Measuring Data Quality in Analytical Projects

Anca Ioana ANDREESCU, Anda BELCIU, Alexandra FLOREA,
Vlad DIACONITA
University of Economic Studies, Bucharest, Romania
anca.andreescu@ie.ase.ro, anda.velicanu@ie.ase.ro, alexandra.florea@ie.ase.ro,
diaconita.vlad@ie.ase.ro

Measuring and assuring data quality in analytical projects are considered very important issues and overseeing their benefits may cause serious consequences for the efficiency of organizations. Data profiling and data cleaning are two essential activities in a data quality process, along with data integration, enrichment and monitoring. Data warehouses require and provide extensive support for data cleaning. These loads and renew continuously huge amounts of data from a variety of sources, so the probability that some of the sources contain "dirty data" is great. Also, analytics tools offer, to some extent, facilities for assessing and assuring data quality as a built in support or by using their proprietary programming languages. This paper emphasizes the scope and relevance of a data quality measurement in analytical projects by the means of two intensively used tools such as Oracle Warehouse Builder and SAS 9.3.
Keywords: *data quality, data profiling, analytical tools, data warehouses*

1 Introduction

Data quality represents an important issue in every business. To be successful, companies need high-quality data on inventory, supplies, customers, vendors and other vital enterprise information in order to run efficiently their data analysis applications (e.g. decision support systems, data mining, customer relationship management) and produce accurate results. As companies develop analytical and business intelligence systems on their transactional systems, the reliability of key performance indicators and data mining predictions will depend entirely on the validity of the data on which they are based. Any type of data quality issue could potentially lead to erroneous data mining and analysis results which in turn could lead to severe consequences, financial or otherwise.

But while the importance of valid data for business decision making is increasing, so does to the same extent the challenge to ensure their validity. Information flows continuously in the company from various sources and systems and a large number of users and so the volume of data being generated is increasing exponentially day by day.

2. Data Quality Assessment

Data quality assessment is a complex process through which we can obtain a complete assessment of an organization's data. Through this process we get not only a full image regarding the data quality issues the company is facing but also an accurate view of the time and effort required to fix those problems.

In the first stages of research on data quality an evaluation method using a vector composed of elements which describe several easily evaluated data quality aspects was proposed [1]. Amongst the quality factors included we mention relevance, accuracy, data actuality etc. Later on, the list of proposed quality factors by researchers such as [2], [3] and [4] grew larger and larger reaching almost 200 elements.

However, defining a number of sophisticated data quality factors, how large that number might be, is not enough to obtain a relevant data quality assessment. As mentioned in [5] data

quality assessment is highly context and application dependent. It is difficult to formulate a general solution that will work in all situations.

The metrics for determining data quality may be difficult to define because they are domain or application specific. A common method to define the quality of the data is represented by data profiling.

Data profiling is the process of analyzing large data sets obtaining a set of statistical indicators regarding that data, such as minimum, maximum, mean, mode, percentile, standard deviation, frequency, and variation as well as other aggregates such as count and sum. During data profiling we could obtain additional metadata information such as data type, length, discrete values, uniqueness, occurrence of null values, typical string patterns, and abstract type recognition [6]. Through data profiling we make an assessment of data to understand its content, structure, quality and dependencies.

There are some common methods used in profiling, no matter the tool selected, as mentioned in [7]: structure discovery – verification if the different patterns of data are valid; data discovery – verifies if the data value is correct, error free and valid; relationship discovery – checking if all the key relationships are maintained and data redundancy where we check if the same data has multiple representations.

Profiling techniques can be grouped in two categories: manual or automated using a profiling tool. Manual techniques involve people who unravel the data to assess their condition, query by query. This is appropriate for small data sets from a single source, with less than 50 fields, where data is relatively simple.

The automated techniques use software tools to gather summary statistics and analysis. These tools are most appropriate for projects with hundreds of thousands of records, many fields, multiple sources and questionable documentation and metadata. Sophisticated technology has been built for data profiling to handle complex problems, particularly for high-profile projects and critical missions.

Choosing an appropriate profiling tool might be a difficult task so it's useful to know what the differences between them are. Mostly they vary in the architecture used to analyze the data and in the work environment they provide for the team that generates the data profile.

From the architectural point of view we distinguish between query based profiles and repository based profiles.

Some profiling tools require having technical skill at running SQL queries on source data or a view of the source data. Although this creates good information about the data it also has several limitations regarding performance:

- **Performance risks** – the queries operated on production systems slow the systems, sometimes significantly. When additional information is needed or if users want to see the actual data, a second query is executed creating more pressure on the system. This risk can be reduced by making a copy of the data, but this requires replicating the entire environment, both hardware systems and software, which can be costly and time consuming.
- **Traceability risks** - the data in production systems is constantly changing. Statistics and metadata extracted from profiles based on queries risk to immediately become outdated.
- **Integrity risks** – it is complicated to acquire comprehensive knowledge using query-based analysis. The queries are based on assumptions, and the objective is to confirm and quantify expectations about what is wrong and right in the data. Given this, it is easy to overlook problems that you have not reported.

Other tools generate data profiles as part of a scheduled process and store the results in a profile repository. Saved results can include content such as summary statistics,

metadata, patterns, key relationships and data values. These results can be further analyzed by the users or be saved for later trend analysis. Profile repositories that allow users to explore information and view the values of the original data in the context of source records are those that offer more versatility and stability for non-technical audiences.

As mentioned before, there is a great number of data quality and profiling tools available on the market and choosing between them can be a difficult task. In [8] was conducted an extensive analysis of the major features of a series of data quality tools which is summarized in figure 1.

Tool	Data sources	Extrac-tion	Loading	Incre-mental updates	Inter-face	Metadata repository	Perfor-mance	Versio-ning	Function library	Language binding	Debug-ging	Excep-tions	Data lineage
Centrus Merge/Purge	DB	-	-	-	G	-	-	-	-	-	-	-	-
ChoiceMaker	DB, FF	-	-	-	G	Y	Y	-	Y	N	Y	Y	Y
Data Integrator	Several	Y	Y	Y	G	Y	Y	Y	Y	-	Y	Y	Y
DataBlade	Informix	-	Informix	-	G	-	-	-	-	-	Y	X	X
DataFusion	DB	Y	DB	Y	G	-	Y	Y	Y	N	Y	X	-
DataStage	Several	Y	Y	-	G	Y	Y	Y	Y	Y	Y	Y	Y
DeDupe	DB	-	-	-	G	-	-	-	-	-	-	-	-
dfPower	Several	Y	Y	-	G	Y	Y	-	-	-	-	-	-
DoubleTake	ODBC	-	-	-	G	-	-	-	-	Y	-	-	-
ETI*Data Cleanser	Several	-	-	-	G	Y	Y	-	Y	Y	-	Y	-
ETLQ	Several	Y	Y	-	G	Y	Y	Y	Y	N	-	-	-
Firstlogic	DB, FF	Y	Y	-	G	Y	Y	-	Y	Y	-	-	-
Hummingbird ETL	Several	Y	Y	Y	G	Y	Y	Y	Y	N	Y	M	Y
Identity Search Server	DB	-	-	Y	G	Y	-	-	-	-	-	-	-
Informatica ETL	Several	Y	Y	Y	G	Y	Y	Y	Y	Y	Y	Y	Y
MatchIT	DB	-	-	-	G	-	-	-	-	-	Y	-	-
Merge/Purge Plus	-	-	-	-	-	-	-	-	-	-	-	-	-
Migration Architect	Several	-	-	-	G	Y	-	-	-	-	-	-	-
NaDIS	-	X	-	X	G	X	-	-	X	X	-	X	X
QuickAddress Batch	ODBC	X	-	X	G	X	-	-	X	X	-	X	X
Sagent	Several	Y	Y	X	G	Y	Y	X	Y	N, SQL	-	-	-
SQL Server 2000 DTS	Several	Y	Y	X	G	X	-	-	Y	N	X	X	X
SQL Server 2005	Several	Y	Y	-	G	-	-	-	Y	N	-	-	-
Sunopsis	DB, FF	Y	Y	Y	G	Y	Y	Y	X	SQL	Y	Y	X
Trillium	Several	Y	Y	-	G	Y	Y	Y	Y	N	Y	-	Y
WizRule	DB, FF	-	-	-	G	-	Y	-	-	-	-	-	-
WizSame	DB, FF	-	-	-	G	-	Y	-	-	-	-	-	-
WizWhy	DB, FF	-	-	-	G	-	Y	-	-	-	-	-	-

Fig.1. General functionalities of commercial data quality tools [8]

The authors have used the following notations for constructing their feature analysis table: Y: supported; X: not supported; -: unknown information; N: native; DB: only relational databases; FF: only flat files; G: graphical; M: manual. Such an analysis can represent a major support element when choosing an appropriate data quality tool. Once the data profiling process is completed and all the problems have been identified, special attention must be paid to data cleaning. Through the cleansing (or scrubbing) of data we can detect and remove errors and inconsistencies and thus improve the quality of data.

The complexity of the data cleaning process varies due to the type of storage of the processed data. If data is stored in single collections the problems that might arise come from incorrect data entry, missing information or other invalid data. If data is to be integrated from multiple sources one of the main problems is redundancy: same data is stored in different representations. Thus it becomes necessary to eliminate duplicate information and consolidate different data representations in order to provide access to accurate and consistent data. When working with data warehouses, which process particularly large amounts of data from a array of sources extensive support for data cleaning is a mandatory requirement, especially since data warehouses are used for decision making, so that the correctness of their data is vital to avoid wrong conclusions. For instance,

as mentioned in [9], duplicated or missing information will produce incorrect or misleading statistics ("garbage in, garbage out").
When selecting a data cleaning approach there are several requirement that should be taken into consideration: detection and removal of all major inconsistencies and errors in individual and multiple data sources; use of appropriate tools in order to limit manual inspection and programming effort; use of tools that can easily cover additional sources as they might appear; specification of mapping functions in a declarative and reusable way for other data sources as well as for query processing.

3. Data Profiling in Oracle Warehouse Builder

Oracle Warehouse Builder (OWB) has a Client - Server architecture, the database being stored on the Server and the Design Centre and the Repository Browser being available for the Client. The Core Features include Enterprise ETL (Extract Load Transform), Data Quality and ERP/CRM Connectors.
The way data Quality can be assured by OWB is shown in figure 2**.**

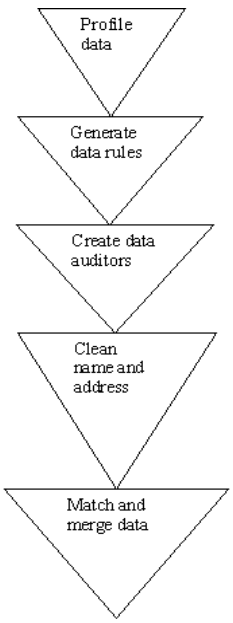


Fig. 2. Data Quality in QWB
(adapted from source: [10])

Data profiling in OWB is "a systematic analysis of data sources" [10] in order to obtain new characteristics of data.
A data rule is an automatic or user - generated expression that allows data to be formatted according to domains, constraints, for it to gain its uniformity and consistency.
According to [10] "data auditors are process flows that evaluate one or more data rules against a given table".
Name and address cleansing assumes some transformations on these types of data in order to improve the quality. The transformations include parsing, standardization, augmentation, division, etc.
Matching and merging of data has the role of determining which values actually refer to the same logic data. This process helps eliminate duplicates and unite data in single row records.
The five steps in achieving data quality are profiling the data, generating data rules, deploying corrections and cleaning the data, as shown in figure 3**.**

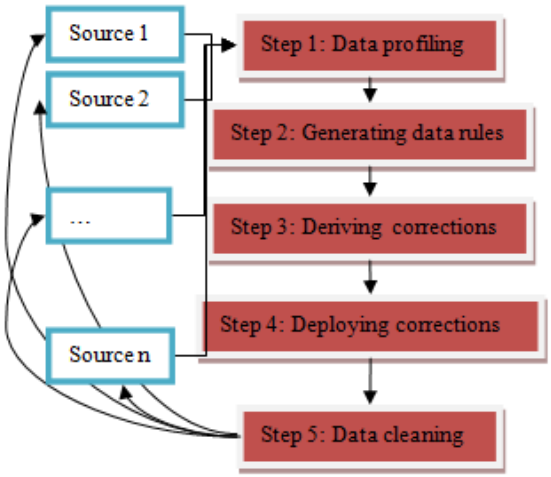


Fig. 3. Steps in achieving data quality
(adapted from source: [11])

The first and the last step work with data directly from its sources, but the cleaning process returns a new set of values.
Data Profiling is one of the ways for assuring Data Quality, next to Anomaly Detection, Business Rules and Audit, as Oracle sees it. Document [12] states the

following: "Warehouse Builder enables to discover the structural content of data, capture its semantics, and identify any anomalies or outliers prior to loading it in a system. With data profiling, one can automatically derive business rules and mappings to clean data, derive quality indices such as Six Sigma, and use auditors to continuously monitor data quality." This way data profiling can be integrated in the ETL process buying time and using quality data.

The steps in performing data profiling are presented in [13] and include: creating data profile objects (which are metadata objects in Oracle Projects), creating data profiles, configuring data profiles, loading all types of configuration parameters (pattern, domain, unique key, functional dependency, redundant column etc.), profiling data.

The main types of data profiling are described in figure 4**.**

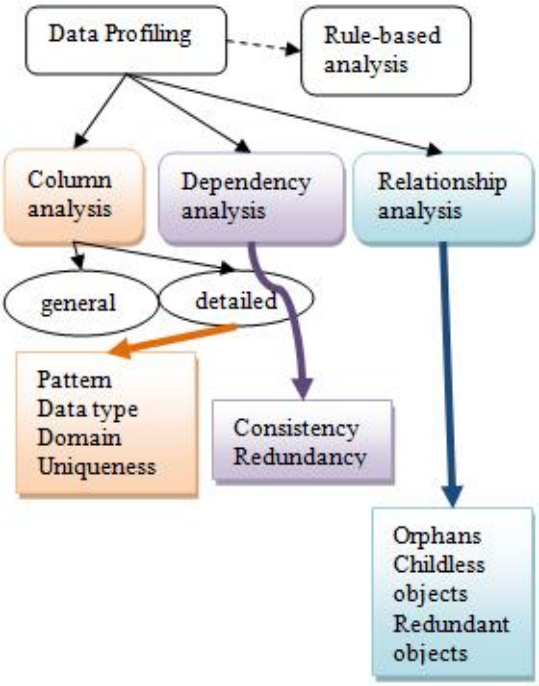


Fig.4. Types of data profiling (adapted from source: [14])

According to [11] data quality process should include four types of analyses as described below:

1. Column analysis that is based on: Uniqueness (metadata analysis); Completeness (the missing or incorrect values of attributes and thus the incomplete entities); Precision (precision and scale of numeric attributes); Uniformity (format analysis of numeric, character and date time attributes).
2. Dependency analysis, that consists of the following sub-analysis: Consistency of data type, length and domain; Primary key uniqueness; Redundancy avoided by using normalization.
3. Relationship analysis is based on: Referential integrity; Correctness using statistical control (minim, maxim, average, median etc.).
4. Rule-based analysis made through business and data rules.

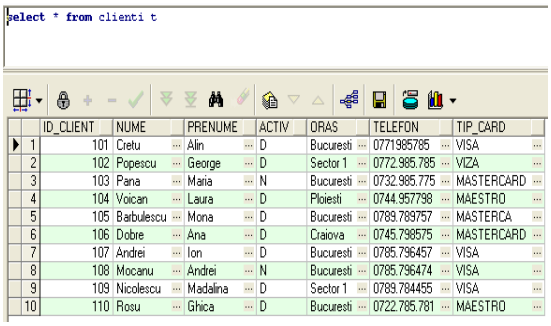
The advantages of using Oracle Warehouse Builder for data profiling are:

- allows discovering hidden things about apparently common data like: anomalies, additional relations between tables, patterns, complete domain of values, etc;
- the user can view the results in tabular or graphical format in Data Profile Editor;
- generates corrective ETL process based on business rules;
- has a simple and flexible design and generates robust ETL processes;
- works with a single repository that drives ETL processes and reporting.

The limitations of using OWB for data profiling are also shown below:

- the database workspace that is used must be Oracle 10g or higher and even though data could be accessed through JDBC connectivity, it must be staged first in an Oracle database and then used for profiling;
- data profiling process can analyze columns at a limit of 165 in each table. If it is necessary to analyze more at a time, an attribute set can be created in order to group more columns;
- complex data types cannot be analyzed if they are located on different database instances.

By following these analyses, we present an example of assuring data quality through data profiling made in Oracle Warehouse Builder. The initial set of data is presented in figure 5.



```
select * from clienti t
```

	ID_CLIENT	NUME	PRENUME	ACTIV	ORAS	TELEFON	TIP_CARD
1	101	Cretu	Alin	D	Bucuresti	0771985785	VISA
2	102	Popescu	George	D	Sector 1	0772.985.785	VIZA
3	103	Pana	Maria	N	Bucuresti	0732.985.775	MASTERCARD
4	104	Voican	Laura	D	Ploiesti	0744.957798	MAESTRO
5	105	Barbulescu	Mona	D	Bucuresti	0789.789757	MASTERCA
6	106	Dobre	Ana	D	Craiova	0745.798575	MASTERCARD
7	107	Andrei	Ion	D	Bucuresti	0785.796457	VISA
8	108	Mocanu	Andrei	N	Bucuresti	0785.796474	VISA
9	109	Nicolescu	Madalina	D	Sector 1	0789.784455	VISA
10	110	Rosu	Ghica	D	Bucuresti	0722.785.781	MAESTRO

Fig. 5. Initial set of data

First we import the metadata in Oracle Warehouse builder, after defining the connection. Once they have been imported we build a profile. Based on the imported data Oracle estimates data types and restrictions, which can be accepted and/or refined by the user.
Domain values are also detected (figure 6). On this basis rules can be derived that will be later applied as data integrity restrictions. Oracle Data Warehouse can correct the data that do not meet these rules.

Columns	Found Domain	% Compliant	Six-Sigma
ACTIV	N | D	100%	7.00
ID_CLIENT		0%	-6.25
NUME		0%	-6.25
ORAS	Bucuresti | Sector 1	80%	2.34
PRENUME		0%	-6.25
TELEFON		0%	-6.25
TIP_CARD	VISA | MASTERCARD | MAESTRO	80%	2.34

Fig. 6. Data domains

We derive rules from these domains and accept only *Visa, Mastercard* and *Maestro* as card types. Also we accept *Bucureşti, Ploieşti, Craiova* as cities, the rest are mapped to unknown, the sectors of Bucharest will be corrected later. Also we say that only active clients are allowed in our analysis. Next we create the corrections. This are the corrections applied to the source data before being copied to the destinations. The bases for this are the derived data rules previously defined. As shown in figure 7 we can define different constraints (the check constraints are autodectected).

Columns | Constraints | Data Rules

Name	Type	Local Columns	...	Check Condition
ACTIV_11	Check constraint			ACTIV IN ('D')
ORAS1_13	Check constraint			ORAS IN ('Bucuresti','Plo...
TIP_CARD_12	Check constraint			TIP_CARD IN ('VISA','MA...
CLIENTI_PK	Primary Key			
		ID_CLIENT		

Fig. 7. Define constraints

We next specify the action and the cleanse strategy for the corrections. We choose to remove the rows that aren't for active customers, correct the card type and the city. The similarity Match uses the built-in Match-Merge functionality in Oracle Warehouse Builder to change the erroneous value to the one that is most similar to it within the column domain. To correct the city we use a custom strategy and we add another function to correct the telephone number. The source code for these functions in the following:

```
//city correction
begin
if lower(oras) like '%sector%'
then return 'Bucuresti';
else return oras;
end if;

//telephone number correction
v_telefon varchar2(20);
BEGIN
For i in 1..lenght(telefon) loop
If substr(telefon,i,1) in
('0','1',
'2','3','4','5','6','7','8','9')
then v_telefon:= v_telefon||
substr(telefon,i,1);
end if;
end loop;
RETURN v_telefon;
EXCEPTION
      WHEN OTHERS THAN NULL;
RETUN NULL:
END;
```

If we examine the mapping that implements the correction, you will note that the mapping first reads data from the original table, and then attempts to load it

into a staging copy of the table with the data rule applied to it.
Those rows that pass the data rule are then copied into the corrected table. Those that fail any of the rules are then cleansed via pluggable mapping that allows you to take a series of mapping steps and "plug" them into another mapping (figure 8).

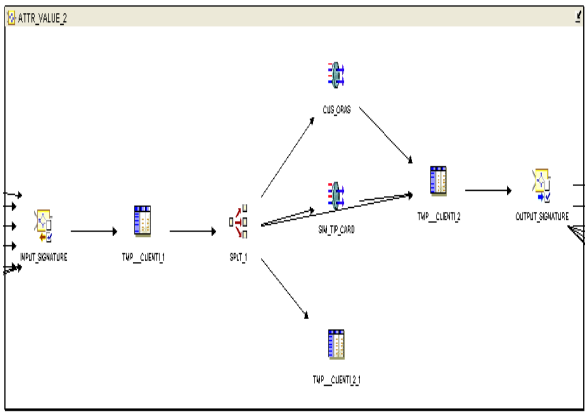


Fig. 8. Pluggable mapping

We deploy the correction objects, transformations, and mappings and then run the correction mapping.
The corrected data are to be found in the target schema as shown in figure 9**.**

```
select * from clienti t order by id_client
```

	ID_CLIENT	NUME	PRENUME	ACTIV	ORAS	TELEFON	TIP_CARD
1	101	Cretu	Alin	D	Bucuresti	0771985785	VISA
2	102	Popescu	George	D	Bucuresti	0772985785	VISA
3	104	Voican	Laura	D	Ploiesti	0744957798	MAESTRO
4	105	Barbulescu	Mona	D	Bucuresti	0789789757	MASTERCARD
5	106	Dobre	Ana	D	Craiova	0745798575	MASTERCARD
6	107	Andrei	Ion	D	Bucuresti	0785796457	VISA
7	109	Nicolescu	Madalina	D	Bucuresti	0789784455	VISA
8	110	Rosu	Ghica	D	Bucuresti	0722785781	MAESTRO

Fig. 9. The corrected data

We notice the profile of clients that have active accounts (two of them were eliminated), their city and phone number were corrected and the type of cards were adjusted to the new domain rules.
We can conclude that be using OWB one can easily define rules beginning by specifying the source and the target of the data. Domain values are detected providing a base for rules that can define the actions to take when data doesn't comply with the domain.

4. SAS Analytic Tools for Data Profiling

SAS has been known for many years as an important player in the market of the business analytics tools. It offers a variety of powerful software tools specialized in data management, data integration, analytics and reporting.
According to [15], across its solutions, SAS includes a large variety of analytical features, and therefore, it is not surprising that many of its functions are perfectly suited to profile and improve data quality. The extensive tool support for data quality in SAS can be broadly classified in two classes:

a) Built-in features offered by tools such as SAS Enterprise Miner, SAS Text Miner, SAS Model Manager, SAS Forecast Server, JMP and Data Flux Data Management Platform. These tools however, may not be available for some SAS users, may require additional training, and may be overkill if an understanding of the content of a file is all that is needed; that is, no data cleansing or other transformations are required [16].
b) Capabilities of SAS language packages, such as Base SAS, SAS/STAT and SAS/ETS.

By its analytics tools and functions, from SAS offers a variety of methods for data profiling that allow better insight into the data quality status and ways to improve it [15], such as:

- Outliners can be detected with statistical measures, while a most plausible value can be detected and calculated.
- Missing values can be imputed with methods varying from simple mean imputation to predictive models.
- Use of mathematical formulas and statistical measures to transform distribution into a more appropriate shape.

- Identify de-duplication of records based on analytical methods that describe the similarity and closeness of records.

In this paper we exemplify the use of Base SAS general quality control features that may be used to check data correctness and completeness. More precisely, we will make use of a SAS mechanism called format, which is a stored set of rules that can be used to restructure the cardinality of a column, either by viewing the data or by recoding the data [17]. These formats can be used to validate the data content through lookup tables for acceptable values for categorical variables or for acceptable ranges for interval variables.

It is worth mentioning that actually SAS formats have no exact analogy in other data management programming languages or analytic tools.

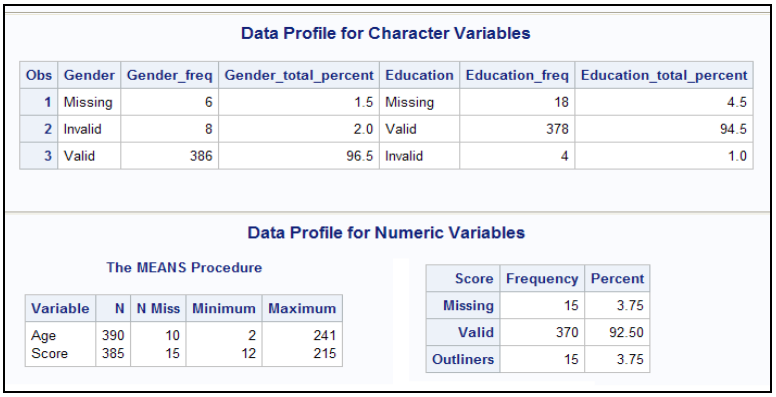
For demonstration purposes we consider a data set containing data collected from a customer satisfaction evaluation program. Four of the main variables included in the data set are Gender, Age, Education level and Score of evaluation. Valid values for these variables are described accordingly to the four SAS formats defined in the code below.

In combination with formats, PROC FREQ it is used in our example in order to determine number of the valid observations, missing values and invalid values both for character and numeric values. Afterwards, PROC MEANS it is used to obtain a data profile only for numeric variables. Below (figure 10) is the source code for creating a succinct data profile for these variables.

```
proc format;
 value $gender 'f', 'F', 'm', 'M' = 'Valid'
               ' ' = 'Missing'
               other = 'Invalid';
 value $educat 'H', 'C', 'M', 'D' = 'Valid'
               ' ' = 'Missing'
               other = 'Invalid';
 value age Low - 17 ='Outliners'
           18 - 70 = 'Valid'
           70  - High='Outliners'
           other = 'Missing';
 value score Low - 0 ='Outliners'
           1 - 100 = 'Valid'
           101  - High='Outliners'
           other = 'Missing';
run;
title 'Data Profile for Character Variables';
proc freq data=biblio.customer;
format Gender $gender. Education $educat. Age age. Score score.;
tables Gender /nocum missing out=freqc_G;
tables Education /nocum missing out=freqc_E;
tables Age /nocum missing out=freqn_A;
tables Score/nocum missing out=freqn_S;
run;
data freqc_total;
merge freqc_g (rename=(count=Gender_freq percent=Gender_total_percent))
freqc_e (rename=(count=Education_freq percent=Education_total_percent));
run;
proc print data=freqc_total;
run;
title 'Data Profile for Numeric Variables';
proc means data=biblio.customer n nmiss min max maxdec=0;
output out=freq_num;
var Age Score;
run;
```

Fig.10. SAS source code for data profiling

Output for the above SAS code is presented in figure 11, indicating an overall data profile for both numeric and character variables. We can observe that the MEAN procedure, by returning the minimum and maximum values for numeric variables, helps in identifying existing outliners.



Data Profile for Character Variables

Obs	Gender	Gender_freq	Gender_total_percent	Education	Education_freq	Education_total_percent
1	Missing	6	1.5	Missing	18	4.5
2	Invalid	8	2.0	Valid	378	94.5
3	Valid	386	96.5	Invalid	4	1.0

Data Profile for Numeric Variables

The MEANS Procedure

Variable	N	N Miss	Minimum	Maximum
Age	390	10	2	241
Score	385	15	12	215

Score	Frequency	Percent
Missing	15	3.75
Valid	370	92.50
Outliners	15	3.75

Fig.11. SAS profiling output

5. Conclusions

Data quality assessment should always be taken into account when managing and analyzing data, especially when large data volumes are or heterogeneous data sources are involved. In this paper it has been pointed out that specialized tool for data warehouses and business analytics offer support for various stages of the data quality process, including data profiling and data validation. Different tools offer different kind of support in this regard, depending on their scope. From the two examples using Oracle and SAS software, we can conclude that there are three main approaches to data quality: 1) to use predefined tools facilities, which should be straightforward; 2) to use programming languages like PL/SQL or SAS to write specialized routines, which is more time consuming, but offers great flexibility; 3) to combine the above approaches in order to customize predefined tool support.

References

[1] Juliusz L. Kulikowski. "Data Quality Assessment". *Handbook of Research on Innovations in database technologies and Applications: Current and Future Trends*, Information Science Reference publishing house, 2009, pp 378-384, ISBN13: 978-160-566-242-8.

[2] Leo L. Pipino, Yang W. Lee, Richard Y. Wang. "Data Quality Assessment", Communications of the ACM - Supporting community and building social capital, Vol.45, Issue 4, pp. 211-218, April 2002.

[3] G. Shanks, P. Darke. "Understanding Metadata and Data Quality in a Data Warehouse", Australian Computer Journal, Vol. 30, pp 122-128, 1998.

[4] Richard Y. Wang, Veda C. Storey, Christopher P. Firth. "A Framework for Analysis of Data Quality Research", IEEE Transactions on Knowledge and Data Engineering, Vol. 7, No. 4, August 1995.

[5] Tamraparni Dasu. "Data Glitches: Monsters in your Data". Handbook, Reference book, 2012. Available: http://www.research.att.com/techdocs/TD_100950.pdf

[6] David Loshin. "Master Data Management", Morgan Kaufmann Publishers, pp. 94-96, ISBN 978-012-374-225-4.

[7] Sanjay Seth. "Data Quality Assessment Approach" Internet. Available: http://hosteddocs.ittoolbox.com/ss052809.pdf

[8] - José Barateiro, Helena Galhardas –„A survey of data quality tools", Datenbank-Spektrum 01/2005; 14:15-21

[9] - Erhard Rahm, Hong Hai Do – „Data Cleaning: Problems and Current Approaches", IEEE Data Engineering Bulletin, Volume 23, 2000

[10] "Oracle Warehouse Builder 10gR2 Transforming Data into Quality Information", January 2006. Available: http://www.oracle.com/technetwork/developer-tools/warehouse/transforming-1.pdf

[11] E. Borowski, H.-J. Lenz, "Design of a workflow system to improve data quality using Oracle Warehouse Builder", Journal

of Applied Quantitative Methods, vol. 3, no. 3, pp. 198-206, Fall 2008.
[12] "Oracle Warehouse Builder User's Guide 10g Release 2 (10.2.0.2)", April 2009, http://docs.oracle.com/cd/B31080_01/doc/owb.102/b28223.pdf
[13] "Oracle Warehouse Builder Data Modeling, ETL, and Data Quality Guide 11g Release 2 (11.2)". Available: http://docs.oracle.com/cd/E11882_01/owb.112/e10935/data_profiling.htm
[14] "Oracle Warehouse Builder User's Guide 11g Release 1 (11.1)", July 2007. Available: http://isu.ifmo.ru/docs/doc111/owb.111/b31278.pdf
[15] G. Svolba, "Data Quality for Analytics Using SAS". SAS Press, 2012, pp. 182-192.
[16] S. J. Nowlin. "Data Profiling Using Base SAS® Software: A Quick Approach to Understanding Your Data". SUGI 31 Proceedings. San Francisco, California March 26-29, 2006. Available: http://www2.sas.com/proceedings/sugi31/161-31.pdf
[17] P. Welbrock. "Validating And Updating Your Data Using SAS Formats". NESUG 2001, Baltimore, USA, 2001.

Anca Ioana ANDREESCU, PhD is an associate professor at the Bucharest University of Economic Studies, Faculty of Cybernetics, Statistics and Informatics, Department of Economic Informatics and Cybernetics. She published over 20 articles in journals and magazines in computer science, informatics and business management fields, over 30 papers presented at national and international conferences, symposiums and workshops. In January 2009 she finished the doctoral stage, the title of her PhD thesis being: The Development of Software Systems for Business Management. She is the author of one book and she is coauthor of five books. Her interest domains related to computer science are: requirements engineering, business analytics, modeling languages, business rules approaches and software development methodologies.

Anda BELCIU has graduated the Faculty of Economic Cybernetics, Statistics and Informatics of the Bucharest Academy of Economic Studies, in 2008. She has a PhD in Economic Informatics and since October 2012 she is a Lecturer. She teaches Database, Database Management Systems and Software Packages seminars and courses at the Economic Cybernetics, Statistics and Informatics Faculty. She is co-author of 4 books, has 11 articles published in prestigious journals included in international recognized databases (SCOPUS, Elsevier, EBSCO, ProQuest, or DOAJ) and also 17 papers in the volumes of national and international scientific manifestations, of which 4 are indexed Thomson ISI Web of Science. Her scientific fields of interest and expertise include database systems, e-business, e-learning, spatial databases. She has experience in 5 research projects, participating as a team member.

Alexandra Maria Ioana FLOREA has graduated from the Faculty of Cybernetics, Statistics and Economic Informatics of the Academy of Economic Studies in 2007 and also from the Faculty of Marketing in 2008. Since then she is a PhD candidate, studying to obtain her PhD in the field of economic informatics. At present she is assistant lecturer at the Academy of Economic Science from Bucharest, Economic Informatics Department and her fields of interest include integrated information systems, information

system analysis and design methodologies and database management systems.

Vlad DIACONITA is a member of the IEEE and INFOREC organizations and member of the technical team of the Database Systems Journal. As part of the research team he has worked in 8 different phases of 3 UEFISCDI funded grants. He has published more than 30 papers in peer reviewed journals and conference proceedings, many indexed in ISI or SCOPUS. He is the co-author of four books.