Mit Oracle EDQ schnell und einfach die Datenqualität sicherstellen

Dr. Holger Dresing
Oracle Deutschland B.V. & Co. KG

Schlüsselworte:
EDQ – Enterprise Data Quality, DIS – Data Integration Solutions, ETL – Extract, Transport and Load, Data Quality, Data Profiling, Data Cleansing

Einleitung

Um konsistente und von den Anwendern akzeptierte Ergebnisse mit den Business Intelligence Lösungen zu generieren, müssen die Dateninhalte und -strukturen, die in das Data Warehouse hineinlaufen, auch hinreichend bekannt sein. Das ist umso schwieriger, da es im Unternehmen vielfältige operative Systeme verschiedener Hersteller geben kann, deren Daten untereinander abgeglichen werden müssen. Allgemein hat sich dafür der Begriff „Dataqualität“ herausgebildet. Vor allem Entscheider sehen in diesem Begriff Probleme mit ihren Daten, die ihr Geschäft behindern. Aber eigentlich geht es darum, die Daten zu analysieren (Data Profiling), die Daten zu standardisieren und abzugleichen (Data Cleansing), Dupletten zu erkennen (Data Deduplation) oder als wichtiger Einsatzfall Adressen zu verifizieren (Adress Verification).
Oracle bietet für die Sicherung der Datenqualität verschiedene Produkte an, die alle wichtigen Bereiche zur Datenqualität abdecken: Master Data Management, Produkt Data Manamgent, Enterprise Data Quality, Watchlist Screening oder Siebel UCM, um nur einige zu nennen. Für den Bereich Datenintegration und ETL gibt es aber nur ein Produkt, das Basis für die meisten vorher genannten Produkte ist: Enterprise Data Quality (EDQ) für den Oracle Data Integrator (ODI). EDQ ist das strategische Produkt für die Datenqualität und hat eine integrierte Schnittstelle zum ODI, so dass die mit ODI generierten ETL-Strecken mit EDQ die Datenqualität sicherstellen können. Im folgenden wird ausschließlich auf die EDQ Features eingegangen, die sich mit ODI verbinden lassen und typisch für ETL-Strecken sind. Anhand eines Beispiels wird erklärt, wie einfach und schnell ein Prozeß zur Datenqualität erstellt werden kann.
Es wird davon ausgegangen, dass die grundlegenden Begriffe im Umfeld von Data Quality bekannt sind bzw. es werden die bei Oracle üblichen Definitionen benutzt. Das Produkt EDQ steht im Vordergrund.

Oracle EDQ

Architektur

Oracle Enterprise Data Quality (OEDQ) ist eine Java Web Application mit einer Client/Server Architektur. Diese besteht aus:

- einem Web Start Graphical User Interfaces (Client Application),
- einem Business Layer (dem Server), der eine Java Servlet Engine benutzt und
- einem SQL RDBMS (für das Data Repository).

Zusätzlich kann EDQ mit verschiedenen Datenbanken (Datenquellen und -zielen) und Dateien (z. B. XML oder MS Excel) kommunizieren, um von dort Daten zu lesen bzw. zu schreiben.

Die Komponenten von OEDQ können auf mehreren Computern installiert werden. Damit wird auch der Lastausgleich und die Skalierbarkeit sichergestellt. Beispielsweise kann OEDQ als Single Instance auf einem Computer installiert werden, so daß alle Komponenten auf diesem einen Computer laufen. Alternativ können die einzelnen Komponenten auf verschiedenen Computern im Netzwerk installiert werden, um ausreichend Kapazität für einen Multi-user-Betrieb bereitzustellen. Bei einer solchen Umgebung könnte der Server und das Data Repository auf einer Maschine installiert werden und jeder Anwender würde von seinem Arbeitsplatzrechner auf EDQ zugreifen. Es ist jedoch auch denkbar, dass das Repository ebenfalls auf einer eigenen Maschine installiert wird.

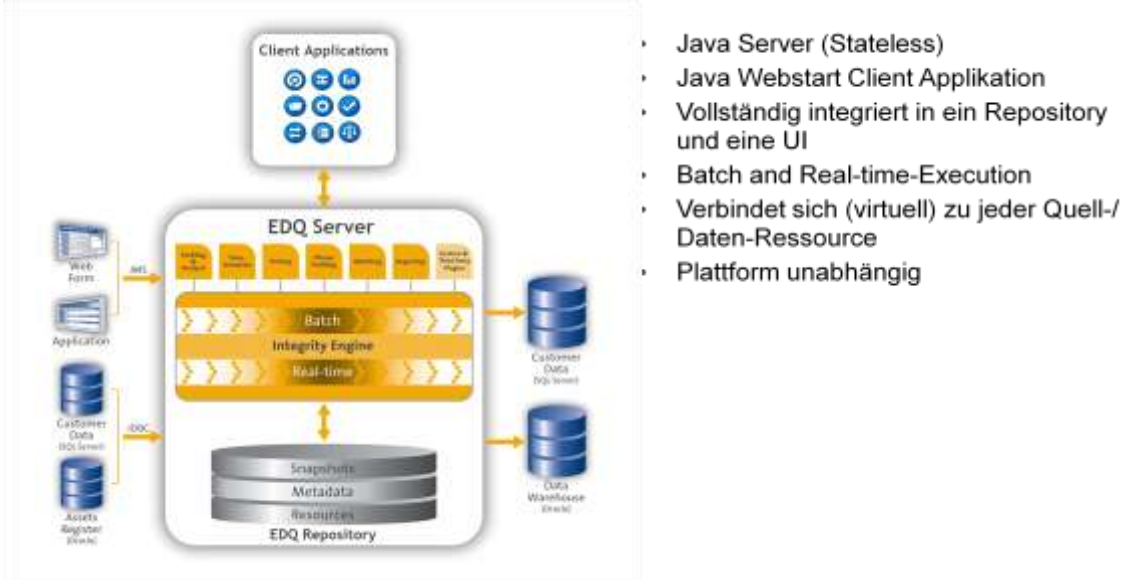


Abb. 1: Architektur von EDQ

Benutzeroberfläche

Die graphische Oberfläche von EDQ ist nicht nur für Administratoren und Entwickler gedacht, sondern auch für Fachanwender und Datenqualitätsbeauftragte im Unternehmen. Diese wissen häufig sehr viel besser als die IT-Abteilungen, wie die Daten aussehen müssen und können mit EDQ geschäftsrelevante Regeln zur Datenqualität definieren. Idee in EDQ ist, dem Fachanwender eine Datei mit Daten zu geben, damit dieser die Daten prüft und geschäftsrelevante Regeln erstellt. Die IT synchroniert anschließend die Repositories von EDQ und ODI. Ab jetzt können die Daten schon im ETL-Prozess geprüft werden und nur qualitätsgesicherte Daten fließen in das Data Warehouse.

Das Web Start Graphical User Interface stellt den Zugang zu den verschiedenen Komponenten von EDQ her: beispielsweise zur Administration und Konfiguration, zur Entwickleroberfläche dnDirector oder zu auf EDQ basierenden Anwendungen.

Der dnDirector ist das User Interface, anhand derer die Prozesse zum Sichern der Datenqualität aufgebaut werden. Die Oberfläche besteht ähnlich einem Datenintegrations- oder ETL-Werkzeug aus einem Bereich für die Projektübersicht, in der in einer Baumstruktur alle projektrelevanten Prozesse und Objekte zusammengefaßt sind, der Hauptarbeitsfläche zum Erstellen und Verwalten von DQ-Projekten, der Werkzeugpalette mit allen relevanten Werkzeugen und deren Kombinationen und der Ergebnisanzeige für die Auswertungen.

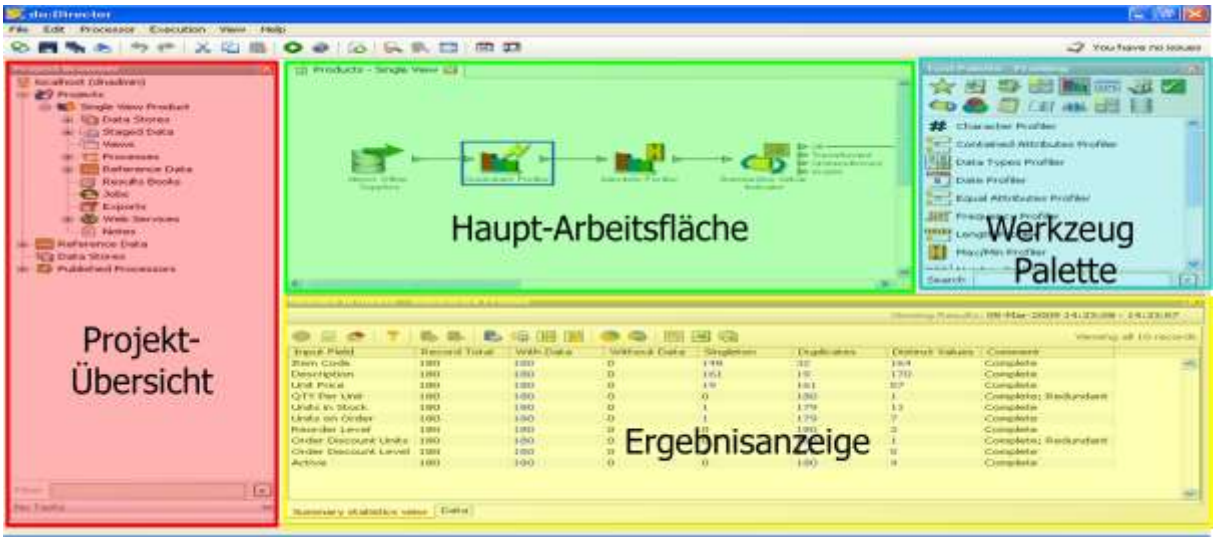


Abb. 2: dnDirector

Prozessoren

Die Struktur der Werkzeugpalette zeigt auch die wesentlichen Bereiche zum Aufbau von Datenqualitätsregeln. In EDQ werden diese zu Projekten zusammengefaßt. Projekte bestehen aus Prozessoren (engl. Prozessors), die nicht nur die Qualitätsregeln enthalten, sondern auch Schritte zum Lesen und Schreiben der Daten und deren Auswertungsergebnisse, zum Analysieren und zum Aufbereiten der Daten.

„Profiling Prozessoren“ helfen die Daten zu verstehen, d. h. die technischen Strukturen zu erfassen, Probleme in den Daten zu analysieren und geschäftliche Inkonsistenzen zu erkennen. Profiler beschreiben keine Geschäftsregeln und führen keine Änderungen durch. Sie dienen lediglich zum Analysieren der Daten. Profiler werden häufig zu Beginn einer Datenanalyse eingesetzt, können jedoch auch helfen, die Ergebnisse der Analysen besser darzustellen.

„Audit Prozessoren“ (auch Checks) stellen Geschäftsregeln dar, die die Quelldaten prüfen. Sie kategorisieren jeden Eingabesatz und geben zurück, ob er gültig oder ungültig ist. Bei einigen Prozessoren wie „List Check“ kann auch „unkannt“ zurückgegeben werden, d. h. der Datensatz konnte nicht klassifziert werden.

„Transformation Prozessoren“ transformieren die Eingabeattribute und geben sie in neue Attribute aus. Es ist wichtig zu verstehen, dass die Eingabeattribute niemals direkt geändert werden, der Anwender kann die Attribute vergleichen, bevor sie im Verarbeitungsprozess weiter laufen. Transformation Prozessoren werden bei der Migration in andere Systeme oder bei der Analyse der Daten für die Datenqualität, z. B. beim „auditing“ oder „matching“ eingesetzt. EDQ erlaubt es, die Eingabedaten aus der Quelle direkt zu analysieren, z. B. nicht valide Werte für ein Attribut. Diese können dann mit einer Referenzmenge verglichen werden, die invaliden Werte werden ersetzt und anschließend wird ein neues Attribut mit korrigierten Werten erzeugt.

„Match Prozessoren“ gleichen verschiedene Datensätze miteinander ab. Es gibt verschiedene Typen von Match Prozessoren: zum Dedublizieren, zum Abgleichen gegenüber verschiedenen Datenmengen, zum Konsolidieren oder zum Gruppieren. Dabei wird in immer der gleichen Reihenfolge vorgegangen: Identifiziere die Datenmenge und deren relevante Attribute, bilde daraus Cluster für die Abgleiche, erstelle Regeln zum Bearbeiten, wende die Regeln auf die Daten an und führe die Daten abschließend zusammen (Merge).

„Parser Prozessoren“ helfen dem Anwender, die Attribute genauer zu verstehen und zu transformieren. Benutzt werden diese Prozessoren z. B. für Namen, Adressen oder Produktbezeichnungen. Der Parser Profiler analysiert die Attribute und generiert daraus eine Liste von Token. Mit diesen Token können die Eingabedaten analysiert werden, um daraus anwendungsspezifische Regeln zu erstellen.
Die Prozessoren können auf einzelne oder mehrere Attribute angewendet werden. Gruppen von Prozessoren können zu Templates (in EDQ als Package bezeichnet) zusammengefasst und wiederbenutzt werden. Diese können generische Parameter oder auf Referenzdaten verweisen, die regelspezifisch mit Daten versorgt werden.

Einfach und schnell entwickeln anhand eines Beispiels

Nachfolgend wird ein Prozess in EDQ vorgestellt. Es werden alle Prozessortypen sowie Packages benutzt. In Abb. 3 wird der gesamte Prozess gezeigt. Nachfolgend werden nur noch relevante Ausschnitte der Benutzeroberfläche in den Abbildungen dargestellt. Es geht um das Projekt „Cloud Auto Demo“ und dort um den Prozess „Cloud Auto Customers“. Als Erstes ist die Datenquelle (Data Store) zu spezifizieren. In dieser Demo ist die Quelle eine csv-Datei. Die Daten werden als Tabelle dargestellt. Ein Data Store kann auch eine Datenbank sein und mehrere Tabellen umfassen. Damit die Daten aus den Tabellen benutzt werden können, ist ein Snapshot unter „Staged Data“ zu erstellen. In einem Snapshot können zusätzliche Filter für die Daten festgelegt werden, der Datenbestand kann für Testfälle reduziert werden, z. B. kann durch einen Parameter der Datenbestand auf 30% aller Datensätze beschränkt werden oder durch ein SQL Statement wird die Datenmenge definiert.

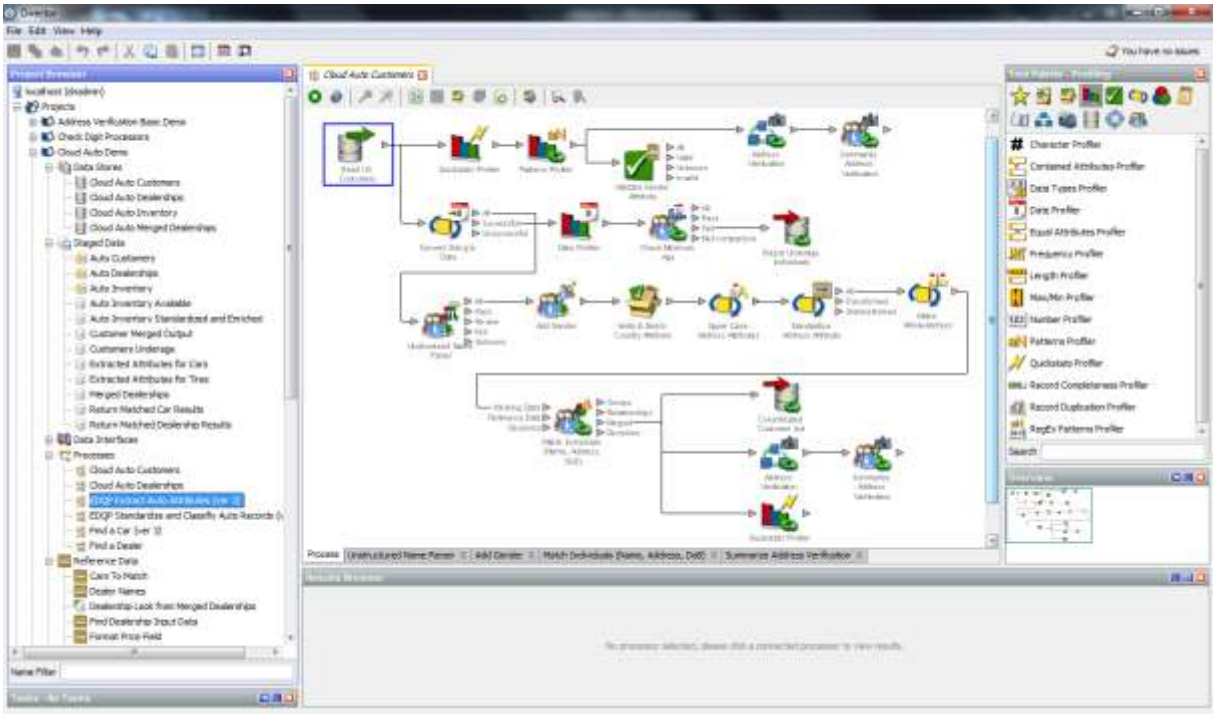
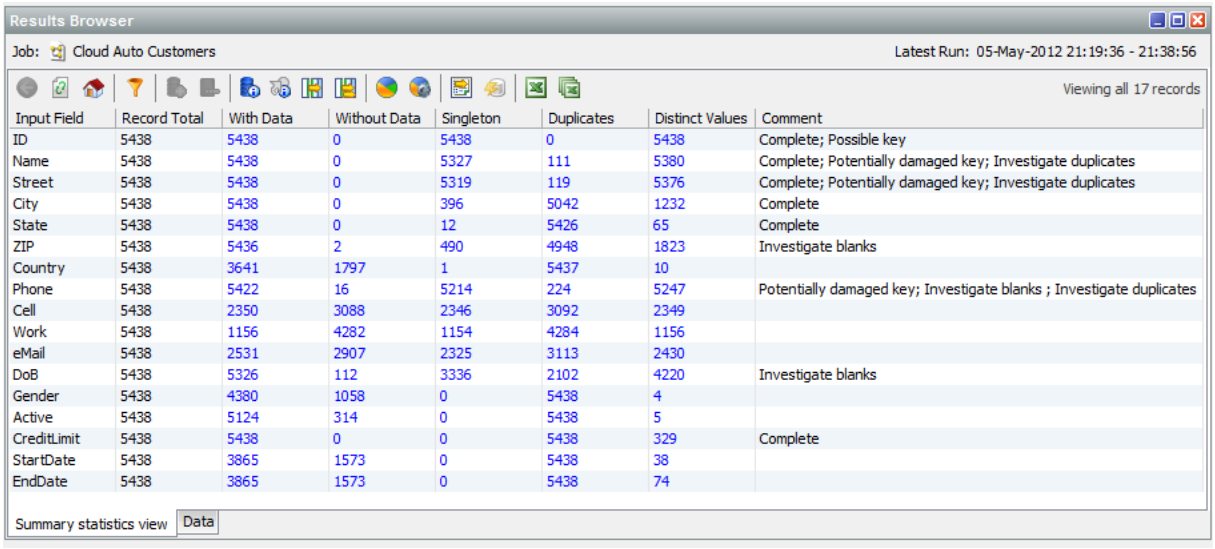


Abb. 3: Beispiel im dnDirector von EDQ

Jeder Prozess beginnt mit einem Reader Prozessor. Danach teilt sich die Verarbeitung in zwei Abschnitte. Hier wird zunächst der Teil mit den Profiling Prozessoren betrachtet. Es läuft ein Quickstat Profiler und anschließend ein Pattern Profiler, um die Daten zu analysieren.



Input Field	Record Total	With Data	Without Data	Singleton	Duplicates	Distinct Values	Comment
ID	5438	5438	0	5438	0	5438	Complete; Possible key
Name	5438	5438	0	5327	111	5380	Complete; Potentially damaged key; Investigate duplicates
Street	5438	5438	0	5319	119	5376	Complete; Potentially damaged key; Investigate duplicates
City	5438	5438	0	396	5042	1232	Complete
State	5438	5438	0	12	5426	65	Complete
ZIP	5438	5436	2	490	4948	1823	Investigate blanks
Country	5438	3641	1797	1	5437	10	
Phone	5438	5422	16	5214	224	5247	Potentially damaged key; Investigate blanks ; Investigate duplicates
Cell	5438	2350	3088	2346	3092	2349	
Work	5438	1156	4282	1154	4284	1156	
eMail	5438	2531	2907	2325	3113	2430	
DoB	5438	5326	112	3336	2102	4220	Investigate blanks
Gender	5438	4380	1058	0	5438	4	
Active	5438	5124	314	0	5438	5	
CreditLimit	5438	5438	0	0	5438	329	Complete
StartDate	5438	3865	1573	0	5438	38	
EndDate	5438	3865	1573	0	5438	74	

Abb. 4: Result Browser des Quickstat Profilers

Danach folgt eine weitere Verzweigung. Im ersten Zweig wird mit einem List Prozessor gegen einen Referenz-Datenbestand (in EDQ: Reference Data) geprüft, ob die Spalte „gender“ gültige oder ungültige Einträge enthält. Der andere Zweig führt eine Adressprüfung durch. Dafür gibt es eine Adress Verification Option, die mit einem Standard Rule Set für das jeweilige Land eine Prüfung der Adresse auf Plausibilität durchführt. In EDQ gibt es nach dem Installieren dieser Option einen Prozessor Adress Verification. Über eine Schnittstelle sind lediglich die benötigten Adressdaten zu definieren und EDQ generiert daraus einen passenden Adressdatensatz einschließlich des Längen- und Breitengrades.

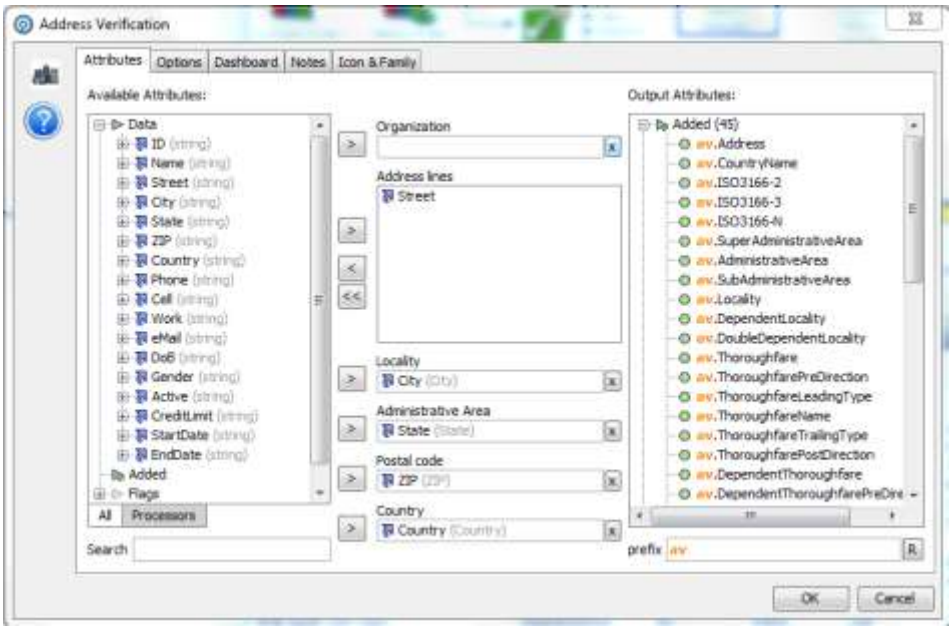


Abb. 6: Adress Verification

Anschließend folgt ein Schritt „Summarize Adress Verification“ mit statistischen Angaben zu den Adressen. Dies ist ein Package (Template), das verschiedene Transformationen und anschließend einen Frequency Profiler ausführt.

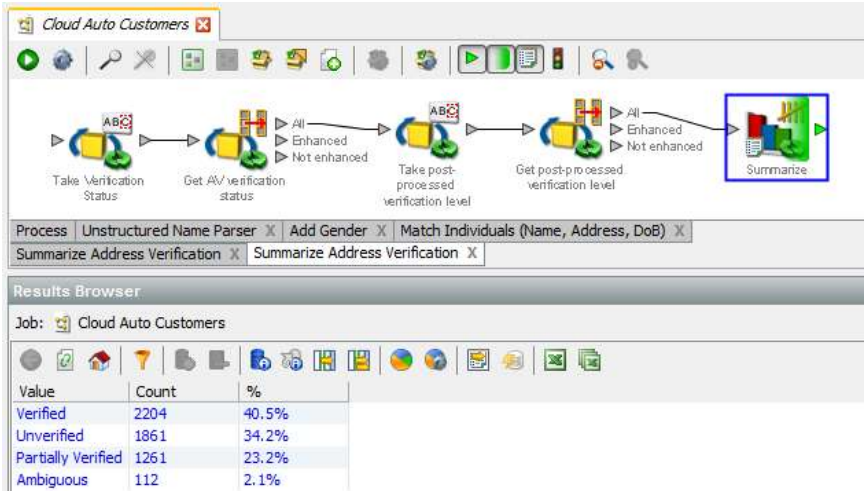


Abb. 7: Package „Summarize Address Verification“

Zurück geht es zum ursprünglichen Reader Prozessor. Der zweite Zweig beginnt mit einer Transformation „convert string to date“. Die Quelle war ein Flat File und damit wird der Datumswert in der Spalte DoB (Date of Birth), der eine Zeichenkette (string) ist, in einem Datumswert für eine Datenbank konvertiert. Jeder Prozessor hat passende Outputparameter, die den weiteren Datenfluß steuern. An dieser Stelle gibt es die Outputparameter: „all“, „successful“ oder „unsuccessful“, d. h. es wird mit allen Daten, nur mit den erfolgreichen oder den nicht erfolgreichen in einen Datumswert konvertierten Daten weiter gearbeitet.
Anschließend geht es mit den erfolgreich konvertierten Datensätzen in den Date Profiler und danach in ein Package „check minimum age“. Dort wird das aktuelle Datum hinzugefügt, die Differenz zwischen dem Geburtstag des Kunden und dem aktuellem Datum gebildet und dann alle Kunden aussortiert, bei dem die Differenz kleiner als 18 ist. Dafür wurde der Writer Prozessor eingefügt.

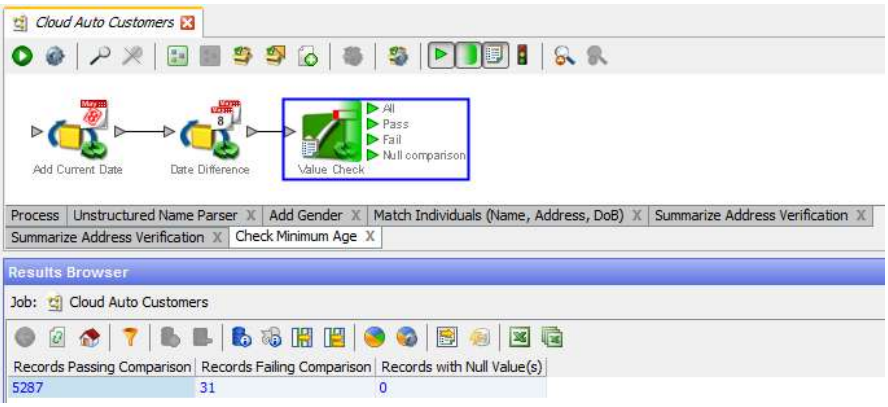
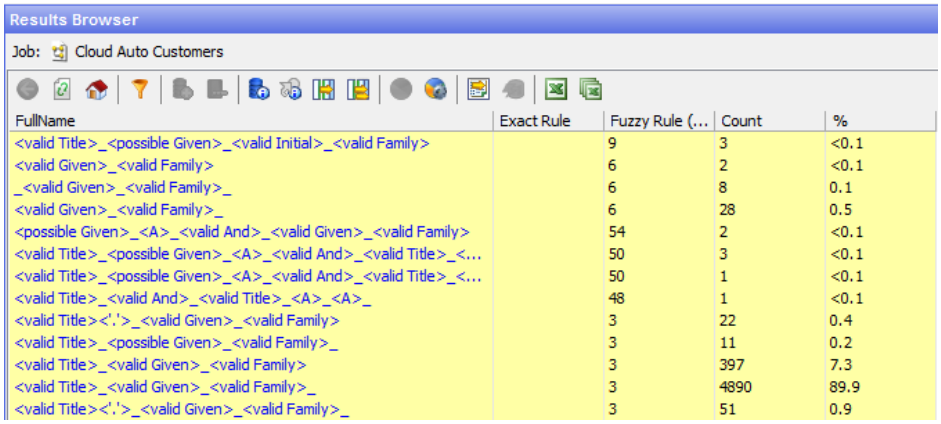


Abb. 8: Altersermittlung durch „check minimum age“

In einem weiteren Zweig wird das Feld „Name“ mit einem Parser analysiert. Der Parser bildet Token, die das Feld genauer analysieren. Der Name kann aus Titel, Vorname oder Familienname bestehen. Es können auch mehrere Personen dort eingegeben sein. Daraus werden Regeln (Fuzzyregeln) gebildet, und der Name wird in mehrere Felder aufgeteilt.



FullName	Exact Rule	Fuzzy Rule (...	Count	%
<valid Title>_<possible Given>_<valid Initial>_<valid Family>		9	3	<0.1
<valid Given>_<valid Family>		6	2	<0.1
<valid Given><valid Family>_		6	8	0.1
<valid Given>_<valid Family>_		6	28	0.5
<possible Given>_<A>_<valid And>_<valid Given>_<valid Family>		54	2	<0.1
<valid Title>_<possible Given>_<A>_<valid And>_<valid Title>_<...		50	3	<0.1
<valid Title>_<possible Given>_<A>_<valid And>_<valid Title>_<...		50	1	<0.1
<valid Title>_<valid And>_<valid Title>_<A>_<A>_		48	1	<0.1
<valid Title><'.'>_<valid Given>_<valid Family>		3	22	0.4
<valid Title>_<possible Given>_<valid Family>_		3	11	0.2
<valid Title>_<valid Given>_<valid Family>		3	397	7.3
<valid Title>_<valid Given>_<valid Family>_		3	4890	89.9
<valid Title><'.'>_<valid Given>_<valid Family>_		3	51	0.9

Abb. 9: Fuzzyregeln im Parser

Der Prozess geht über mehrere Prozessoren und Packages weiter. Dabei wird das Feld „Gender“ geprüft, es werden die Ländernamen (Country) standardisiert und verschiedenen Felder in Großbuchstaben umgewandelt.
Anschließend kommt ein Match Prozessor zum Dedublizieren. Im Match Prozessor werden zunächst Cluster über die zu vergleichenden Attribute gebildet und mit diesen Attributen werden dann die Matching-Regeln erstellt. In diesen Regeln kann auf Gleichheit von Attributen abgefragt werden, es ist jedoch auch möglich, weniger scharfe zu definieren. Beispielsweise können zwei Namen als „gleich“ oder „zu prüfen“ angesehen werden, wenn sie sich nur um 2 Zeichen unterscheiden oder es werden die Leerzeichen in zwei Felder nicht zur Prüfung herangezogen.

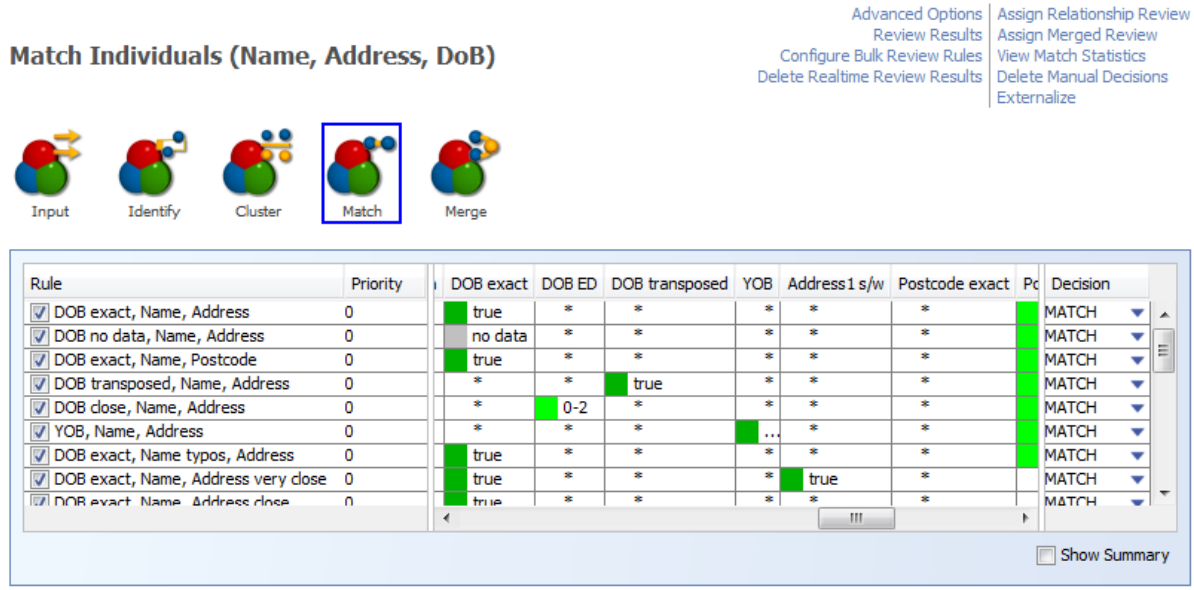


Abb. 10: Prüfregeln und deren Inhalte in einem Match Prozessor

Zum Abschluß gibt es nochmal 3 Zweige: im ersten wird die konsolidierte Liste ohne die Dupletten ausgegeben, im zweiten läuft noch einmal eine Adress Verifikation über die konsolidierte Liste und anschließend werden statistische Informationen über die Anzahl der modifizierten oder nicht zuordnenbaren Datensätze ausgegeben und im dritten Zweig läuft nochmal ein Quickstats Profiler über die Daten, um die Änderungen zwischen den Eingangsdaten und den qualitätsgesicherten Ausgangdaten zu dokumentieren.

Integration in ODI

In ODI gibt es die Möglichkeit, Constraints zu formulieren, Attribute zu transformieren, neue Attribute zu erstellen und Daten in Fehlertabellen zu kopieren. Diese Funktionen werden häufig mit denen von EDQ verglichen. Aber in ODI ist der Anwender immer von den Möglichkeiten der Technologie wie z. B. der eingesetzten Datenbank abhängig und er muß seine Regeln in SQL formulieren. Insbesondere für Fachanwender wird es dann schwierig. Aber ODI unterstützt eine Vielzahl von unterschiedlichen Quell- und Zielsystemen und ODI kann mit großen Datenmengen umgehen.
In EDQ gibt es eine graphische Benutzeroberfläche, die auch für fachliche Benutzer schnell und einfach zu erlernen ist. Für alle relevanten Transformationen und Prüfungen gibt es graphische Prozessoren, die über Parameter ohne Programmierung benutzt werden können. Beispielsweise gibt es in EDQ die Möglichkeit, bei Vergleichen von Zeichenketten nicht nur die Groß-/ Kleinschreibung zu berücksichtigen, sondern gleichzeitig auch noch festzulegen, dass Leerzeichen ignoriert werden sollen und bis zu 3 Zeichen Differenz akzeptabel sind und die Zeichenketten als gleich oder zu X% gleich angesehen werden.

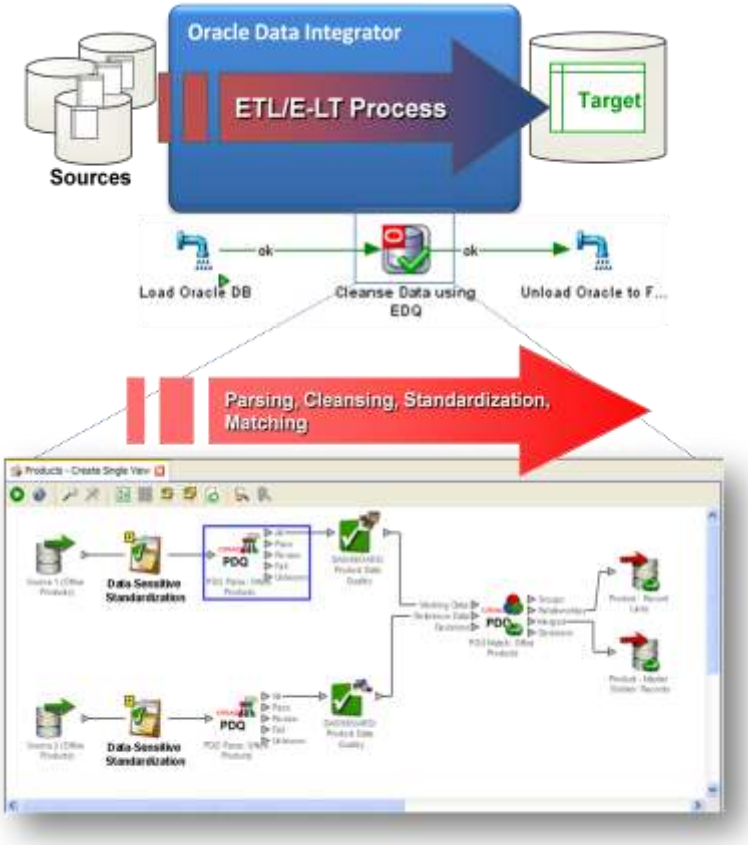


Abb. 11: Integration von EDQ in ODI

Beide Produkte haben also ihre Stärken, sie können aber auch in Kombination eingesetzt werden. Ab ODI 11.1.1.6.x gibt es in den Packages von ODI ein „EDQ Tool“, das EDQ im Batch aufruft und dort Prozesse ausführt. Die Ergebnisse liegen in der Repository-Datenbank von EDQ und können mit ODI-Mitteln weiter verarbeitet werden. Mit einem BI Tool können die Ergebnisse von EDQ mit den Daten verknüpft werden. Die EDQ internen Reportingwerkzeuge können zwar die Ergebnisse anzeigen,

diese können aber nicht mit den Quelldaten verknüpft werden. Falls EDQ die Technologie der Quell- oder Zielsysteme nicht untersützt, kann man ODI nutzen, um die Daten in das Repository von EDQ zu laden, dort die DQ Prozesse mit EDQ ausführen (siehe Abb. 11) und anschließend die Ergebnisse mit ODI weiter verarbeiten. Somit hat jedes Tool seine Stärken: ODI für heterogene Umgebungen mit performanceoptimierten Zugriffen, EDQ für komplexe Operationen für DQ Prozesse.

Fazit

EDQ ist ein Java-basiertes Werkzeug, das auf einem Applikationsserver arbeitet. Der Fachanwender kann einfach und schnell komplexe Datenqualitätsregeln erstellen und nutzen. Es stellt sich die Frage, wieviele Datensätze mit ODI eigentlich verarbeitbar sind. Der letzte veröffentlichte Benchmark hat folgende Situation simuliert: Auf einen fiktiven Adreßbestand mit 20 Mio. Kunden/Datensätzen wurden die folgenden Schritte durchgeführt: Die Daten wurden aus einem Textfile geladen, anschließend liefen mehrere Profiling Processoren und eine Aufbereitung der Daten (Profiling und Audit). Fachlich wurde eine Deduplikation durchgeführt. Danach wurden die Daten bereinigt und aufbereitet (Match and Merge). Zusätzlich wurde die Adreßoption auf die Kundendaten angewandt. Die Analysen könnte man als durchschnittlich komplex bezeichnen. Die Prozesse liefen auf einem 2 CPU/4 Cores Windows 64 Bit Server mit 16 GB Hauptspeicher. Installiert war EDQ 9.0.5. Es wurden keine Tuning Maßnahmen durchgeführt. Ohne Adress Verifikation lief der Benchmark in 6 hr und 33 Minuten.

Benchmark Test	Time	Rate
Data Capture (aus Flatfiles mit je 1 GB Größe)	9 Min.	35,563 DS/Sek.
Profiling	1 Std., 30 Min.	3,536 DS/Sek.
Audit	7 Min.	45,977 DS/Sek.
Cleanse	20 Min.	16,474 DS/Sek.
Address Verification	5 Std., 15 Min.	1,059 DS/Sek.
Match	4 Std., 27 Min.	1,176 DS/Sek.

Abb. 12: Benchmark Ergebnisse mit EDQ (Quelle: Oracle)

Kontaktadresse:
Dr. Holger Dresing
Oracle Deutschland B.V. & Co. KG
Data Integration Solutions DIS - EMEA
Thurnithistr. 2
D-30519 Hannover

Telefon/Fax: +49 511-95787 118
E-Mail holger.dresing@oracle.com
Internet: www.oracle.com