An Open Platform of Data Quality Monitoring for ERP Information Systems

Pawel Sieniawski[1] and Bogdan Trawinski[2]

Wroclaw University of Technology, Institute of Applied Informatics
Wybrzeze S. Wyspianskiego 27, 50-370 Wroclaw, Poland
[1]p.sieniawski@columb-technologies.com, [2]trawinski@pwr.wroc.pl

Abstract. In the paper an Open Platform of Data Quality Monitoring developed to audit data maintained in any enterprise resource planning (ERP) system is presented. Data quality of a database is verified according to a control schema defined in XML. Elementary tests can be developed using external test library written in .NET code embedded in XML and therefore can be easily incorporated into the Platform. Openness of the Platform makes it possible to use complex control techniques without the necessity to introduce any specific meta language. In order to evaluate the Platform tests for six different ERP systems were carried out using several data quality metrics. Results of the investigation proved the usefulness and flexibility of the Platform.

Key words: Data quality monitoring, data quality metrics, ERP information systems, problem intensity charts

1 Introduction

Computer viruses caused total loss of about 55 milliard dollars in 2003, according to the Trend Micro's study. However, yearly loss resulting from a low quality of data is estimated to 611 milliard dollars for USA companies [11]. Nevertheless, most of investments aim at the protection against outer attacks and the protection of data possessed against inner erosion is rather marginal. Data quality examination is usually carried out only when the secondary usage is attempted, for example during the construction of a corporate data warehouse [3,6]. After completing their projects only 20 per cent of companies continue regular data quality monitoring [12]. Most often the process of data quality monitoring is the introductory part of a more general process of data quality improvement [1]. It is focused on the analysis of defect occurrence in order to remove them automatically. Human verification and approval is needed to solve many problems [3], thus it is suggested to distinguish clearly both processes. At present, the definitions of good quality data focus mainly on its consumers and its use [2,9,8,11], and they often take the form of the question to what extent data satisfy the requirements of their intended use. There are some different approaches to determine metrics of the quality of data sets, e.g. local metrics [6], goal metrics [7] and generic metrics [8] and others are proposed. The construction of the Open Platform of Data Quality Monitoring presented in the paper differentiates from some solutions proposed in [4,5,6], because no specific language to define data correctness has been developed.

It has been assumed to use commonly known programming languages to detect errors in data, e.g. those included in the Microsoft .NET environment. The methodology developed incorporates the best aspects of theoretical models [6,8,9] extending them of the analyses at the strategic level. The investigation conducted by means of the Platform on six different enterprise resource planning systems made it possible to evaluate the solution proposed, to indicate significant features of different metrics and to assess some aspects influencing quality of data in the systems of this kind.

2 Features of the Monitoring Platform

The Open Platform of Data Quality Monitoring (OPDQM) has been developed to audit data gathered in any enterprise resource planning (ERP) system. It enables to obtain detailed lists of errors found in data, visualize the results in form of different graphs and to present general output calculated using several data quality metrics.

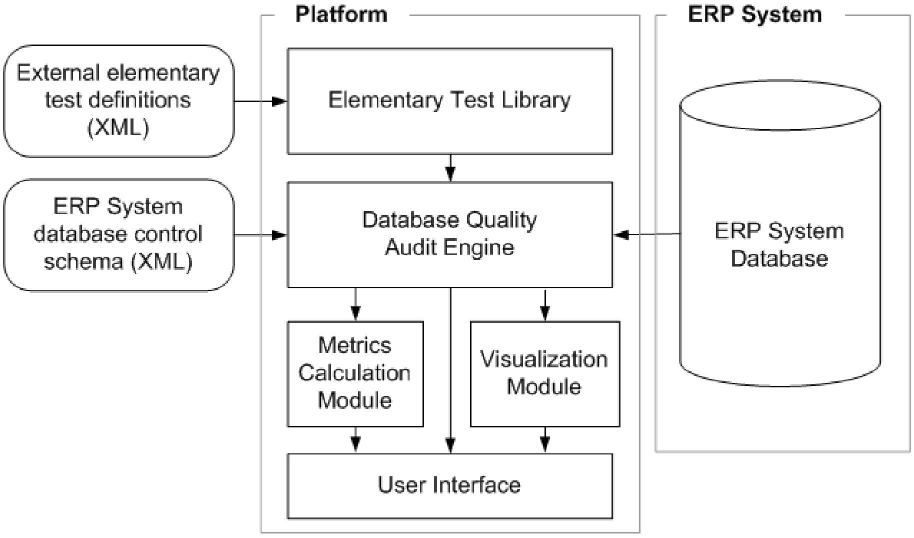

Platform
ERP System
External elementary test definitions (XML)
ERP System database control schema (XML)
Elementary Test Library
Database Quality Audit Engine
Metrics Calculation Module
Visualization Module
User Interface
ERP System Database

Fig. 1. The architecture of the OPDQM Platform

2.1 Architecture of the Platform.

Data quality of any database is verified using a control schema defined in XML. The database being verified can be deployed on any known database management system, only adequate ODBC or OLE DB drivers are required. Elementary test sets can be written in .NET code embedded in XML and easily incorporated into external test library. The architecture of the OPDQM platform is shown in Fig. 1. The main component of the platform is the Audit Engine, which executes elementary tests. The results of tests are passed directly to user interface in the form of list of errors found and to modules responsible for metrics calculation and visualization. The library of elementary tests can be extended by external tests in a form of source code to be compiled and incorporated as an integral part of the platform during its operation.

2.2 Uniform error messages

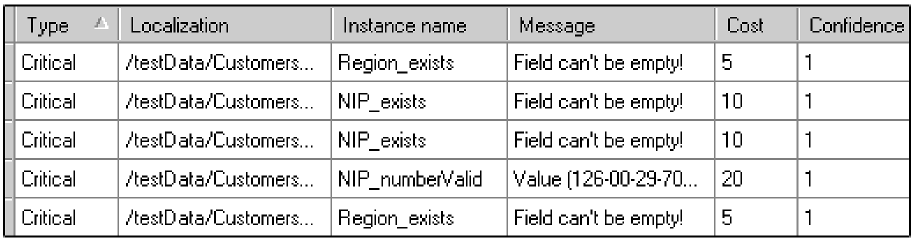
A unified way of error messaging has been designed in the OPDQM platform. Each elementary test can return any number of uniform data error messages. However, testing a single row or single field return usually no more than one message per one test run. Information contained in an error message is presented in Table 1.

Tab. 1. The structure of an error message

Element	**Values**	**Description**
Type	critical, warning, information, external	Determines error importance, points out also errors occurring out of the system (external)
Localization	structure pointing out a table, record or field in a database	Contains information of the nearest error occurrence place possible to be localized
Test instance name	text	Test instance name assigned in control schema.
Message	text	Error message returned by testing function
Confidence	number from the interval $[0, 1]$	Determines the probability of error occurrence
Repair cost	number	Repair cost expressed in currency, effort or other form, determined in test schema
Operator	identifier	Determines ERP system operator responsible for faultiness.
Time	datatime	Data and time of performing a test

2.3 Presentation of test results

The simplest form of presentation are tables containing all error messages obtained during test runs (Fig. 2). In order to assure effective use of the results achieved the sorting, filtering, selecting and colouring functions are provided. It is also possible to export messages in XML format. The other way of presenting quality of data are graphs showing the values of different metrics. They are calculated on the basis of error messages or received directly from a database tested.



Type	Localization	Instance name	Message	Cost	Confidence
Critical	/testData/Customers...	Region_exists	Field can't be empty!	5	1
Critical	/testData/Customers...	NIP_exists	Field can't be empty!	10	1
Critical	/testData/Customers...	NIP_exists	Field can't be empty!	10	1
Critical	/testData/Customers...	NIP_numberValid	Value (126-00-29-70...	20	1
Critical	/testData/Customers...	Region_exists	Field can't be empty!	5	1

Fig. 2. Error messages in the form of a table

Very useful form of presentation are problem intensity charts which have been designed to visualize errors detected in a single table. In the problem intensity chart the x axis represents records of a table tested and the y axis its columns. A vertical bar stands for a problem detected and the intensity of colours corresponds to the number of problems occurring in a given place, i.e. in a column of a given table row. In turn, the spaces (blank places) indicate records without any error. An example of a problem intensity chart is shown in Fig. 3. The intensity of colours may alternatively denote the costs of removing errors or the severity of problems.

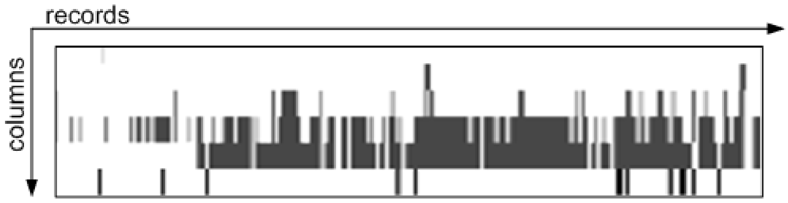


Fig. 3. An example of a problem intensity chart

3 Overview of metrics implemented in the Platform

Managers need aggregate information on quality of data gathered in an ERP system in order to be able to take a decision about repair activities. The data quality metrics seem to be the most appropriate means to provide such information. According to the rules of a good metrics [10] they are characterized using such features as readability, complexity, ability to compare different databases and mobilization of the management to undertake repair activities. So far, twelve following metrics have been implemented in the Platform. Beneath the following denotation is used: E is the set of all error messages, T is the set of all elementary test runs, R is the set of records tested, $R \leftarrow E$ is a set of all records referenced by at least one error message, F is the set of fields tested, $F \leftarrow E$ is a set of all fields referenced by at least one error message. In turn, $card(E)$, $card(T)$, $card(R)$, $card(R \leftarrow E)$, $card(F \leftarrow E)$ are the cardinalities of these sets respectively.

(1) Number of errors. It is the total number of errors detected in a database which can be expressed by the following formula:

$$DQM_E = card(E) \tag{1}$$

(2) Percentage of errors detected in tests performed. It is the ratio of the number of errors detected in a database to the number of all tests performed, expressed by the following formula:

$$DQM_{E/T} = \frac{card(E)}{card(T)} * 100\% \tag{2}$$

(3) Number of errors per 1000 records. It is equal to the number of errors detected falling on 1000 records tested and is expressed as:

$$DQM_{E/R/1000} = \frac{card(E)}{card(R)} * 1000 \tag{3}$$

(4) Number of invalid records. It is the number of records where at least one error was detected and is equal to the number of records referenced by at least one error message. It can be expressed by the following formula:

$$DQM_{R \leftarrow E} = card(R \leftarrow E) \tag{4}$$

(5) Percentage of invalid records in records tested. It is the ratio of the number of records where at least one error was detected to the number of all records tested, expressed by the following formula:

$$DQM_{R \leftarrow E/R} = \frac{card(R \leftarrow E)}{card(R)} * 100\% \tag{5}$$

(6) Number of invalid records per 1000 records. It is equal to the number of records where at least one error was detected falling on 1000 records tested, expressed as:

$$DQM_{R \leftarrow E/R/1000} = \frac{card(R \leftarrow E)}{card(R)} * 1000 \tag{6}$$

(7) Number of invalid fields. It is the number of fields where at least one error was detected and is equal to the number of fields referenced by at least one error message. It can be expressed by the following formula:

$$DQM_{F \leftarrow E} = card(F \leftarrow E) \tag{7}$$

(8) Percentage of invalid fields in fields tested. It is the ratio of the number of fields where at least one error was detected to the number of all fields tested, expressed by the following formula:

$$DQM_{F \leftarrow E/F} = \frac{card(F \leftarrow E)}{card(F)} * 100\% \tag{8}$$

(9) Number of invalid fields per 1000 records. It is equal to the number of fields where at least one error was detected falling on 1000 records tested, expressed as:

$$DQM_{F \leftarrow E/R/1000} = \frac{card(F \leftarrow E)}{card(R)} * 1000 \tag{9}$$

(10) Weighted average of percentage of errors, invalid records and invalid fields. It is the weighted average of three metrics (2), (5), (7). This hybrid metrics is expressed as follows:

$$DQM_{wav} = \frac{w_1 * DQM_{E/T} + w_2 * DQM_{R \leftarrow E/R} + w_3 * DQM_{F \leftarrow E/F}}{w_1 + w_2 + w_3} \tag{10}$$

The above weights were determined experimentally and during tests were assigned the following values: $w_1 = 0.5$, $w_2 = 0.3$ and $w_3 = 0.2$.

(11) Cost of database repair. Expressed in terms of money or effort which should be expended in order to remove all errors from the database. The value of the metrics equals to the sum of repair costs assigned to errors detected:

$$DQM_{rep} = \sum C_{rep}(e_i) \tag{11}$$

where $C_{rep}(e_i)$ is the cost of repair of i-th error detected.

(12) Database depreciation. It is the ratio of the cost of database repair to the total value of database, expressed by the following formula:

$$DQM_{rep} = \frac{\sum C_{rep}(e_i)}{\sum V_{rec}(r_j)} * 100\% \tag{12}$$

where $V_{rec}(r_j)$ is the value of j-th record in the database.

4 Data preparation for tests

The investigation has been carried out using data taken from six ERP systems exploited in medium size companies functioning on the market of the FMCGs. The systems under study ranged from single systems developed by order to the brand ones delivered by world leading producers. The characteristics of data used during tests are presented in Table 2, where the names of systems have been anonymized. In order to assure comparability of the results, a small fragment of data was chosen for tests. It was the table of clients which can be found in each system. In order to simplify the verification, only the rows containing data of companies located in Poland were taken into account.

Tab. 2. Characteristics of data used in the investigation

ERP System denotation	ERP System origin	Number of records to test	Year of data origin	ISO quality standard introduced
System 1	local	1626	2006	
System 2	local	5102	2005	+
System 3	local	6057	2005	
System 4	foreign	613	2004	
System 5	foreign	1417	2004	+
System 6	local	2228	2006	+

The list of elementary tests applied is presented in Table 3. The basic value of each record tested with correct data was assumed as equal to 100. This value could be increased by 5 when optional fields such as Phone or E-mail, were filled. For evaluation of error repair costs artificial unit of DQ$ was assumed.

5 Results of the investigation

5.1 Comparison of metrics

The results of database quality monitoring using different metrics are shown in Fig. 4. The metrics Id conform the denotation used in chapter 3. Left part of the Fig. 4 (a)

Tab. 3. Elementary test applied

Field scope	**Elementary test**	**Error repair cost [DQ$]**
Name	Check if not null Check correctness Check unique identity (min. length 6 characters) Check unique name	20 2 5 20
Adress	Check if not null	10
City	Check if not null Check if present in the list of 10 thousand of Polish towns and villages	5 5
Region	Check if not null	5
Country	Check if takes one of three values: Polska, PL, Poland	5
ZIP	Check if not null Check the mask of Polish ZIP code (xx-xxx)	10 10
TIN	Check if not null Check the mask of Polish tax identification number. Check if control sum conforms with Luhn's algorithm	10 5 20
Phone	Check length (min. 7 digits) Check the mask of local, intercity, international or cell number	10 5
E-mail	Check the conformity with e-mail address standard	5
Row/table scope	**Elementary test**	**Error repair cost [DQ$]**
Row	Check the conformity of ZIP code with town and province using external data source	5
Table	Detect duplicates using Levenshtein's distance calculated on the basis of name and tax id with threshold value of 96%	20

comprises the comparison of percentage scale metrics. All the metrics, beside the database depreciation (no. 12), range the data quality of the systems tested in the same order, what means that the system 1 contains the best data. It could be also observed that the metrics of percentage of invalid records (no. 5) is much more sensitive to error occurrence than others. It reaches values about 5 times higher than other metrics and perhaps therefore is the most frequently mentioned in the publications in the field [13]. Moreover, it is the only metrics differentiating the significance of errors detected. Further analysis of data in the system 5 revealed that its database contained considerable number of duplicates, what led to the high cost of repair. Right part of the Fig. 4 (b) comprises the results of the tests performed using the linear scale metrics. These metrics cannot serve the comparison of different databases, however they are a good starting point to the estimation of the cost or effort of database repair.

5.2 Comparison of data quality in ERP systems

There are many factors having impact on data quality in an ERP system. The most important are the quality of a system application, the history of data migration, corporate standards and regulations, organizational culture and first of all its users and administrators. Data monitoring results are given in Fig. 5. The relatively high quality of data in the system 1 is the consequence of reach prompt and control mechanisms available in the process of data input.

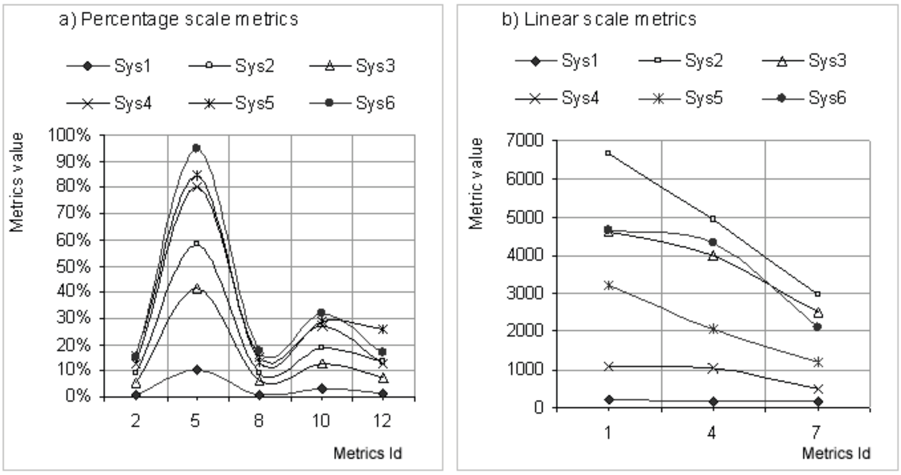


Fig. 4. Results of tests performed using different data quality metrics

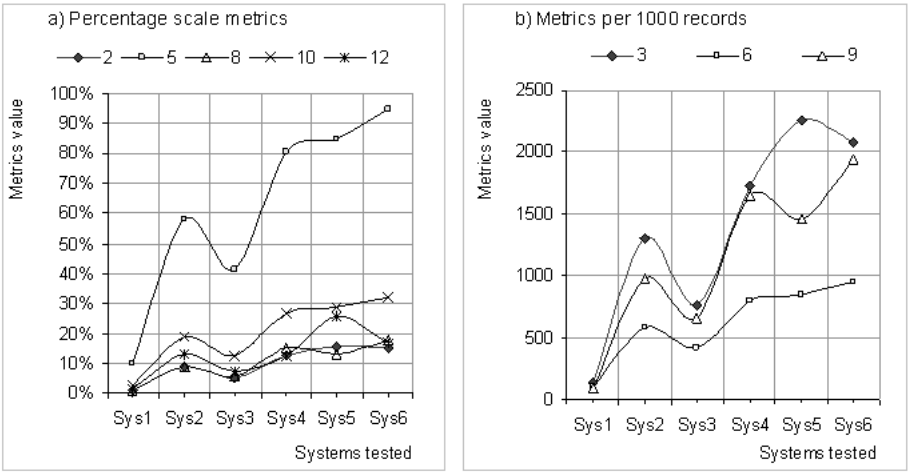


Fig. 5. Results of data quality monitoring of six ERP systems

An attempt to verify the hypothesis that the ISO quality standard introduced by the corporation influences positively the data quality is shown in Fig. 6 a). However it does not result in data quality worsening, but in practice has no effect. It turned out that the ISO quality standard does not cover the issue of the quality of corporate databases. The correlation between the ERP system origin and the data quality is presented in Fig. 6 b). The results obtained suggest that local systems, produced by Polish software companies, are equipped with more effective and better localized control tools. For example the tax identification number differs in each country in its length, format and the method of control digit calculation.

Problem intensity charts for six ERP systems are presented in Fig. 7-12. The order of records shown in charts conforms the sequence of their input into each system.

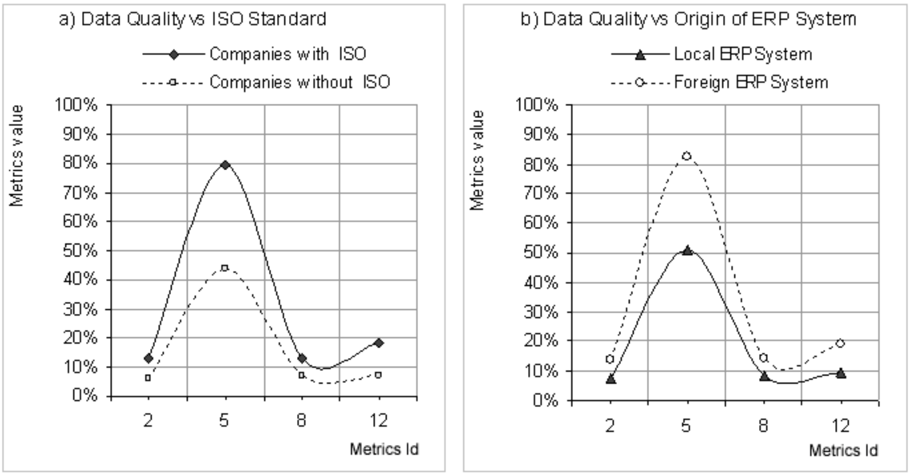


Fig. 6. The impact of external factors analysis on data quality

In the chart of the System 2 (Fig. 8) a series of records with two errors occurring simultaneously could be observed. Probably those records were imported form a previous ERP system and were not adjusted to the requirements of the new one.

In the chart for the System 4 a group of records of low quality could be observed, which significantly decrease quality of a whole database. However, there is a series of records of comparable size without any error in this system too. In the system 6 the field of Region was used inappropriately to its assumed purpose to place there different data for which there were no especially dedicated fields.

The problem intensity charts may turn out to be useful for detecting repeatable errors which can be limited at the level of ERP system applications eg. in the case of the systems 4, 5 and 6.

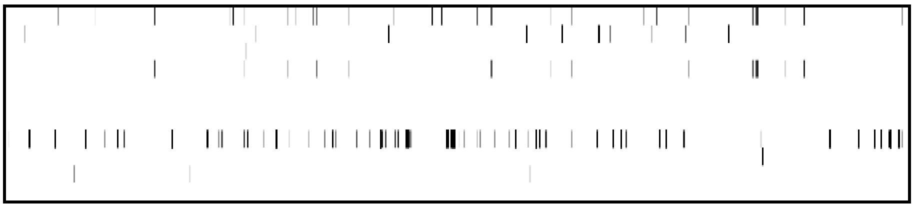


Fig. 7. Problem intensity chart for System 1

6 Conclusions and future work

The Open Platform of Data Quality Monitoring has been proved to be useful to monitor data quality of ERP systems, but it could be also used to audit other classes of information systems based on relational databases. The openness of the Platform makes it possible to use complex control techniques without the necessity to introduce a

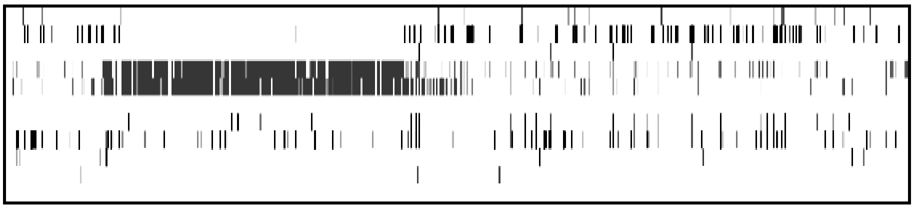


Fig. 8. Problem intensity chart for System 2

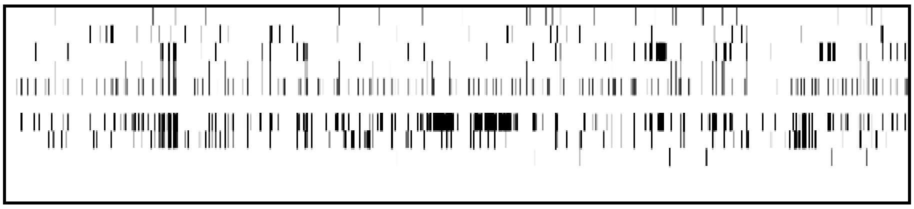


Fig. 9. Problem intensity chart for System 3

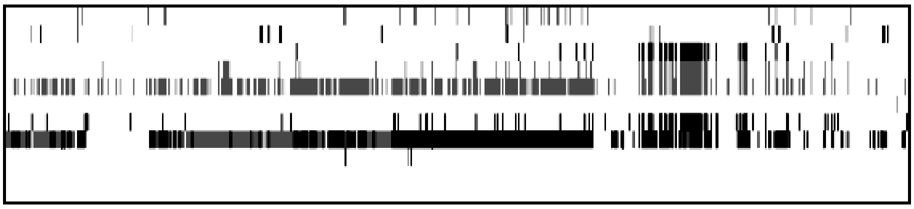


Fig. 10. Problem intensity chart for System 4

specific meta language to define data correctness. Therefore this enables to implement the Platform in fast and flexible way and achieve an acceptable level of performance.

The metrics applied in the Platform can be classified into two groups. First group comprises metrics useful to compare the data quality of different databases or parts of the same database as well as to trace the effectiveness of activities undertaken to ameliorate the quality of corporate data. These are percentage scale metrics and metrics calculated per 1000 records. In turn the second group constitute linear scale metrics which can be used to estimate effort and costs of database repair. Both groups are very important tools of data quality management.

The problem intensity charts turned out to be a usable method of visualizing the results of data monitoring. They allow to identify groups of repeatable problems occurring in data and in consequence they may contribute to the improvement of the process of collecting data.

The investigations showed also that the ERP systems developed in Poland are customized better to local regulations and standards and therefore can achieve higher quality of their databases. Moreover, the introduction of the ISO quality standard does not have practically any impact on quality of data collected by the corporation.

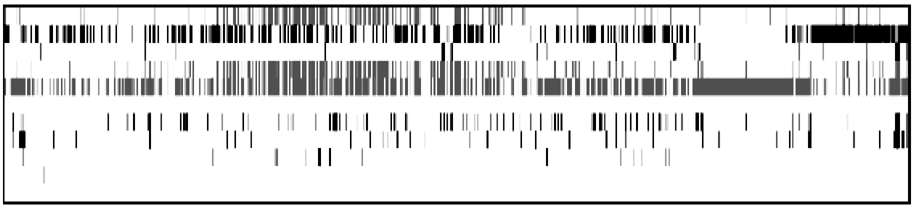


Fig. 11. Problem intensity chart for System 5

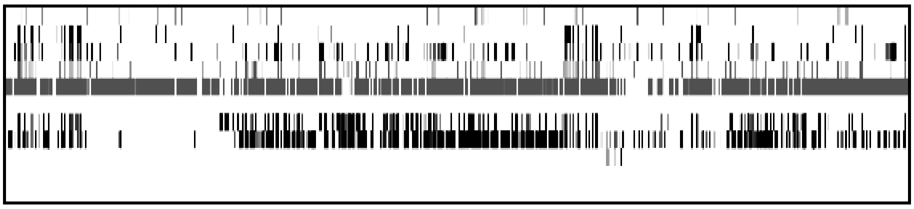


Fig. 12. Problem intensity chart for System 6

References

1. Data Monitoring: Taking Control of Your Information Assets, DataFlux Corp., (2004)
2. Defining and Measuring Traffic Data Quality, Office of Policy Federal Highway Administration, (2002)
3. English L.: Improving Data Warehouse and Business Information Quality. Wiley (1999)
4. Galhardas H., Florescu D., Shasha D., Simon E.: An Extensible Framework for Data Cleaning. ICDE 2000 poster paper, San Diego (2000)
5. Galhardas H., Florescu D., Shasha D., Simon E. and Saita C.: Declarative Data Cleaning: Language, Model and Algorithms, VLDB 2001, Rome (2001)
6. Jarke M., Jeusfeld M., Quix C.,: Design and Analysis of Quality Information for Data Warehouses. Proceedings of the 17th Internat. Conf. on Conceptual Modeling (ER'98), Singapore (1998)
7. Kovac R., Lee Y. W., Pipino L. L.: Total Data Quality Management: The Case of IRI. The 1997 Conference on Information Quality, Cambridge (1997)
8. Lee Y. W., Pipino L. L., Wang R. Y.: Data Quality Assessment. Communications of the ACM, (April 2002) 211-218
9. Lee Y. W., Strong D. M., Wang R. Y.: Data Quality In Context. Communications of the ACM, (May 1997) 103-110
10. Loshin D.: Developing Information Quality Metrics. DM Review Magazine, (May 2005)
11. Olsen J. E.: Data Quality: The Accuracy Dimension. Morgan Kaufmann Publishers, (2003)
12. Zellner G., Helfert M., Sousa C.: Data Quality Problems and Proactive Data Quality Management in Data-Warehouse-Systems. Proceedings of BITWorld, (2002)
13. Loshin D.: Developing Information Quality Metrics. DM Review Magazine, (May 2005)