

# Extraktion, Transformation, Laden (ETL)

---

- ETL-Prozeß
- Integrationsschritte
- Integrationsprobleme
  - ◆ Konflikte und deren Klassifikation
  - ◆ Behebung von Konflikten
- Data Cleaning

# ETL-Prozeß

---

- Häufig aufwendigster Teil des Data Warehousing
  - ◆ Vielzahl von Quellen
  - ◆ Heterogenität
  - ◆ Datenvolumen
  - ◆ Komplexität der Transformation
    - Schema- und Instanzintegration
    - Datenbereinigung
  - ◆ Kaum durchgängige Methoden- und Systemunterstützung, jedoch Vielzahl von Werkzeugen vorhanden

## ETL-Prozeß

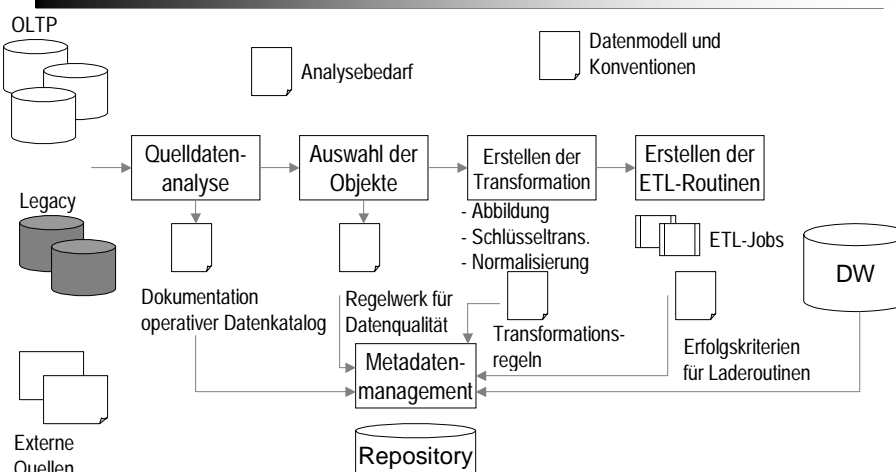
- **Extraktion:** Selektion eines Ausschnitts der Daten aus den Quellen und Bereitstellung für Transformation
- **Transformation:** Anpassung der Daten an vorgegebene Schema- und Qualitätsanforderungen
- **Laden:** physisches Einbringen der Daten aus dem Arbeitsbereich (staging area) in das Data Warehouse (einschl. eventuell notwendiger Aggregationen)

Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-3

## Definitionsphase des ETL-Prozesses



Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-4

## Integrationssschritte

---

1. Vorintegration
  - Analyse der Schemata
  - Auswahl der Integrationsstrategie
  - Vergabe von Präferenzen für einzelne Schemata
  - ggf. semantische Anreicherung
2. Vergleich der Schemata
  - Entdeckung von Korrespondenzen zwischen Schemata
  - Finden möglicher Konflikte

## Integrationssschritte (II)

---

3. Anpassung der Schemata
  - Modifikation der lokalen Schemata für eine anschließende Zusammenführung
  - Beseitigung der Konflikte
4. Zusammenführung und Restrukturierung
  - Bestimmung eines globalen (evtl. partiellen) Schemas
  - Überprüfung der Qualität des Schemas

## Vorgehensweisen

---

- Bottom-Up:
  - ◆ Ausgangspunkt: existierende Schemata
  - ◆ Konstruktion eines davon abgeleiteten, integrierten Schemas, das möglichst den vollständigen Informationsgehalt aller Quellen enthält
- Top-Down:
  - ◆ Ausgangspunkt: vorgegebenes Zielschema (z.B. durch Standards)
  - ◆ Finden der Korrespondenzen zu lokalen Schemata und Definition der Abbildungen

## Integrationsprobleme in Data Warehouse

---

- Schwerpunkt:
  - ◆ Probleme der Datenintegration
- Ausgangspunkt:
  - ◆ Daten liegen in den operativen Informationssystemen
  - ◆ unterschiedliche Systeme

→ *Heterogenität*

## Überwindung der Heterogenität

---

- Zusammenführung der Daten im Data Warehouse erfordert
  - ◆ Schemaintegration
  - ◆ Datenintegration
    - Identifizierung übereinstimmender Strukturen, die semantisch vergleichbare Informationen repräsentieren
    - Bestimmen eines geeigneten Schemas für das Data Warehouse

## Anforderungen an Integration

---

- alle relevanten Daten aus den operativen Systeme müssen im Data Warehouse aufgenommen werden können
- Überführung unterschiedliche Strukturierungen / Darstellungen semantisch gleicher oder zusammengehöriger Daten aus den Quellsystemen in eine gemeinsame Repräsentation
- Identifizierungen gleicher Informationen, die aus mehreren Systemen stammen
  - Beseitigung ungewünschter Redundanz, die Analyseergebnisse verfälschen kann

## Aspekte der Heterogenität

---

- verschiedene Datenmodelle
  - ◆ bedingt durch autonome Entscheidung über Anschaffung von Systemen in den Unternehmensbereichen
  - ◆ verschiedene und verschieden mächtige Modellierungsstrukturen, d.h. Anwendungssemantik in unterschiedlichem Ausmaß erfaßbar
  - ◆ Abbildung zwischen Datenmodellen nicht eindeutig

## Aspekte der Heterogenität (II)

---

- unterschiedliche Modellierungen für gleiche Sachverhalte der Realwelt
  - ◆ bedingt durch Entwurfsautonomie
  - ◆ selbst im gleichen Datenmodell verschiedene Modellierungen möglich, z.B. durch unterschiedliche Modellierungsperspektiven der DB-Designer

## Aspekte der Heterogenität (III)

---

- unterschiedliche Repräsentation der Daten
  - ◆ Unterschiedliche Datentypen möglich
  - ◆ unterschiedliche Umfang der unterstützten Datentypen
  - ◆ unterschiedliche interne Darstellung der Daten
  - ◆ auch unterschiedliche „Werte“ eines Datentyps zur Repräsentation derselben Information

## Kriterien für Integrationsmethoden

---

- generelle Anforderungen an Integrationsverfahren [Batini et al., 1986]
  - ◆ Vollständigkeit
  - ◆ Korrektheit
  - ◆ Minimalität
  - ◆ Verständlichkeit

*Basis für den Vergleich verschiedener Verfahren*

## Kriterien für Integrationsmethoden (II)

---

- Vollständigkeit (engl. *completeness*)
  - ◆ integriertes Schema muß alle Konzepte beinhalten, die in irgendeinem lokalen Schema enthalten sind
  - ◆ es darf keine in einem lokalen Schema enthaltene Information verloren gehen
- Minimalität (engl. *minimality*)
  - ◆ Real-Welt-Konzepte, die in mehreren lokalen Schemata modelliert sind, dürfen nur einmal im integrierten Schema repräsentiert sein
- Verständlichkeit (engl. *understandability*)
  - ◆ integriertes Schema sollte leicht verständlich sein

## Kriterien für Integrationsmethoden (III)

---

- Korrektheit (engl. *correctness*)
  - ◆ alle in dem integrierten Schema enthaltenen Informationen müssen in mindestens einem lokalen Schema semantisch äquivalent vorhanden sein
  - ◆ Inter-Schema-Beziehungen, die während der Integration neu hinzugekommen sind, dürfen nicht im Widerspruch zu Informationen aus den lokalen Schemata stehen, d.h. nur konsistente Ergänzungen der bestehenden Schemata sind erlaubt

## Klassifikation von Integrationskonflikten

---

- *hier*: eine oft verwendete Klassifikation mit vier Klassen von Konflikten [Spaccapietra et al. 1992]
  - ◆ Semantische Konflikte
  - ◆ Beschreibungskonflikte
  - ◆ Heterogenitätskonflikte
  - ◆ Strukturelle Konflikte
  
- in der Regel kombiniertes Auftreten dieser Konfliktarten
- *zusätzlich* - für Data Warehouses besonders wichtig:
  - ◆ Datenkonflikte

Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-17

## Semantische Konflikte

---

- semantisch überlappende Weltausschnitte mit einander entsprechenden Klassen
- *Problem*:
  - ◆ oft nicht genau die gleiche Menge von Real-Welt-Objekten repräsentiert
  - ◆ Unterscheidung zwischen semantisch
    - äquivalenten,
    - sich einschließenden,
    - überlappenden und
    - disjunktenKlassenextensionen notwendig
- Inter-Schema-Korrespondenzen:  $A=B$ ,  $A\subseteq B$ ,  $A\cap B$ ,  $A\neq B$

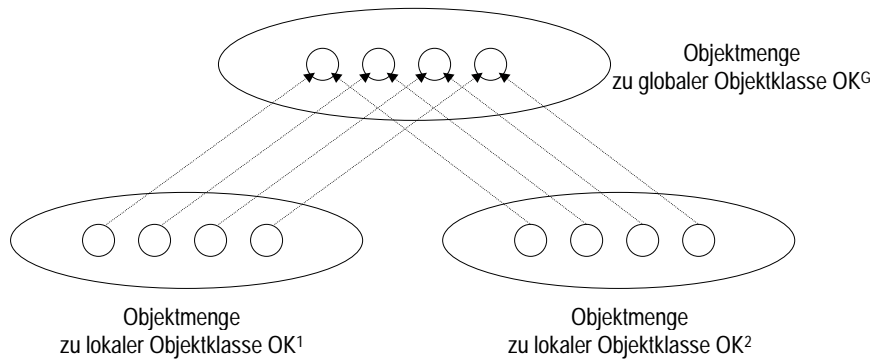
Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-18

## Semantische Konflikte (II)

a) semantisch äquivalente Klassenextensionen:



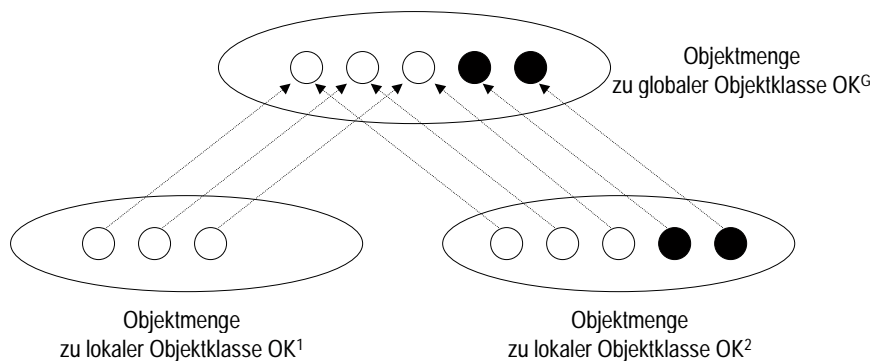
Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-19

## Semantische Konflikte (III)

b) semantische Inklusion zwischen Klassenextensionen:



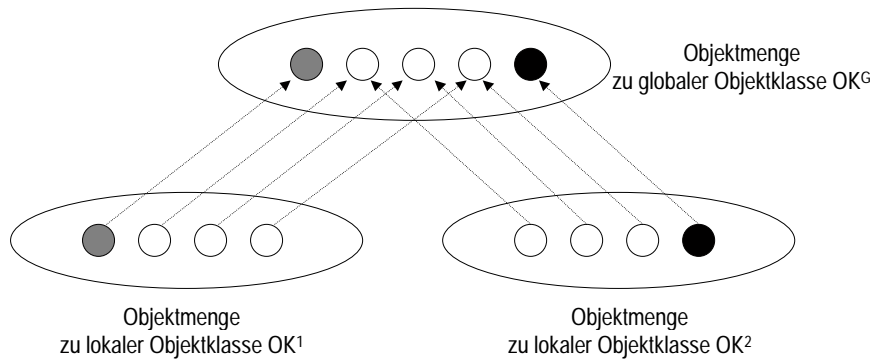
Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-20

## Semantische Konflikte (IV)

- c) semantisch überlappende Klassenextensionen:



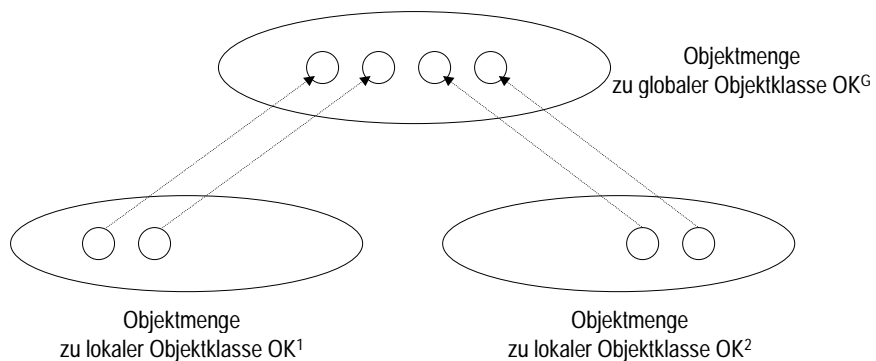
Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-21

## Semantische Konflikte (V)

- d) semantisch disjunkte Klassenextensionen:



Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-22

## Beschreibungskonflikte

---

- unterschiedliche Eigenschaften/Attribute derselben Objekte in den lokalen Schemata
- homonyme und synonyme Bezeichnungen
- Datentypkonflikte / Wertebereichskonflikte: unterschiedliche Datentypen / Wertebereiche für die gleiche Eigenschaft
- Skalierungskonflikte: Verwendung unterschiedlicher, aber ineinander umrechenbarer Maßeinheiten
- Konflikte durch zugehörige Integritätsbedingungen oder Manipulationsoperationen

## Beschreibungskonflikte (II)

---

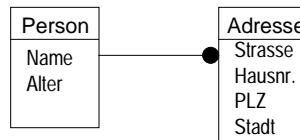
- **Beispiel:**  
Eigenschaften von Waren in verschiedenen Datenbanken

DB1: <i>attributes</i>	Preis	(USD)
	Quantität	(integer)
DB2: <i>attributes</i>	Wert	(PTA)
	Anzahl	(float)

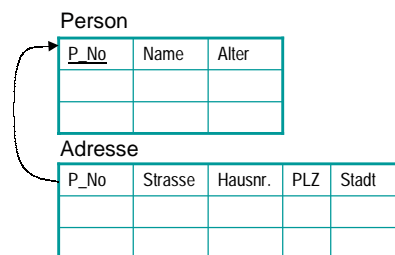
## Heterogenitätskonflikte

- unterschiedliche Datenmodelle der zu integrierenden Schemata
  - ◆ unterschiedliche Modellierungskonstrukte und Ausdruckskraft
  - ◆ impliziert oft auch strukturelle Konflikte
- Auflösung durch Transformation in ein gemeinsames globales Datenmodell

Objektorientiert:



relational:



Kai-Uwe Sattler  
Stefan Conrad

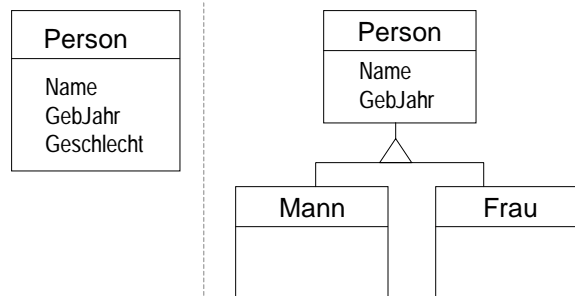
Vorlesung Data-Warehouse-Technologien

4-25

## Strukturelle Konflikte

- selbst bei Verwendung desselben Datenmodells oft unterschiedliche Modellierung eines Sachverhaltes insbesondere bei *semantisch reichen* Datenmodellen (mit vielen Modellierungskonstrukten)

Beispiel:



→ Festlegen der globalen Darstellung & Angabe der Abbildung(en)

Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-26

## **Detaillierte Klassifikation für relationale Datenbanken [Kim et al. 1995]**

### **I. Schemakonflikte**

- A. Tabellen-Tabellen-Konflikte
  - 1. eine Tabelle vs. eine Tabelle
    - a. Tabellennamenkonflikte
      - 1) verschiedene Namen für gleiche Tabellen
      - 2) gleiche Namen für verschiedene Tabellen
    - b. Tabellenstrukturkonflikte
      - 1) fehlende Attribute
      - 2) fehlende, aber implizite Attribute
    - c. Integritätsbedingungskonflikte
  - 2. viele Tabellen vs. viele Tabellen

Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-27

## **Detaillierte Klassifikation für relationale Datenbanken (II)**

### **Fortsetzung: Schemakonflikte**

- B. Attribut-Attribut-Konflikte
  - 1. ein Attribut vs. ein Attribut
    - a. Attributnamenkonflikte
      - 1) verschiedene Namen für gleiche Attribute
      - 2) gleiche Namen für verschiedene Attribute
    - b. Default-Wert-Konflikte
    - c. Integritätsbedingungskonflikte
      - 1) Datentypkonflikte
      - 2) Bedingungskonflikte
  - 2. viele Attribute vs. viele Attribute
- C. Tabelle-Attribut-Konflikte

Kai-Uwe Sattler  
Stefan Conrad

Vorlesung Data-Warehouse-Technologien

4-28

# Detaillierte Klassifikation für relationale Datenbanken (III)

## II. Datenkonflikte

- A. falsche Daten
  - 1. nicht korrekte Einträge
  - 2. veraltete Daten
- B. unterschiedliche Repräsentationen
  - 1. verschiedene Ausdrücke
  - 2. verschiedene Einheiten
  - 3. Unterschiedliche Genauigkeit

## Beispiel: Datenkonflikte

Personen	Name	Geb.Jahr	Beruf
	Peter Meier	1962	Dipl.-Inform.
	Ingo Schmitt	1928	Dichter
	...	...	...

Personen	Name	Geb.Jahr	Beruf
	Meier, Peter	62	Informatiker
	Schmitt, Ingo	28	Lyriker
	...	...	...

## Datenkonflikte: Beispiele

---

- inkorrekte Einträge:
  - ◆ Tippfehler bei Eingabe von Werten
  - ◆ falsche Einträge aufgrund von Programmierfehlern in einzelnen Anwendungsprogrammen

→ *i.d.R. nicht automatisch behebbar !!!*
- veraltete Einträge:
  - ◆ durch unterschiedliche Aktualisierungszeitpunkte
    - z.B. weil Aktualität einer Quelle ausreicht
    - z.B. weil Aktualisierung durch eine Quelle verzögert wird (Ausführungsautonomie)
  - ◆ „vergessene“ Aktualisierungen in einzelnen Quellen
    - z.B. durch Fehler in Anwendungsprogrammen

## Datenkonflikte: Beispiele (II)

---

- verschiedene Ausdrücke:
  - ◆ verschiedene Datentypen:
    - z.B. Aufzählungstyp oder numerischer Typ für Notenwerte
      - „sehr gut“, ... , „nicht ausreichend“
      - 1, ... , 5
  - ◆ gleicher Datentyp:
    - z.B. bei Strings für Adressen

"Breitestraße"	"Breitestrasse"	"Breitestr."
"Breite Straße"	"Breite Strasse"	"Breite Str."
"Breite-Straße"	"Breite-Strasse"	"Breite-Str."

## Behebung von Datenkonflikten

---

- explizite Werteabbildung
  - ◆ z.B. bei unterschiedlichen Aufzählungstypen
  - ◆ exakt bei gleicher Kardinalität der Wertebereiche
- Einführung von Ähnlichkeitsmaßen
  - ◆ bei Tippfehlern
  - ◆ bei leicht unterschiedlichen Schreibweisen erkennt ggf. zu viel oder zu wenig als *ähnlich*
- Bevorzugung der Werte aus einer lokalen Quelle
  - ◆ bei unterschiedlicher Aktualität der Daten
  - ◆ bei unterschiedlicher Vertrauenswürdigkeit

## Behebung von Datenkonflikten (II)

---

- Verwendung von Hintergrundwissen
  - ◆ über Konventionen z.B. bzgl. Schreibweisen
  - ◆ über Homonyme und Synonyme (Wörterbücher, Thesauri)
  - ◆ über Zusammenhänge von Begriffen und Konzepten im Anwendungsgebiet (Ontologien)

→ ***Einsatz wissensbasierter Verfahren***

## Data Cleaning

---

- Korrektur inkorrektur, inkonsistenter oder unvollständiger Daten
- Auch: Data Cleansing, Data Scrubbing
- Techniken:
  - ◆ Konvertierungs- und Normalisierungsfunktionen
  - ◆ Domänenspezifische Bereinigung
  - ◆ Domänenunabhängige Bereinigung
  - ◆ Regelbasierte Bereinigung

## Konvertierungs- und Normalisierungsfunktionen

---

- Transformation und Standardisierung heterogener Datenformate
  - ◆ Konvertierung unterschiedlicher Formate (z.B. Textdateien in DB-Tabellen über Oracle SQL\*Loader)
  - ◆ Normalisierung: Abbildung von Datenfeldern in ein gemeinsames Format
    - Zeichenketten in Großschreibung
    - Datumsformat: dd/mm/yyyy
    - Währungen

## Domänenspezifische Bereinigung

---

- Nutzung von Domänenwissen zur Bereinigung einzelner Felder
- Einsatz spezielle Werkzeuge möglich (häufig auf Basis von Wörterbüchern)
- Beispiele:
  - ◆ Produktbezeichnungen im Pharmabereich
  - ◆ Adressen über Adreßdatenbanken (Postleitzahlen, Telefonvorwahl)
  - ◆ Synonyme und Abkürzungen („Str.“ für „Straße“)

## Domänenunabhängige Bereinigung

---

- Verschmelzen von Objekten aus unterschiedlichen Quellen, die ein Real-Welt-Objekt repräsentieren
  - ◆ Schlüsselvergleich („exact matching“)
  - ◆ Attributvergleich bzw. –ähnlichkeit („fuzzy matching“)
  - ◆ Beispiel: Felder mit komplexen Zeichenketten
    - Übereinstimmung := Anzahl der übereinstimmenden atomaren Strings / durchschnittl. Anzahl von atomaren Strings

## Regelbasierte Bereinigung

---

- Nutzung von Regeln zur Erkennung von Übereinstimmung zwischen Objekten basierend auf der Kombination mehrerer Felder
- Formen:
  - ◆ *benutzer-spezifizierte Regeln:*
    - Kodieren der Regeln durch Benutzer
    - in C oder PL/SQL bzw. im Rahmen spezieller Werkzeuge
  - ◆ *Automatisch abgeleitete Regeln:*
    - Ableiten von Regeln aus Analyse der Daten z.B. durch Klassifikationsverfahren (Entscheidungsbäume) oder Assoziationsregeln
    - Beispiel: WizRule

## Regelbasierte Bereinigung: Beispiele

---

- Mathematische Regel:  
A = B \* C  
WHERE  
A = Total, B = Quantity, C = Unit Price  
Rule's accuracy level: 0.99  
rule exists in 1890 records
  - ◆ Regelgenauigkeit:= Anzahl der Fälle für die Formel gültig ist / Anzahl der relevanten Fälle

## Regelbasierte Bereinigung: Beispiele

---

- IF-THEN-Regel:  
IF Customer IS "Summit" AND Item IS Computer type X  
THEN Salesperson = "Dan Wilson"  
Rule's probability: 0.98  
rule exists in 102 records  
error probability < 0.1
- Regelwahrscheinlichkeit := Anzahl der Datensätze für die Bedingung und Ergebnis gilt / Anzahl der Datensätze mit der Bedingung