DESIGN OF A WORKFLOW SYSTEM TO IMPROVE DATA QUALITY USING ORACLE WAREHOUSE BUILDER[1]

Esther BOROWSKI
Institut für Informatik, Freie Universität Berlin, Germany

E-mail: Esther.Borowski@fu-berlin.de

Hans-J. LENZ
PhD, University Professor, Institut für Statistik und Ökonometrie,
Freie Universität Berlin, Germany

E-mail: Hans-J.Lenz@fu-berlin.de

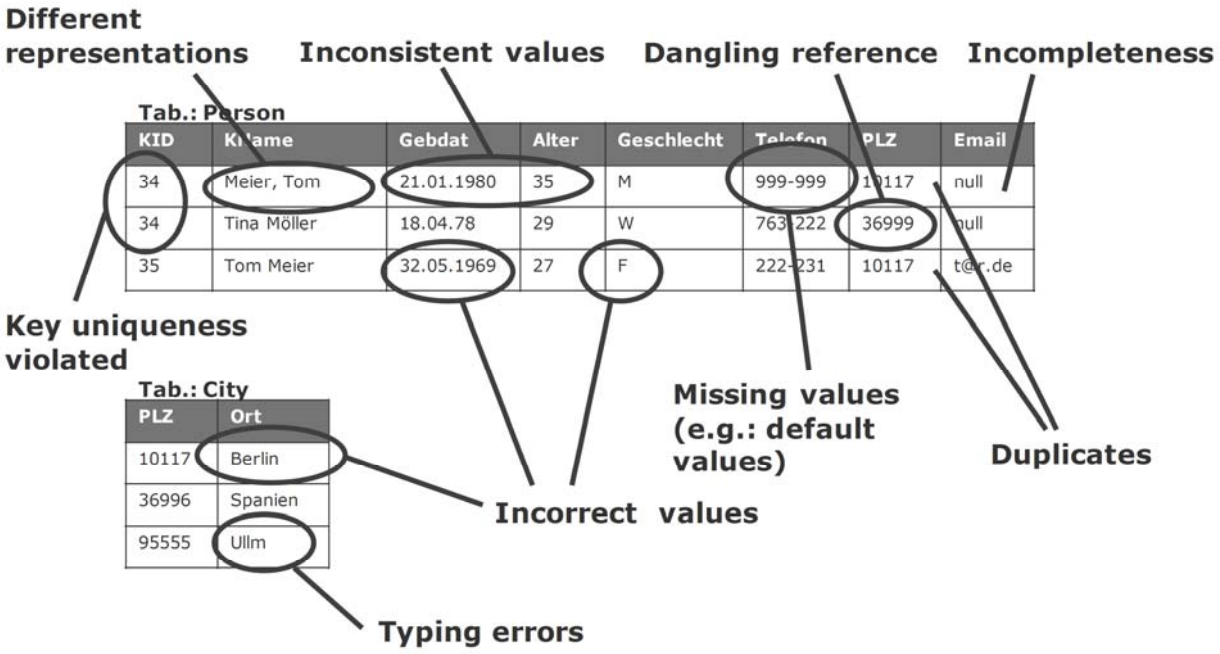
Abstract: We are concerned with the development of a workflow system for data quality assurance. The approach is based on the Oracle Warehouse Builder (OWB) in its latest available version. We motivate the data quality problem mainly companies are faced to, discuss some basics about data quality assurance of (mostly) non-metric data, and introduce our proposal for a workflow machine. We very shortly present some striking features, i.e. the architecture and data quality cycle of the Oracle Warehouse Builder having set the "Data Quality Option". Next we report about our software DaRT , an acronym representing Data Quality Reporting Tool. We close with some remarks and perspectives for ongoing research.

Key words: data quality; Oracle Warehouse Builder; OWB; DaRT

1. Motivation and Basics

Given a database and/or data at the data entry a data conflict exists if the data collected is not fit for use related to purposes and quality targets of a given company. Such data conflicts are represented in Table 1 below.

Table 1. Typical data conflicts arising in the business domain



The big trouble makers are errors, missing values as a kind of incompleteness and duplicates. We can classify the data conflicts according to conflicts which exist in the schema, i.e. exist as schema conflicts in the database schema – even before any data entry, and data conflicts which are bounded to the data set itself. We can summarize the "problem sphere" according to the following so called dimensions of data quality; cf. Batini & Scannapieco (2006):

- Accuracy;
- Completeness;
- Time related dimension:
  - currency;
  - volatility;
  - timeliness;
- Consistency.

While computer scientists like Chrisman (1984) coined the slogan "Data Quality is Fitness for use", we prefer the extended version saying "Data Quality is Fitness for use given a purpose". Alternatively, we may stick to Wertz (1915): "Quality reflects the ability of an object to meet the purpose". The purpose of data quality control is mandatory. It would be rubbish to first compute business or economic aggregates and then locate errors in the derived figures and reject them. Validation rules are a generic part of metadata management and are mandatory for any modern statistical information system or, synonymously, data warehouse.

2. Design of a Workflow System for Data Quality Improvement

There exists a lot of experience and self-evidence that data quality of economic or business information systems based on modern relational databases can be achieved by data quality checks, i.e. semantic rules, statistical tests, and validation rules (logical and numerical edits).

- Semantic Checks
  - Inspection /Comparing of real data against metadata
  - Ex.: Estimation of the completeness of an attribute (simple ratio)

$$Q_V(A) := 100 \cdot \frac{\sum_{i=1}^{|A|}(null(w_i))|}{|A|}$$

$$null(w) := \begin{cases} 0, & \text{falls w} = \text{Null} \\ 1, & \text{sonst.} \end{cases}$$

- Statistical Measures
  - Exploration and Analysis of a data set or database by descriptive statistics like (Minimum, Maximum, arithmetic Mean, Median, Standard Deviation, (quite obscure) Six Sigma Rule as preferred by non statisticians etc.)

- Validation Rule Sets
  - Checking of real data using business or economic rules of type „if…then…"

- Edits as simple, logical or numerical statements, cf. Boskovitz (2008), as well as statistical or fuzzy logic based edits, cf. Köppen and Lenz (2006).

The coding of checks, statistical measures and edits for DQ reporting is simple but very tedious. We give below an example of the PL/SQL source code for a very quick and dirty outlier detection of attribute x using a three-sigma rule as $|x\text{-}avg| > 3\sigma$.

```
DECLARE
        l_prj VARCHAR2(4000) := NULL;
        l_sql VARCHAR2(4000) := NULL;
BEGIN
l_sql := 'SELECT *
          FROM '|| :P_TABLENAME ||'
          WHERE '|| :P_COLUMN ||' >
          (
                SELECT AVG_VALUE + (3 * STDDEV_VALUE)
                FROM ALL_IV_PROFILE_COLUMNS          /* Public View */
                WHERE COLUMN_ID = '|| :P_COLUMNID ||'
          )
```

```
                OR '|| :P_COLUMN ||' <
                (
                        SELECT AVG_VALUE - (3 * STDDEV_VALUE)
                        FROM ALL_IV_PROFILE_COLUMNS              /* Public View */
                        WHERE COLUMN_ID = '|| :P_COLUMNID ||'
                )
                ';
RETURN l_sql;
END;
```

If any data quality management is trying to do so the question arises how to proceed, and what kind of road map exists to "produce data quality. The straight fort answer to this question is using a workflow machine for data quality improvements. We sketch in Figure 1. a taxonomy of data quality (DC) criteria which forms the basement of our proposal of such a workflow.

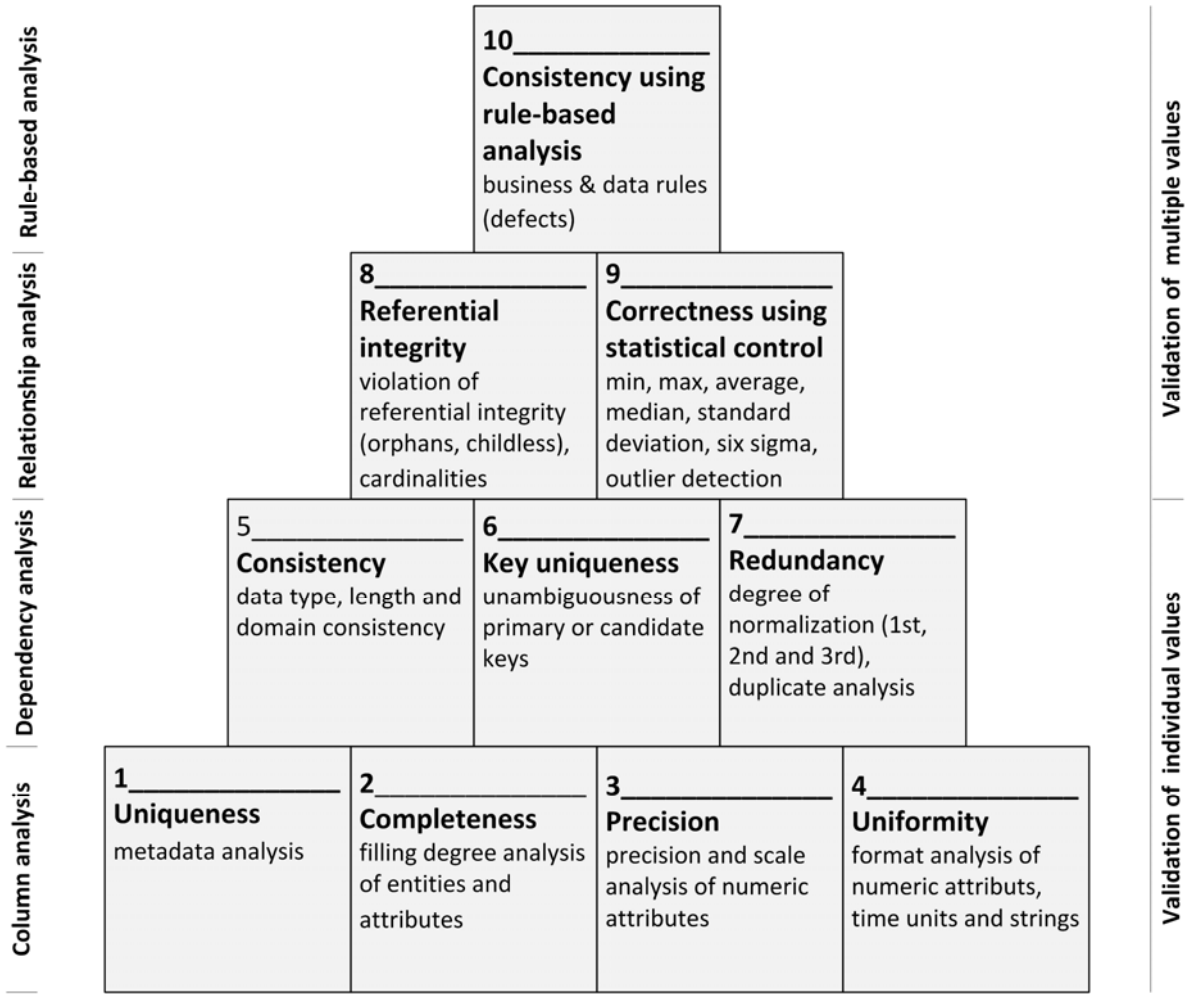


Figure 1. Taxonomy of data quality criteria used in DaRT – a new data quality reporting tool

3. Oracle Warehouse Builder (OWB) with "Data Quality Option"

As our DQ workflow machine is an added-on to the Oracle Warehouse Builder we show below a diagram which explains the architecture to some degree. Of course, further details would need more pages which are limited in this paper due to given restrictions.

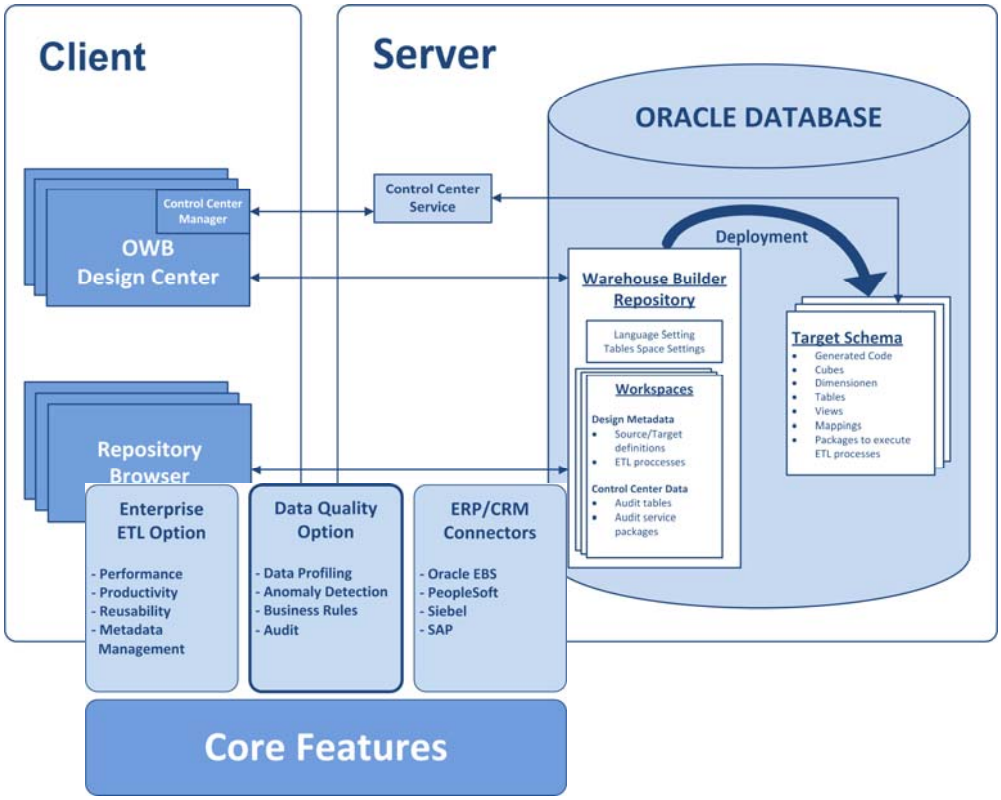


Figure 2. OWB Architecture with Data Quality Option (Source: Oracle 2007a)

It is evident from Figure 2. that DaRT is embedded into a client server architecture. This does not give any hints which steps are necessary for data quality improvement. Moreover, the archi-tectture is perpendicular on how these steps are sequenced. This view on our DQ engine is represented in the following chapter**.**

4. The Data Quality Cycle

Figure 3. represents our proposal of a data quality circle.

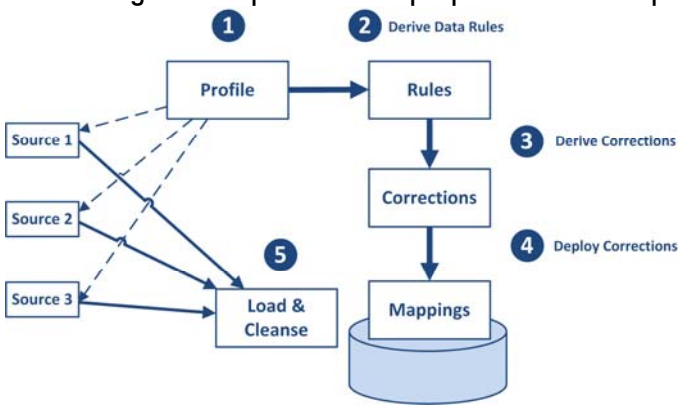


Figure 3. Data Quality Cycle of DaRT within OWB

The five steps are to some extent self evident. According to our definition of DQ as "Fitness for use given a Purpose" we start with a target and profiling on stage 1. Then the rules to be used come in a stage 2, followed by error location and correction in step 3-4, and finally by loading and cleansing data in step 5.

5. The Data Quality Reporting Tool - DaRT

The next point we have to consider is the extent of data quality activities or the functionality of DQ covered by DaRT. For the sake of compactness of discussion we present just a diagram which, as we believe, says more than one thousand words.

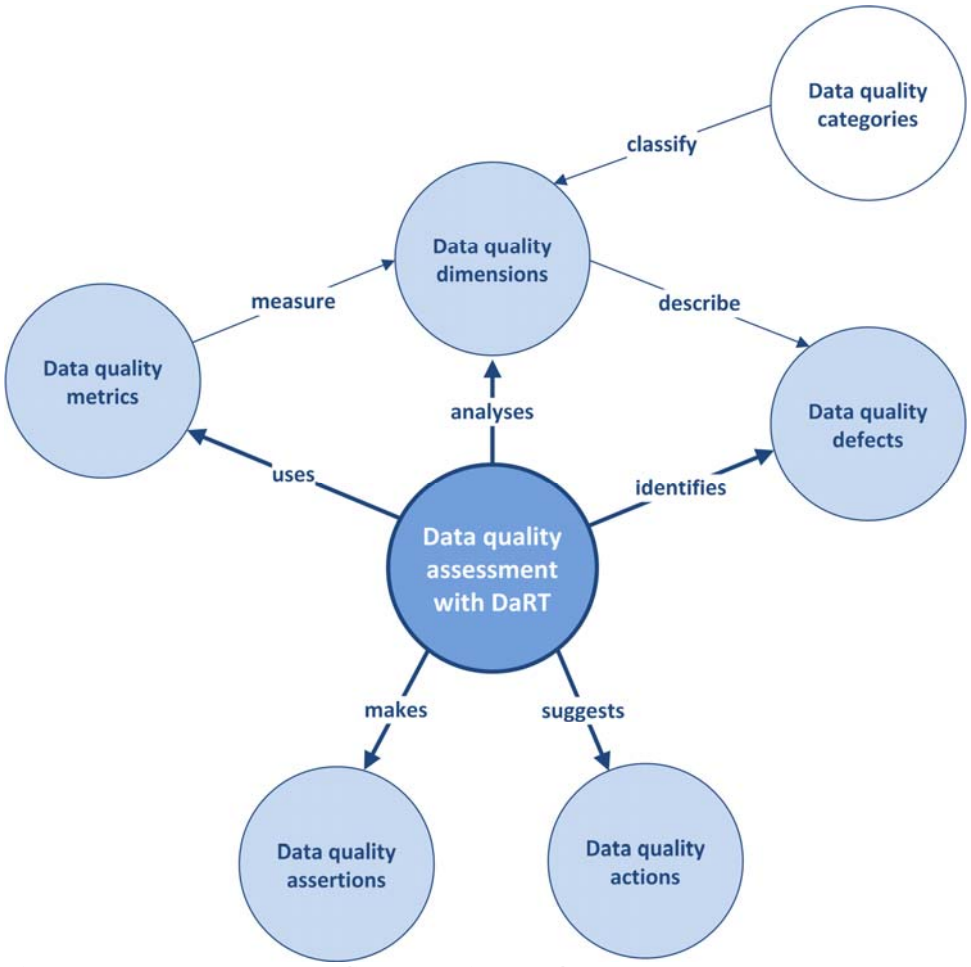


Figure 4. DaRT – Data Quality Reporting Tool

6. The Meta Database Model for Data Quality

Our workflow engine "DaRT" is not completely explained without displaying the conceptual schema of the underlying meta database system. The entity types are to some extent self evident. We have a relation with attributes, quality characteristics and quality dimensions or categories. The quality indicators have a so called "quality metric", and are used for metering data quality, accordingly. Of course, data quality specifications ("purpose" of data quality above), statements or judgements about current data quality of a table, and quality actions are to be considered further. We collect these views together in the conceptual schema as "DaRT inside" underpinning the data quality engine in Figure 5.

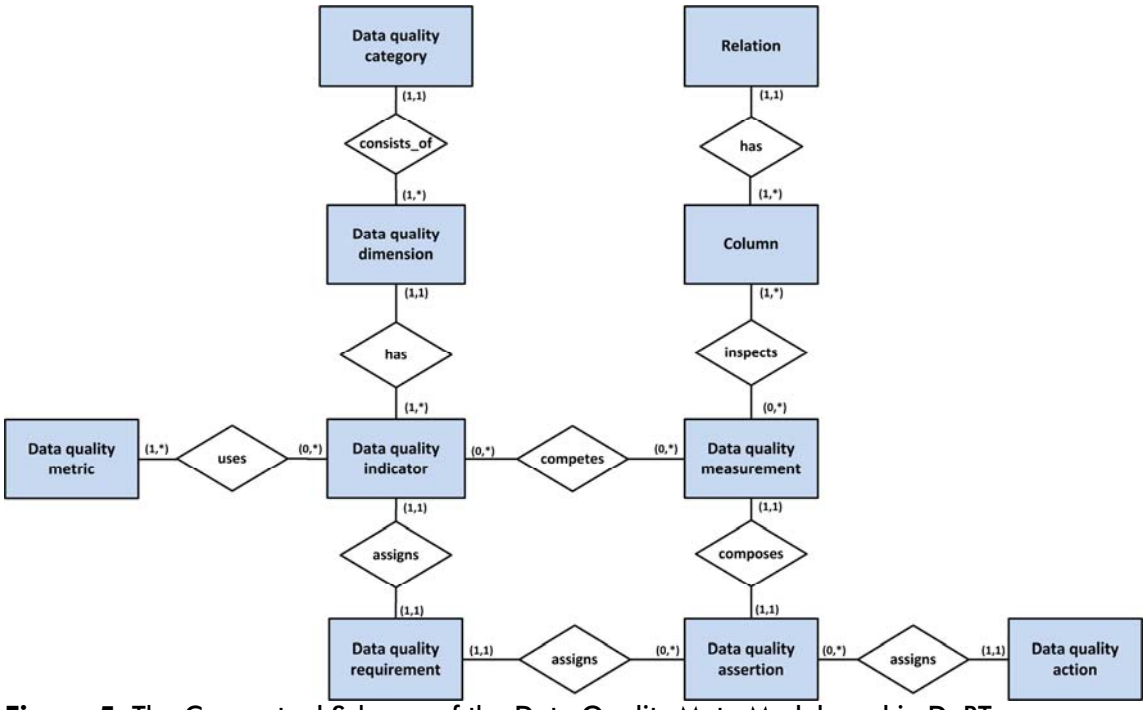


Figure 5. The Conceptual Schema of the Data Quality Meta Model used in DaRT

The last question the interested reader may raise is about the IT technologies used so far. Figure 6 represents what was used to implement DaRT.

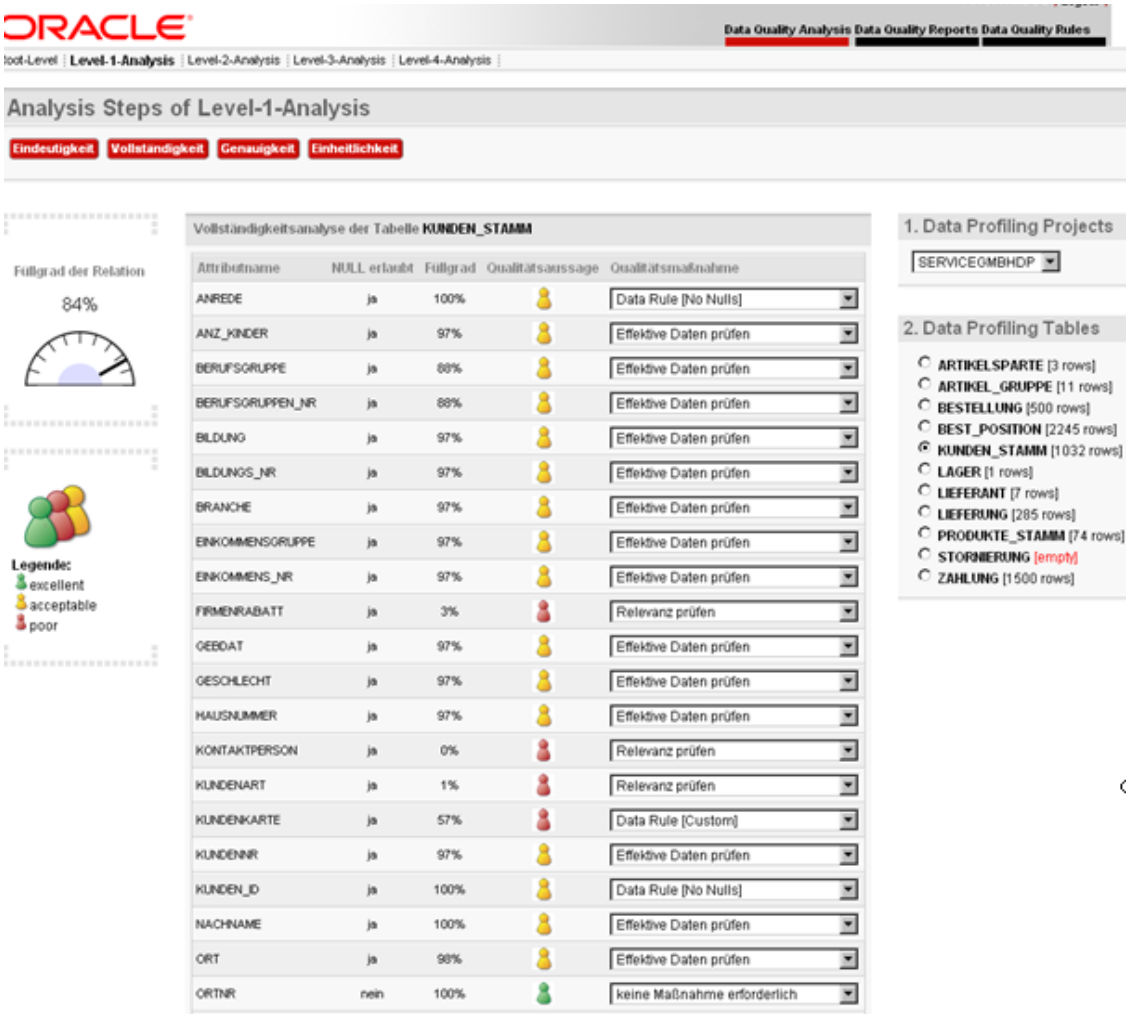
7. The used IT Tools of DaRT

The used IT tools for DaRT are the followings:

- ETL-Tool (Data Extraction, Integration, Loading)
  - Oracle Warehouse Builder
- Web Application Framework
  - Application Express (APEX)
- Programming Languag
  - PL/SQL
- Database Developer
  - SQL Developer (incl. SQL*Plus)

8. The frontend for the data quality analyst

We limit ourselves with respect to the user interface (GUI) to present only part of the frontend devoted to the functionality of data quality reporting. The semantics of the reports is concer-ned wit a table "Customer" which has attributes house no, contact person, type of customer clients file, customer id etc all printed in German language in this sequence.



Attributname	NULL erlaubt	Füllgrad	Qualitätsaussage	Qualitätsmaßnahme
ANREDE	ja	100%		Data Rule [No Nulls]
ANZ_KINDER	ja	97%		Effektive Daten prüfen
BERUFSGRUPPE	ja	88%		Effektive Daten prüfen
BERUFSGRUPPEN_NR	ja	88%		Effektive Daten prüfen
BILDUNG	ja	97%		Effektive Daten prüfen
BILDUNGS_NR	ja	97%		Effektive Daten prüfen
BRANCHE	ja	97%		Effektive Daten prüfen
EINKOMMENSGRUPPE	ja	97%		Effektive Daten prüfen
EINKOMMENS_NR	ja	97%		Effektive Daten prüfen
FIRMENRABATT	ja	3%		Relevanz prüfen
GEBDAT	ja	97%		Effektive Daten prüfen
GESCHLECHT	ja	97%		Effektive Daten prüfen
HAUSNUMMER	ja	97%		Effektive Daten prüfen
KONTAKTPERSON	ja	0%		Relevanz prüfen
KUNDENART	ja	1%		Relevanz prüfen
KUNDENKARTE	ja	57%		Data Rule [Custom]
KUNDENNR	ja	97%		Effektive Daten prüfen
KUNDEN_ID	ja	100%		Data Rule [No Nulls]
NACHNAME	ja	100%		Effektive Daten prüfen
ORT	ja	98%		Effektive Daten prüfen
ORTNR	nein	100%		keine Maßnahme erforderlich

Figure 6. DQ Report on customer data using DaRT

It may be worthwhile mentioning that the default sequencing of DQ steps may be changed according to real environments and use cases. The flexibility given by DaRT is reflected by its update capabilities as represented in Figure 7.

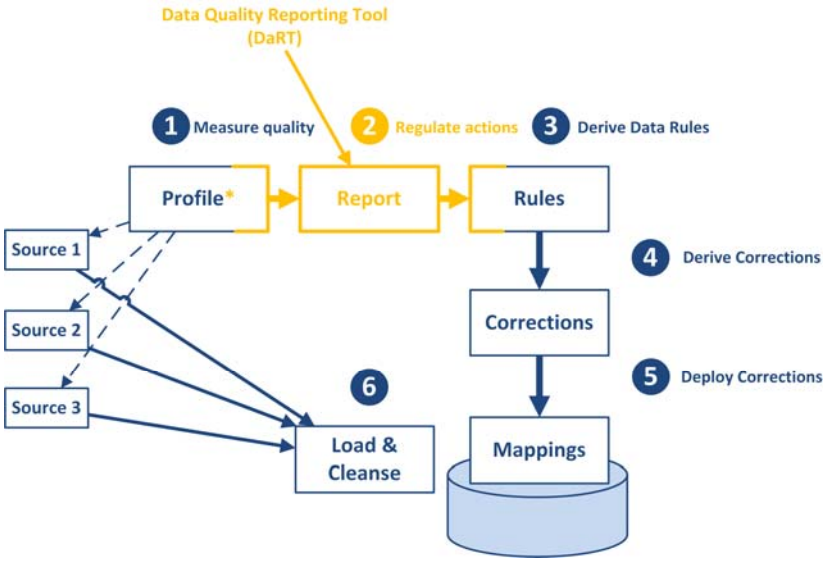


Figure 7. Updating the data quality cycle of DaRT

9. The main Results and Perspectives for DQ

The main results of designing and implementing DaRT are as follows:

- Fixing of data quality attributes (indicators) which can be measured
- Sequencing of steps of DQ analyses
- Designing a conceptual meta database model
- DQ Analysis cockpit as GUI
- Web based workflow engine.

Our perspective after completing the development of a first data quality workflow engine but before any shop floor evaluation is:

- Expanding the workflow engine wit respect a large range of edits
- Improving the conceptual Schema
- Capturing the semantics of various application scenarios
- Transferring our prototype "DaRT" into an Oracle product added on OWB.

We close with a statement coined in German language by Albert Einstein: „Es ist leichter, Datenqualitätsprobleme zu lösen, als mit Ihnen zu leben."

References

1. Hinrichs, H. **Datenqualitätsmanagement in Data-Warehouse-Systemen,** Universität Oldenburg, Diss., 2002
2. Köppen, V. and Lenz, H.-J. **A comparison between probabilistic and possibilistic model for data validation,** COMPSTAT, Rome, 2006
3. Lenz, H.-J. **Data Cleaning II,** Lecture on „Data Warehousing und Statistische Datenbanken". Freie Universität Berlin, 2007
4. Oracle, **Oracle Database - Application Express User's Guide Release 3.1.** http://download.oracle.com/docs/cd/E10513_01/doc/appdev.310/e10499.pdf, Version: 2008b
5. Oracle, **Oracle Warehouse Builder - Data Quality Option,** http://www.oracle.com/technology/products/warehouse/pdf/warehouse-builder-11g-data-qualitydatasheet.pdf. Version: 2007a
6. Oracle, **Oracle Warehouse Builder - Installation and Administration Guide 11g Release 1 (11.1) for Windows and UNIX,** http://download.oracle.com/docs/cd/B28359_01/owb.111/b31280.pdf. Version: 2008a
7. Oracle, **Oracle Warehouse Builder 11g - Examining Source Data Using Data Profiling,** http://www.oracle.com/technology/obe/11gr1_owb/owb11g_update_extend_knowledge/less2_data_profiling/less2_data_profiling.htm. Version: 2007b

[1] **Acknowledgement**
We cordially thank Alfred Schlaucher (Business Integration & Data Warehouse, ORACLE Deutschland GmbH, Hamburg) for his efficient collaboration.