accuracy, 2 for completeness, 3 for consistency, and 4 for currency.
x_i	data quality target value, where $0 \le x_i \le 1$.
y_i	whether or not to perform the data quality improvement task, $y_i = 0$ or 1.
c_i	unit cost of having a unit of data quality problem for dimension i, where the unit for accuracy is per data error; completeness is per omission/ redundancy error; consistency is per invalid version of copy; and timeliness is per day delay.
k_i	setup cost for performing data quality improvement i.
OH_i	unit overhead cost of a data quality improvement i.
a_i	original value of data quality I, where $0 \le a_i \le 1$.
D	total time allowed for data quality improvements.
R	available monetary resources for the entire data quality improvement program.
N	is the size of the view, schema, or fact table
t_i	unit D&C time for dimension i.
$\mathcal{L}_i(x)$	nonlinear, quadratic D&C time function for dimension i

The initial error rates a_i should be provided by data source vendors, or it can be obtained by using statistical and stochastic metrics. Metrics of measuring syntactic errors can be seen in Morey (1982), Firth (1996), Pierce (1997), and Wang et al. (2001). Process (the procedure of transferring or loading data) reliability problems, e.g. Type I (error ignorance) and Type II (false alarm) errors ought to be considered according when estimating the initial values.

Constraint (2) describes that the total costs (the setup costs, error detection and correction cost, and the crashing cost) cannot exceed budget R. Constraint (3) denotes that the total time for the improvement cannot exceed D, which is the total time allowed. Constraint (4) is the switching constraint. Constraint (5) and (6) are the domain constraints.

4. ILLSTRATIVE EXAMPLE

ABC Ltd., located in Taipei Taiwan, is a consulting company that deals with DW applications for its customers. When the data from all places are loaded and the star schema is generated, a DW manager would like to ensure that the data view they prepare for the customer is of good quality. Some data quality cleansing tasks and audits are then performed according to the four dimensions mentioned in earlier sections. However, due to the large volume of data, and labor and time constraints, they would like to know improvement investments on different levels of data quality would not cost too much while yield optimal result of saving on data quality loss.

Assume that the data warehousing action is to prepare a star schema for a client to mail the sales promotion packages to all possible customers. Therefore, the parameters of the model are identified and given as follows: c_i = [1, 0.6, 0.4, 0.8]; k_i = [100, 500, 250, 250]; OH_i = [0.5, 10, 10, 10]; a_i = [0.9125, 0.8845, 0.9312, 0.8233]; and R = 1600 USD. Suppose that there are $2*10^5$ (= n) records on the prepared schema. The scheduled amount of time for the system to prepare and process the information is 72 working hours. To simplify the problem, quadratic nonlinear D&C functions: $\mathcal{L}_1(x_1)$, $\mathcal{L}_2(x_2)$, $\mathcal{L}_3(x_3)$ and $\mathcal{L}_4(x_4)$ are defined as: $0.0006\ x_1^2 + 0.00002$, $0.0015\ x_2^2 + 0.00005$, $0.0012\ x_3^2 + 0.00004$, and $0.0024\ x_4^2 + 0.00008$, respectively.

Two mathematic optimization tools, LINGO 8.0 Enterprise and QM for Windows, were employed to solve the model and compare the results.

4.1. The optimization results

Suppose that the errors are evenly distributed in the target data set. Both LINGO and QM for Windows utilize branch and bound techniques and total enumeration for solving integer programming related problems. The local optimal presented by the tool is the global optimal because the possible solutions by the concave objective function and the convex constraints form a convex solution set. Total enumeration is applied to verify the above observation. Table 1 illustrates the result from both software packages.

4.2. Discussion

The above results can be expressed in Table 2 in a

managerial way, which shows target quality levels for different dimensions. Decision makers may not be interested to know academically the target quality values. In fact, he/she may be interested in learning: (1) how the resources should be assigned for different aspects of quality improvements in order to obtaining maximum beneficial results; and (2) the improvement should not have been performed at all if the improvement costs are spent but the quality level turns out to be unchanged. Therefore, another objective of the models should provide such kind of information, in addition to just numerically predict the final data quality levels.

For example, from the table illustrated above, the consistency improvement in case I does not need to be performed because the improvement cost is more than the loss at the quality level in this case. The model hence suggests not perform consistency improvement at all and save the money for other improvement actions. Therefore, such kind of derived, useful information for these two cases from the models is summarized in the following table.

The cost (c_i) due to poor data quality for each dimension depends on the concern of different data warehousing applications. For example, if a schema is prepared to discover rules or patterns of customer purchase behavior, then consistency and currency (that is, semantics of the values among fields) are relatively more important Moreover, if a schema is prepared to collect all residents' information for mailing sales promotion packages, then accuracy and completeness are relatively more important in that the correctness/validity of syntax, completeness, and the update (currency) of values in fields, e.g. ADDRESS, impacts the success of the package delivery, which is the main concern of the data warehousing action in this case.

In addition, this research does not study a comprehensive index/number resulting from the composition of these four dimensions. The reason is that such a composite index does not provide too much information. For example, if a validity level is 0.9, then it is so general (a generalization problem again) that we have no idea about what specifically are wrong. However, because of the property of polymorphism (an object concept), any errors of these four dimensions can be regarded as a data validity/correctness problem, e.g. an accuracy error is a validity/correctness error.

Table 1. The optimization results of the example

Decision Variables	Optimal Solution by LINGO	Total Enumeration
Case I		
Z (savings)	44224.77	44224.77
Improve Accuracy? (Y1)	1	1
Improve Completeness? (Y2)	1	1
Improve Consistency? (Y3)	0	0
Improve Currency? (Y4)	1	1
Target Accuracy Value (X1)	1.0000000	1.0000000
Target Completeness (X2)	0.9963388	0.9963387
Target Consistency (X3)	0.9312000	0.9312000
Target Concurrency (X4)	0.9064695	0.9064694
Number of Iterations	184	16 cases
Techniques	Branch & bound	Simplex
Case II		
Z (savings)	41722.22	41722.22
Improve Accuracy? (Y1)	1	1
Improve Completeness? (Y2)	0	0
Improve Consistency? (Y3)	0	0
Improve Currency? (Y4)	1	1
Target Accuracy Value (X1)	1.0000080	1.0000000
Target Completeness (X2)	0.8845080	0.8845000
Target Consistency (X3)	0.9312080	0.9312000
Target Concurrency (X4)	0.9746889	0.9746889
Number of Iterations	27	16 cases
Techniques	Branch & bound	Simplex

Table 2. The suggestion of resource investments in the example

DQ Improvements	Suggested Resources Investment	
	Time	Money
Case I		
Accuracy	9.958	105
Completeness	30.826	808
Consistency	0	0
Currency	31.216	562
Case II		
Accuracy	17.5	255
Completeness	0	0
Consistency	0	0
Currency	54.5	1244

5. CONCLUSION

In this paper, an object-oriented, multi-dimensional quality framework has been presented to study the generalization and specialization problems among quality dimensions. Such a framework offers a way to understand data quality under different concerns/aspects. In addition, two optimization models based on the above quality framework have been developed and are used: (1) to comprehensively understand data quality, and (2) to optimize aspect-oriented data quality improvements with limited resource allocation.

Typical data quality research focuses on single data quality dimension improvement for assessing the quality of the data. In this paper, four specialized dimensions are operationally defined with different concerns of quality. Once data quality being understood, resources for data quality improvements should be assigned according to different quality aspects.

ACKNOWLEDGEMENTS

The researchers gratefully thank Taiwan National Science Consortium (NSC) for the support and funding of this research project (Research Grant Code: 92-2213-E-182-003)

REFERENCES

1. Ballou, D. and Tayi, K. (1999). Enhancing data quality in data warehouse environment. *Communication of ACM*, 42(1): 73-78
2. Ballou, D. and Tayi, K. (1989). Methodology for allocating resources for data quality enhancement. *Communications of the ACM*, 32(3): 320-329.
3. Chen, C.Y. (2002). Measuring data accuracy with consideration of domain-specific impact costs in data warehouses, *Data Quality Journal*, 7(1).
4. English, L. (1999). *Data Warehouse and Business Information Quality*. John Wiley and Sons Inc, New York.
5. Firth, C.P. (1996). Data quality in practice: experience from the frontline. *The 1996 Information Quality Conference*, Massachusetts Institute of Technology, October 25-26.
6. Fisher, C.W. and Kingma, B.R. (2001). Criticality of data quality as exemplified in two disasters. *Information and Management*, 39(2): 109-116.
7. Garfinkel, R.S., Kunnathur, A.S, and Liepins, G.E. (1986). *Optimal Imputation of Erroneous Data: Categorical Data*, General Edits. Operations Research, 34: pp. 744-751.
8. Hamlen, S. (1980). A chance-constrained mixed integer programming model for internal control design. *The Accounting Review*, LV(4): 578-593.
9. InduShobha, C. Ballou, D.P, and Pazer, H.L. (1999). The impact of data quality information on decision making: an exploratory analysis. *IEEE Transactions on Knowledge and Data Engineering*, 11(6): 853-864.
10. Levitin, A. and Redman, T. (1995). Quality dimensions of a conceptual view. *Information Processing & Management*, 31(1): 81-88.
11. Morey, R. (1982). Estimating and improving the quality of information in a MIS. *Communications of the ACM*, 25(5): 337-342.
12. Orr, K. (1998). Data quality and systems theory. *Communications of the ACM*, 41(2): 66-73.
13. Pierce, E. (1997). Modeling database error rates. *Data Quality Journal*, 3(1): 14-30.
14. Ragsdale, C.T. and McKeown, P.G. (1996). On solving the continuous data editing problem. *Computers & Operations Research*, 23(3): 263-273.
15. Redman, T. (1996). *Data Quality for the Information Age*. Artech House Inc., Boston, USA.
16. Schaffer, J. (1987). Procedure for solving the data-editing problem with both continuous and discrete data types. *Naval Research Logistics*. 34: 879–890.
17. Shanks, G. (1997). Conceptual data modeling: an empirical study of expert and novice data modelers. *Australian Journal of Information Systems*, 4(2): 63-73.
18. Srinivasan, R. (1999). Impact of information quality and decision-maker quality on decision quality: a theoretical model and simulation analysis. *Decision Support Systems*, 26: 275-286.
19. Strong, D. (1997). IT process designs for improving information quality and reducing exception handling: a simulation experiment. *Information & Management*, 31(5): 251-263.
20. Svanks, M.I. (1998 Dec). Integrity analysis: Methods for automating data quality assurance. *Information and Software Technology*, 595-605.
21. Wang, R., Stroey, V., and Firth, C. (1995). A framework for analysis of data quality research. *IEEE Transactions on Knowledge and Data Engineering*, 7(4): 623-640.
22. Wand, Y. and Wang, R. (1996). Anchoring data quality dimensions in ontological foundations. *CACM*, 39(11).