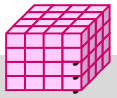
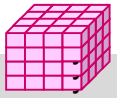
4. ETL: Datenvorverarbeitung und -integration

- ETL-Überblick
- Schemaintegration
  - Bottom-Up- vs. Top-Down-Integration
  - Semantische Heterogenität
- Schema Matching
  - Verfahren
  - Prototypen / Tools
- Data Cleaning
  - Probleme
  - Teilaufgaben
- Entity Matching (record linkage)
  - Blocking, Matching, Clustering
  - Systeme/Prototypen: MS SQL-Server, Dedoop, Famer



ETL-Prozess

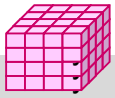
- Data Warehousing und ETL: materialisierter Ansatz zur Datenintegration
  - Erzeugung einer aggregierten, materialisierten Datenquelle für Online-Analysen
  - komplexer, aufwändiger Integrationsprozess
  - Offline-Durchführung erlaubt höhere Datenqualität gegenüber virtueller Datenintegration (Datentransformation während Query-Verarbeitung)
- *Extraktion*: Selektion eines Ausschnitts der Daten aus Quellen
  - ausgeführt an den entsprechenden Quellen
- *Transformation*: Aufbereitung und Anpassung der Daten an vorgegebene Schema- und Qualitätsanforderungen
  - ausgeführt im temporären Arbeitsbereich (Data Staging Area)
- *Laden*: physisches Einbringen der Daten aus Arbeitsbereich in das Data Warehouse, einschließlich evtl. notwendiger Aggregationen



ETL-Prozess (2)

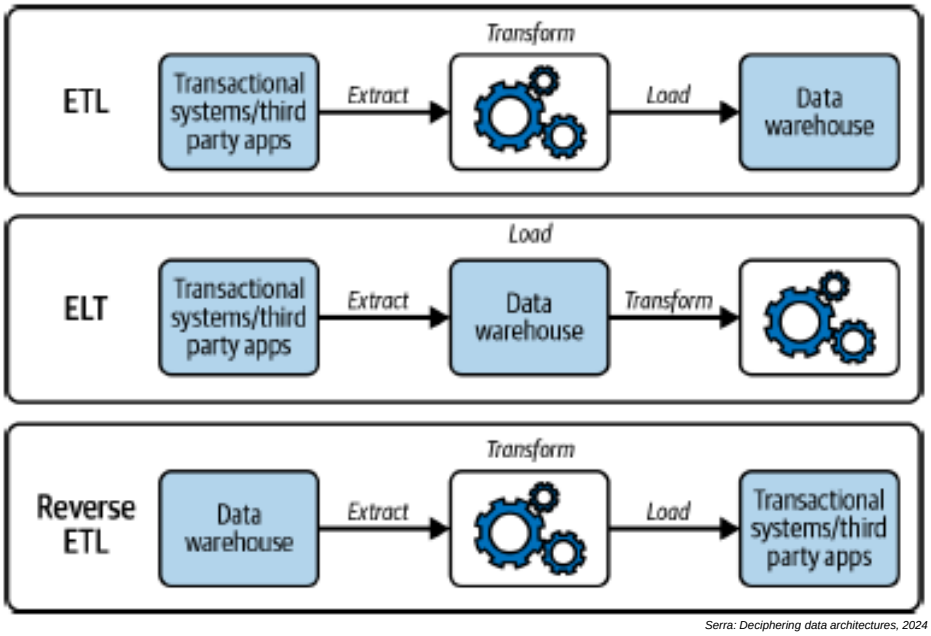
- aufwändigster Teil des Data Warehousing
  - Vielzahl von operativen Quellen
  - Heterogenität der Datenquellen (DBMS, Schemata, Daten)
  - Gewährleistung hoher Qualität der Warehouse-Daten

- entscheidende Rolle im Data Warehousing, da großer Einfluss auf
  - Genauigkeit und Richtigkeit der später durchgeführten Analysen
  - die darauf basierenden Entscheidungen: „Garbage In, Garbage Out“

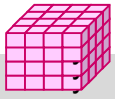


ETL vs ELT vs Reverse ETL

- ELT: schnelleres Laden von Daten, v.a. für Data Lake (Load vor Transform, „Every Load Transforms“)

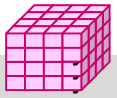


Serra: Deciphering data architectures, 2024

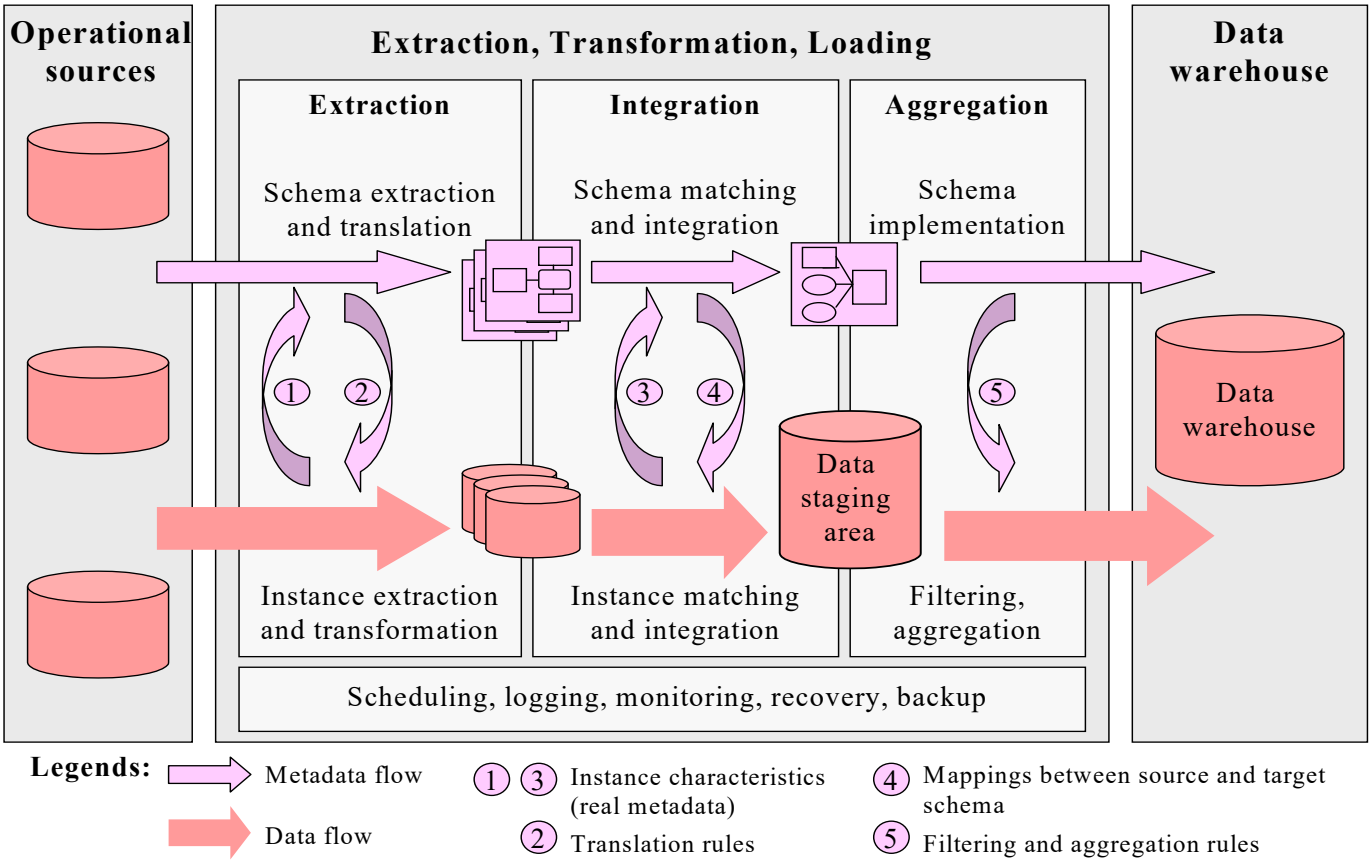


ETL als Integrationsprozess

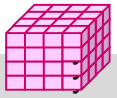
- ETL: Datenintegration auf 2 Ebenen: Schemata und Instanzdaten
- Schemaintegration
  - Abbildung von Quellschemata auf Data-Warehouse-Schema: *Schema Matching*
  - Adressierung der semantischen Heterogenität (Namenskonflikte, strukturelle Konflikte)
  - ggf. zunächst Anpassung der Schemarepräsenationen (z.B. RM)
- Datenintegration / Data Cleaning
  - Transformation heterogener Daten in einheitliche, durch Data Warehouse-Schema vorgeschriebene Repräsentation
  - Entdeckung und Behebung von Datenqualitätsproblemen
  - Entdeckung äquivalenter Objekte/Sätze (Korrespondenzen auf Instanzenebene): *Entity Matching* / Duplikatbehandlung



ETL-Prozess: Ablauf



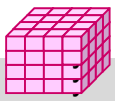
E. Rahm, H. H. Do: *Data Cleaning: Problems and Current Approaches.* IEEE Techn. Bulletin on Data Engineering, 2000



Schemaintegration - Anforderungen

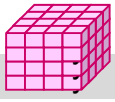
- Minimalität
  - keine Redundanz im integrierten Schema
  - Abbildung mehrerer gleicher/ähnlicher Konzepte in lokalen Schemata auf ein Konzept im integrierten Schema
- Korrektheit
  - Äquivalenz der im integrierten Schema enthaltenen Informationen mit denen in den lokalen Schemata
  - Konsistenz der während der Integration ergänzten Informationen, z.B. Beziehungen zwischen Konzepten im integrierten Schema
- Verständlichkeit

Vollständigkeit (Beibehaltung aller Informationen aus Quellschemas) ?



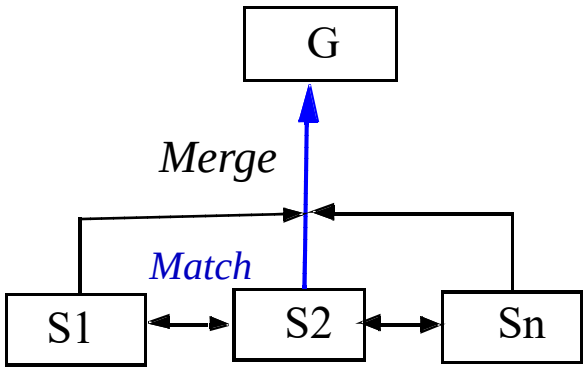
Schemaintegration (2)

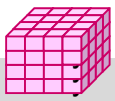
- Probleme der Schemaintegration
  - Heterogenität der Schemarepräsentationen
    - z.B. relational (SQL), JSON, Dateien ...
  - semantische Heterogenität der Schemaelemente (Namenskonflikte, strukturelle Konflikte)
- Alternativen
  - Bottom-Up-Schemaintegration
  - Top-Down-Schemaintegration



Bottom-Up-Integration (*Global as View*)

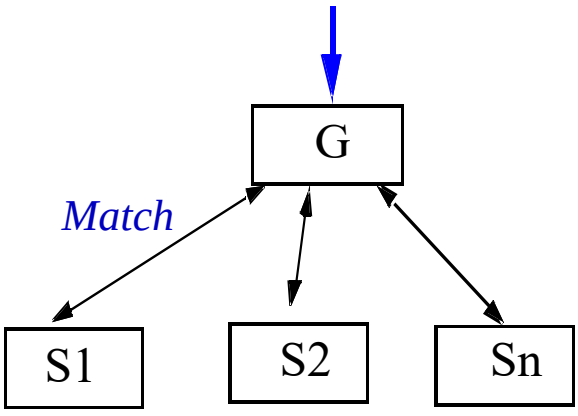
- (vollständiges) Mischen aller Source-Schemata in globales Schema
- setzt Abgleich zwischen Source-Schemas voraus, insbesondere Bestimmung von Korrespondenzen / Konflikten
- globales Schema entspricht gemeinsamer Sicht (View) auf die zugrundeliegenden Quellen
- neue Quelle ändert meist globales Schema

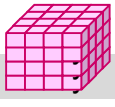




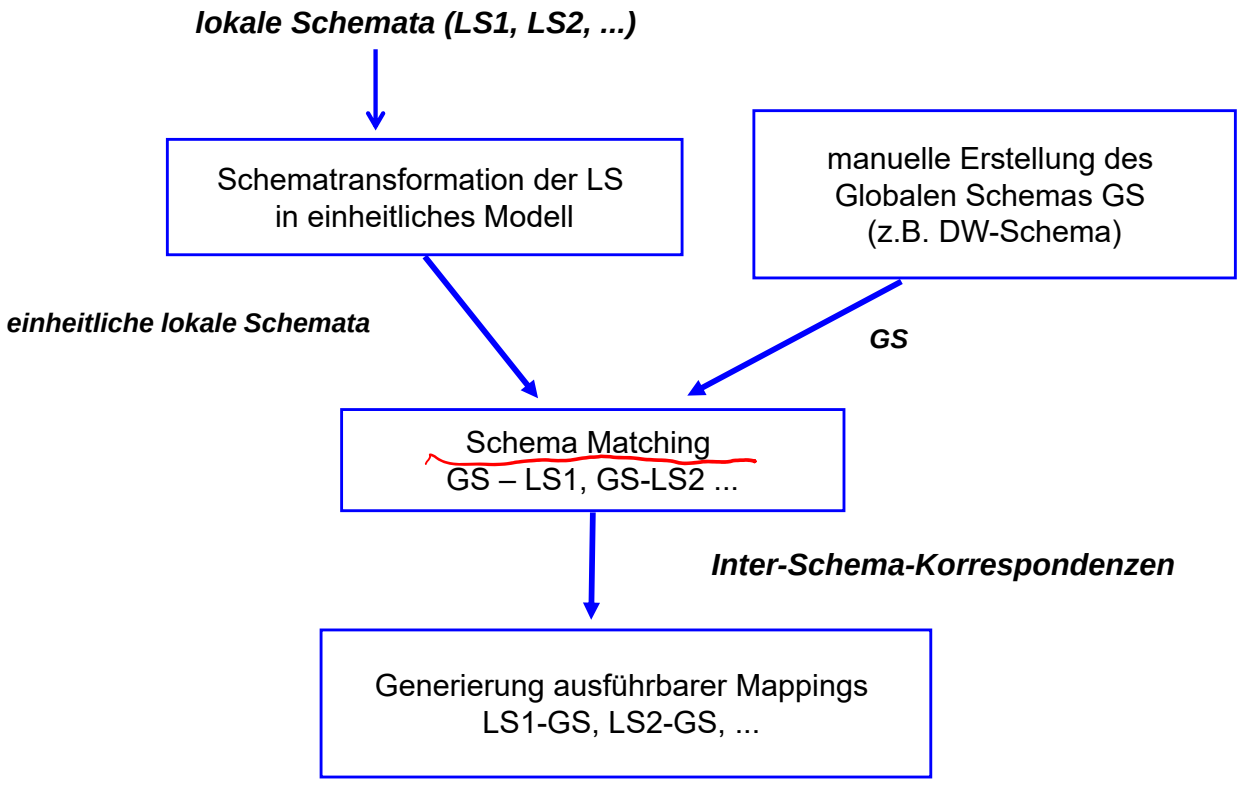
Top-Down-Integration (*Local as View*)

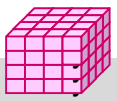
- globales Schema G ist vorgegeben
- jede Source S wird unabhängig von anderen Sources mit globalem Schema abgeglichen, d.h. ein Mapping G - S erstellt (Mapping beschreibt Inhalt der Quelle)
- aufwändige Query-Verarbeitung bei virtueller Integration
- G berücksichtigt i.a. nur Teile der lokalen Schemata





Top-Down-Schemaintegration (2)





Namenskonflikte

Mitarbeiter
Name
Adresse

Angestellte
Name
Anschrift

Firma
Name
Adresse

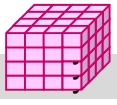
- **Synonyme**: unterschiedliche Namen für das selbe Konzept

  `Mitarbeiter ≈ Angestellte`
  `Mitarbeiter.Adresse ≈ Angestellte.Anschrift`

- **Homonyme**: gleiche Namen für verschiedene Konzepte

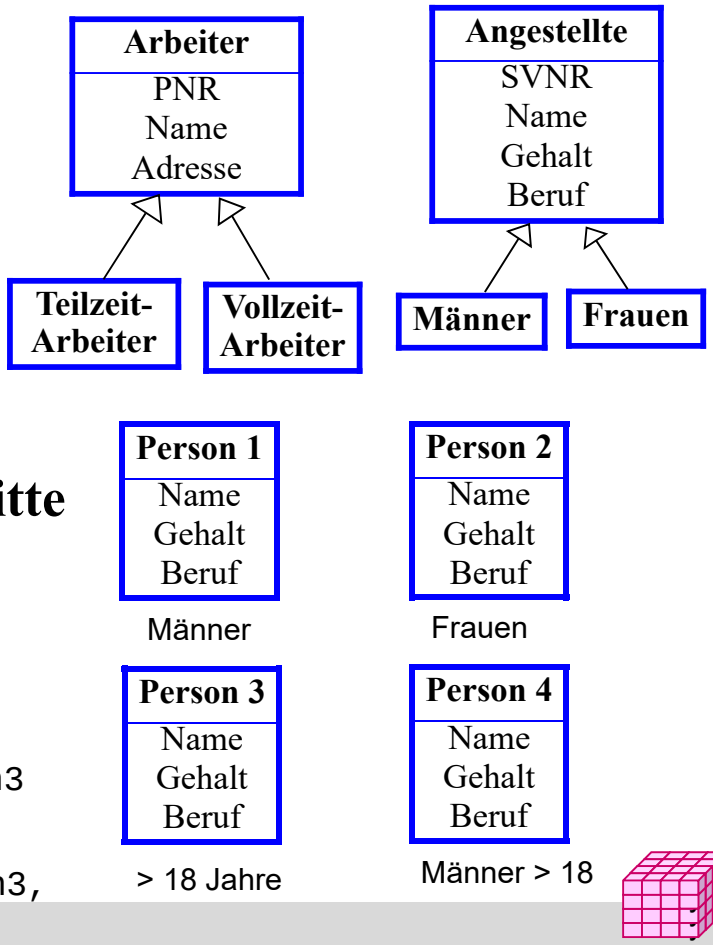
  `(Mitarbeiter.)Name ≠(Firma.)Name`
  `(Mitarbeiter.)Adresse ≠(Firma.)Adresse`

- **Hyponyme/Hyperonyme:** Unter-/Oberbegriffe



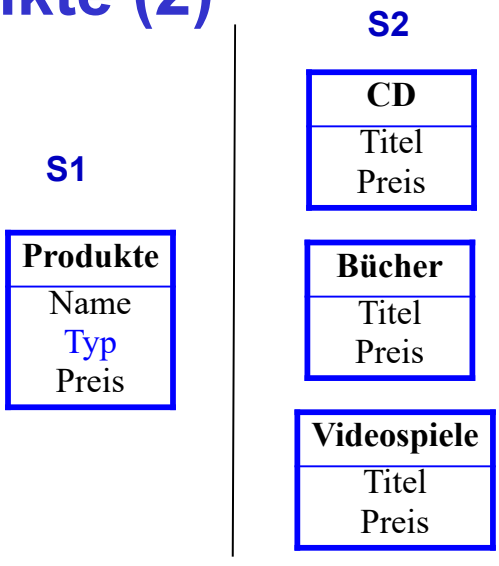
Strukturelle Konflikte

- Entity vs. Entity
  - unterschiedliche Schlüssel
    PNR - SVNR
  - unterschiedliche Attributmengen, fehlende Attribute
  - unterschiedliche Abstraktionsebenen (Generalisierung, Aggregation)
- **unterschiedliche Realitätsauschnitte** (RWS, real world states), Instanzmengen
  disjunkt (disjoint):
  Person1 –Person2
  überlappend (overlaps):
  Person1 –Person3, Person2 –Person3
  enthalten (contains):
  Person4 ⊆Person1, Person4 ⊆Person3,



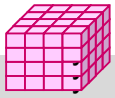
Strukturelle Konflikte (2)

- Attribut vs. Entity-Konflikte
  - Repräsentation von Attributen als eigenständige Entities/Relationen
  - Bsp.: Produkttyp
- Attribut vs. Attribut-Konflikte
  - unterschiedliche Datentypen
    Preis (Float) vs. Preis (String)
  - unterschiedliche Detailgrade
    Name vs. Vorname und Nachname
  - unterschiedliche Einheiten: *$ vs. Euro*
  - unterschiedliche Genauigkeiten: *Tausend Euro vs. Euro*
  - unterschiedliche Integritätsbedingungen, Wertebereiche, Default-Werte ...
    Alter >18 vs. Alter > 21
  - unterschiedliche Zulässigkeit von Nullwerten



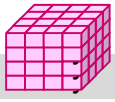
Automatisierungsbedarf

- bisherige Schemaintegrationsansätze weitgehend manuell
  - nutzerdefinierte Korrespondenzen und Konfliktbehandlung
  - Nutzung spezifischen Schema-/Domain-Wissens
  - aufwändig / fehleranfällig vor allem für größere Schemata
  - nicht skalierbar auf viele Schemata
  - hoher Anpassungsaufwand bei Schemaänderungen
- Skalierbarkeit erfordert semi-automatische Lösungen / Tools!
  - vollautomatische Lösungen aufgrund semantischer Heterogenität nicht möglich
  - Namensproblematik (Synonyme, Homonyme)
  - begrenzte Mächtigkeit von Metadaten / Schemasprachen
- (Teil-)Automatisches Schema-Matching
  - v.a. für große Schemata wichtig
  - Nutzer-Feedback notwendig, jedoch im begrenzten Umfang



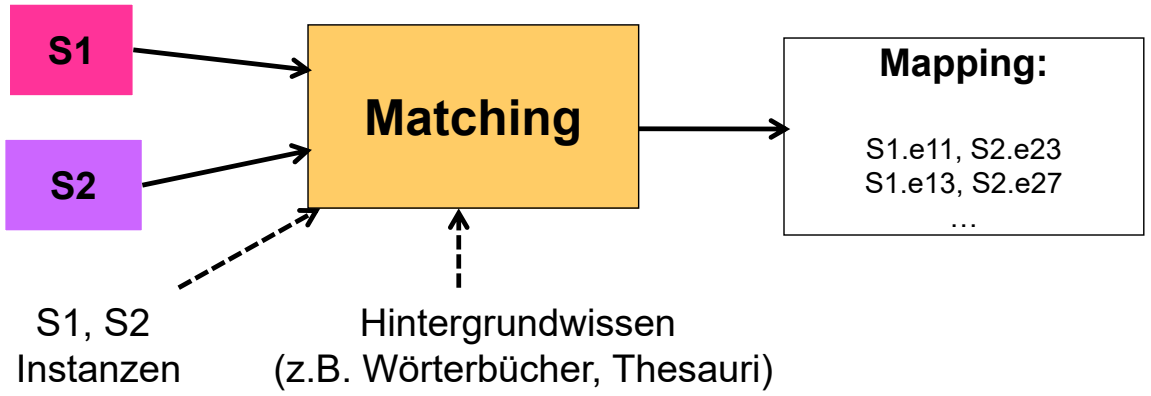
4. ETL: Datenvorverarbeitung und -integration

- ETL-Überblick
- Schemaintegration
  - Bottom-Up- vs. Top-Down-Integration
  - Semantische Heterogenität
- Schema Matching
  - Verfahren
  - Prototypen / Tools
- Data Cleaning
  - Probleme
  - Teilaufgaben
- Entity Matching (record linkage)
  - Blocking, Matching, Clustering
  - Systeme/Prototypen: MS SQL-Server, Dedoop, Famer

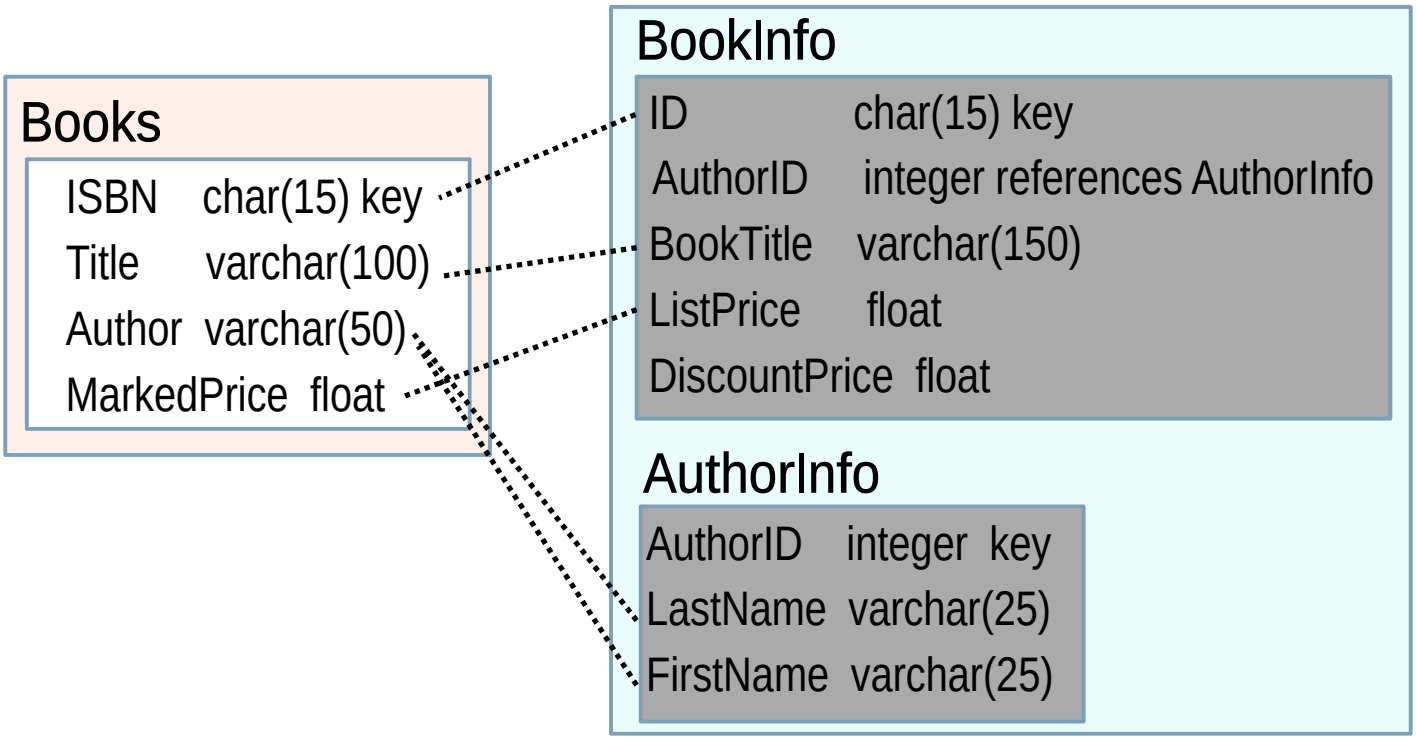


Schema Matching

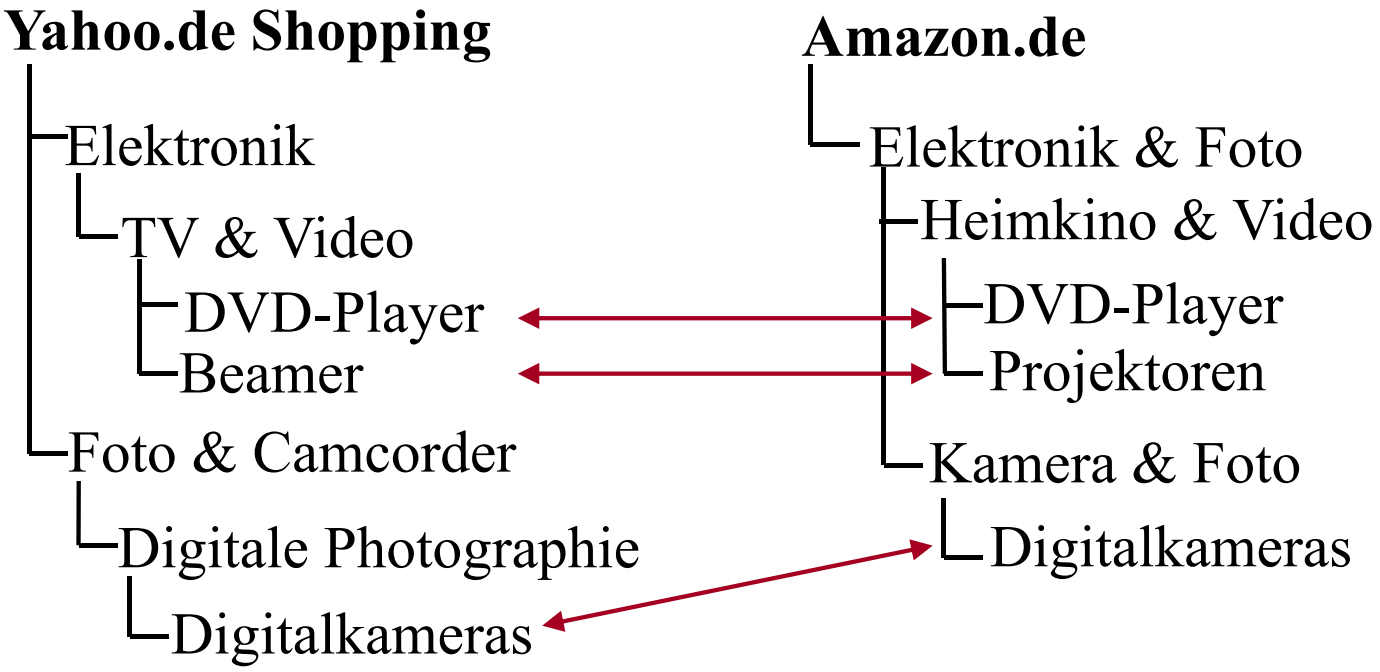
- kritischer Schritt in zahlreichen Applikationen
  - Datenintegration (u.a. Data Warehouses), Datenaustauch (XML Message Mapping), Wissensverarbeitung (Ontologie-Matching)
- Finden semantischer Korrespondenzen zwischen 2 Schemas
  - Input: Schemas und evtl. Instanzbeispiele und Hintergrundwissen
  - Output: paarweise Korrespondenzen zwischen Schemaelementen (z.B. `S1.e1 – S2.e2`), ggf. ergänzt um numerische Ähnlichkeit (zwischen 0 und 1) und Beziehungstyp: equal, is-a, part-of, related, …



Match-Beispiel 1: relationale Schemas



Match-Beispiel 2: Produktkataloge



Match-Beispiel 3: n Produkt-Schemata (Kameras)

Ilgs.net

<page title>: "Pentax K-5 II + 18-55 mm WR – Digital...",
battery type: "D-LI90 Lithium-ion battery",
box contents: "AV cable USB cable Li-ion battery D-LI90...",
colour of product: "Black",
compatible memory cards: "SD/SDHC/SDXC",
dimensions w x d x h: "131 x 72.5 x 97 mm",
interface: "Mini-HDMI and AV outputs compatible with...",
iso sensitivity: "80 to 51 200",
lcd screen size: "3.0",
lens type: "18-55MM WR Lens",
megapixel: "16",
sensor type: "CMOS",
type: "Single Lens Kit",
warranty: "1 Year",
weatherproof: "Yes",
weight: "760 g",
white balance: "Auto Daylight Shade Cloudy Fluorescent..."

Eglobalcentral.co.uk

<page title>: "Nikon D3300 Kit with 18-55mm VR II Len...",
aspect ratio: "3:2",
battery: "EN-EL14a lithium-ion battery and charger",
continuous shooting: "Yes (5.fps)",
dimensions w x h x d: "124 x 98 x 76 mm (4.88 x 3.86 x...",
effective pixels: "24 megapixels",
focal length fmm: "1.5\u00d7",
gps: "None",
hdmi: "Yes (mini-HDMI)",
image processor: "Expeed 4",
image stabilisation: "No",
iso sensitivity: "Auto, 100, 200, 400, 800, 1600, 3200,...",
lcd type: "Fixed",
lens mount: "Nikon F mount",
max resolution: "6000 x 4000",
microphone: "Mono",
sensor size: "APS-C (23.5 x 15.6 mm)",
sensor type: "CMOS",
shutter speed: "30 sec - 1/4000 sec",
storage type: "SD/SDHC/SDXC",
total pixels: "25 megapixels",
usb: "USB 2.0",
viewfinder: "Optical (pentamirror), 95%",
weight inc batteries: "430 g (0.95 lb / 15.17 oz)",
wireless: "Optional"

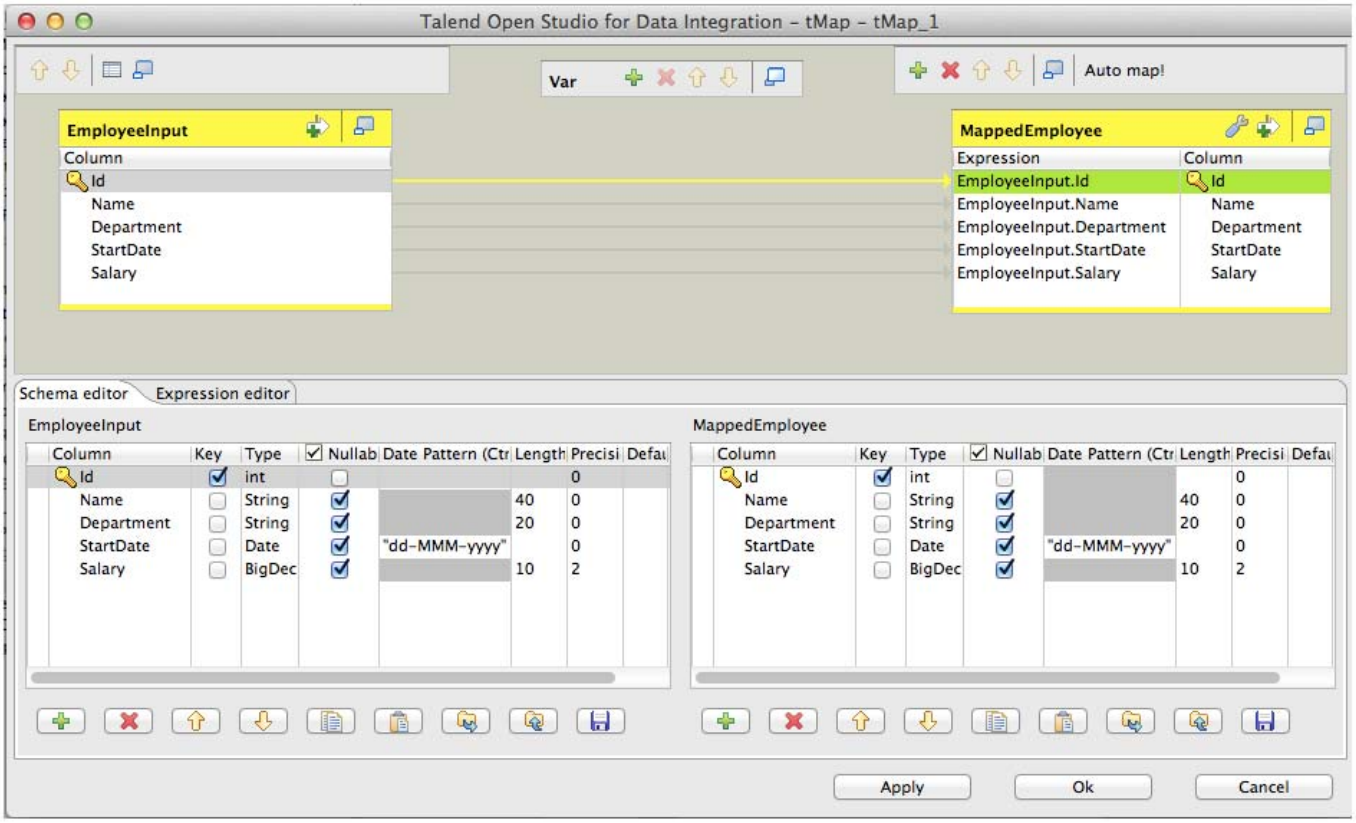
Shopmania.in

<page title>: "Kodak DC220 digital camera prices...",
aperture: "F/4.0-4.8",
autofocus: "Contrast Detect\n/ Live View\n/ Single",
brand: "Kodak",
category: "Digital Cameras",
digital zoom: "2x",
exposure compensation: "-/+ 2 EV range, in 1/2 EV steps",
flash modes: "Auto / Fill-in / Off",
focal length equivalent to 35mm: "29 - 58 mm",
image ratio: "4:3",
image resolutions: "640x480",
light sensitivity iso: "140",
live view: "Yes",
lowest recommended price: "Rs.3,894.19 - http://www...",
max shutter speed: "1/362",
min shutter speed: "1/2",
optical zoom: "2x",
product name: "Kodak DC220",
product rating": "0 out of 5",
resolution: "0.9 MP",
screen size: "2\"",
sensor type: "CCD",
user reviews: "Write a review",
viewfinder type: "Optical",
weight: "550 g"

Mypriceindia.com

<page title>: "Fujifilm Finepix Z70 Price In India...",
aperture range: "F4.0 (W) - F4.8 (T)",
audio formats: "WAV",
auto focus: "Yes, Contrast Detect, Tracking, Single, Live...",
camera resolution: "12 MP",
digital zoom: "6.3x",
focal length: "6.4 - 32 mm (35 mm Equivalent to 36 – 180...",
image format: "JPEG (Fine & Normal JPEG Quality)",
image stablizer: "Yes",
iso rating: "100 - 1600",
lens type: "Fujinon 5x optical zoom lens",
manual focus: "No",
maximum shutter speed: "1/2000 sec",
minimum shutter speed: "4 Sec",
optical zoom: "5x",
other focus features: "Normal Focus Range (80 cm)...",
self timer: "Yes, 2 or 10 sec, Couple, Group",
video format: "AVI",
white balancing: "White Balance Presets (6), Custom..."

Mapping Editor in Talend



Automatische Match-Ansätze

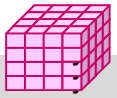
- linguistische Ansätze
  - syntaktische Ähnlichkeit von Attribut/Konzeptnamen: Stringähnlichkeit
  - semantische Ähnlichkeit über *Wörterbücher* (unter Berücksichtigung von Synonymen sowie Hypo/Hypernymen) oder *Wort-Embeddings* aus vortrainierten Vokabularen
  - Ähnlichkeit von Beschreibungen
- strukturelle Ähnlichkeit (bei „tieferen“ Schemata bzw. Ontologien)
  - ähnliche Vorgänger bzw. Nachfolger
- ähnliche Instanzdaten (z.B. Attributwerte)
- Ähnlichkeit zu bereits berechneten Korrespondenzen
- kombinierte Ansätze

CHAT-GPT-LÖSUNG

welche Korrespondenzen bestehen zwischen folgenden Schemata? Schema 1 - Tabelle Books(ISBN char(15) key, Title varchar(100), Author varchar(50),MarkedPrice float) und Scham 2 mit zwei Tabellen BookInfo (ID char(15) key, AuthorID integer references AuthorInfo , BookTitle varchar(150), ListPrice float, DiscountPrice float)) und AuthorInfo (AuthorID integer key, LastName varchar(25), FirstName varchar(25))

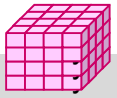
Zusammenfassung der Korrespondenzen

Schema 1 (Books)	Schema 2 (BookInfo/AuthorInfo)	Typ/Korrespondenz
ISBN	ID	Eindeutiger Identifikator
Title	BookTitle	Buch-Titel
MarkedPrice	ListPrice	Standard-/Listenpreis
Author	AuthorID ↔ AuthorInfo (Vor-/Nachname)	Autor (Schema 2 verwendet Normalisierung)



Linguistisches Matching

- einfachster Ansatz: Namensgleichheit
- syntaktische Ansätze: approximate String-Ähnlichkeit von Namen
  - q-grams, Edit distance, TF/IDF ...
- semantische Ähnlichkeit durch Nutzung terminologischer Beziehungen
  - Synonyme: *KFZ ~ Auto*
  - Hyponyme/Hyperonyme: *Buch* is-a *Publikation*, *Artikel* is-a *Publikation*
  - Nutzung von Wörterbüchern / Thesauri, z.B. WordNet
- **Vorverarbeitung** zur Behandlung kryptischer Namen, Auflösung von Abkürzungen, etc.
  - Tokenisierung von Namen: *PO_OrderNum* → {*PO*, *Order*, *Num*}
  - Expansion von Akronymen, Kurzformen: *PO* → *Purchase Order*
    Num → *Number*



Ähnlichkeitsmaße für Strings: Edit-Distance

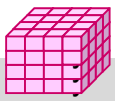
- Gegeben seien zwei Strings a und b
- Editabstand d(a,b) = Anzahl der Edit-Operationen, um a in b zu konvertieren
  - Edit-Operationen: Einfügen, Löschen oder Ersetzen eines Zeichens
- Ähnlichkeit ist normierter Editabstand
  - auch: Levenshtein-Abstand

$$sim_{edit}(a,b) = 1 - \frac{dist(a,b)}{\max(|a|,|b|)}$$

- Beispiel: „Street“ vs. „ShippingStreet“ (Länge 6 bzw. 14)
  - Überführung erfordert Einfügung von 8 Zeichen (S h i p p i n g)

    dist = 8 normierte Distanz: 8/14 = 0,57

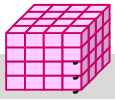
    sim_{edit} = 1 - 8/14 = 0,43



Ähnlichkeitsmaße für Strings: q-gram

- Idee: Überlappung bzgl. Substrings der Länge q
  - Häufig q=3 (Trigramme) oder q=2 (Bigramme)
  - optionals *Padding* der Eingabestrings (q-1 Füllzeichen als Präifx/Suffix)
- verschiedene Änlichkeitsmetriken für Überlappung der q-Gramme (set similarity), v.a. Dice- und Jaccard-Ähnlichkeit
  - Q(a) = Menge der q-Gramme in String a

$$\mathbf{sim_{dice}} = \frac{2\,|Q(a) \cap Q(b)|}{|Q(a)| + |Q(b)|} \qquad \mathbf{sim_{jaccard}} = \frac{|Q(a) \cap Q(b)|}{|Q(a) \cup Q(b)|}$$



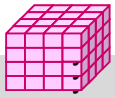
Beispiel q-gram

- „Street“ vs. „ShipingStreet“ (Trigramm-Ähnlichkeiten)

– ohne Padding; `Street: Str, tre, ree, eet (4 3-Gramme)`
`ShippingStreet: Shi,hip,ipp,ppi,pin,ing,ngS,gSt,Str,tre,ree,eet(12)`
sim_{dice} = $sim_{jaccard}$ =

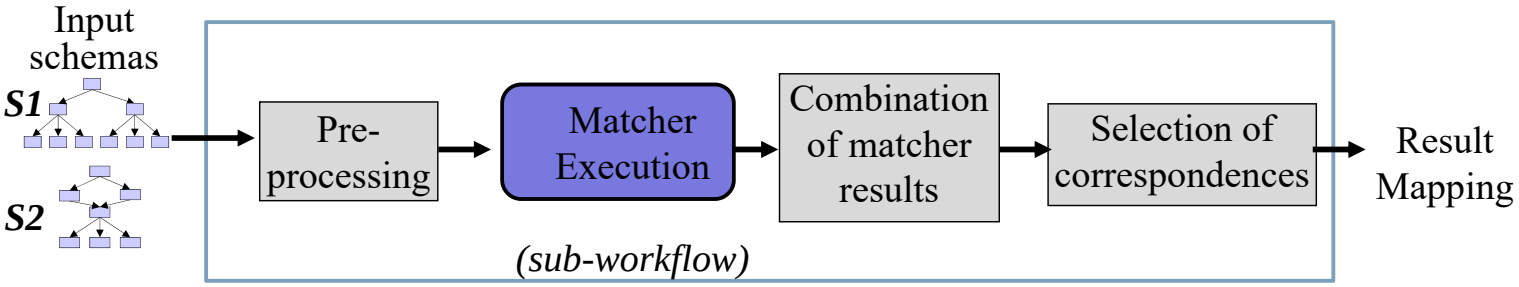
– mit Padding: `Street: __S,_St, Str, tre, ree, eet, et_, t__ (8 3-Gramme),`
`ShippingStreet: 12+4=16 Q-Gramme (6 übereinstimmend mit Street)`

sim_{dice} = $sim_{jaccard}$ =

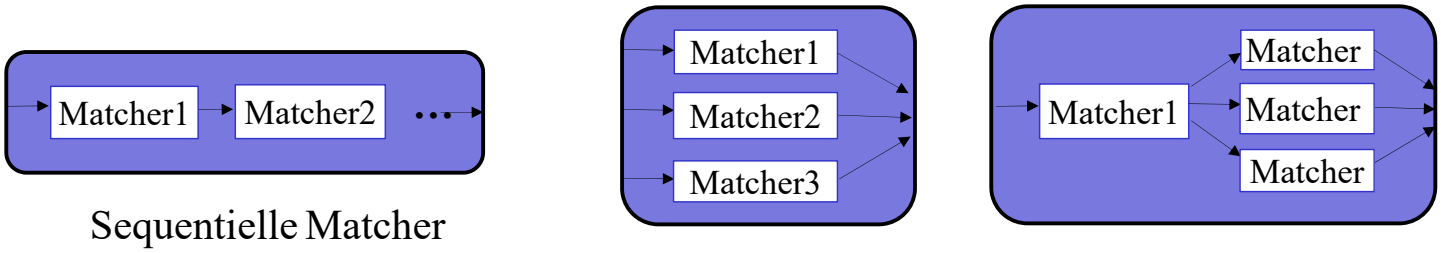


Kombination von Matchern

allgemeiner Match-Workflow (COMA, …)



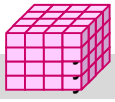
Matcher-Ausführung:



Sequentielle Matcher

Parallele (unabhängige) Matcher

Gemischte Strategie



COMA-Architektur (VLDB 2002)

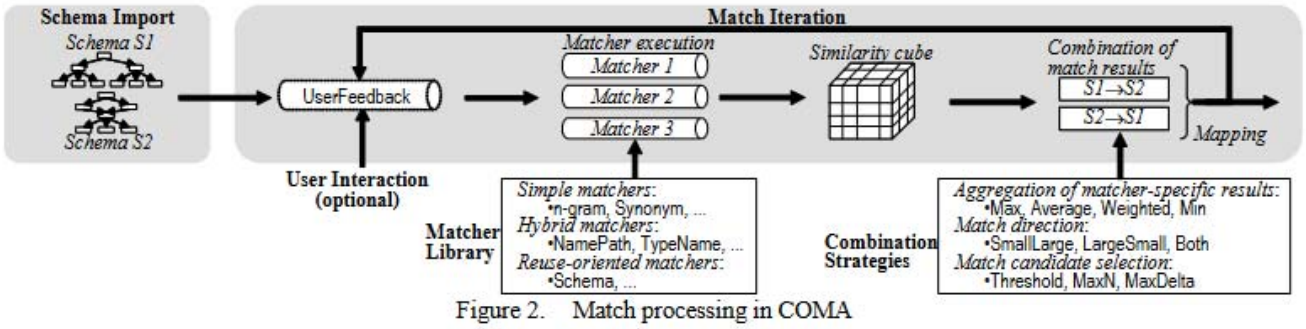
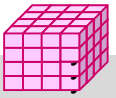


Figure 2. Match processing in COMA

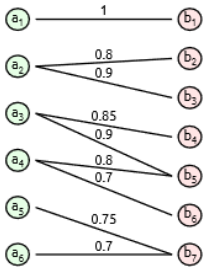
- Auswahl der Match-Kandidaten
  - oft sinnvoll nur Best-Match pro Schemaelement zu betrachten
  - best match (S1.a1), S2.b1, kann anderen best-match in S1 haben, zB S1.a2

  -> Problem der *Stable Marriage*

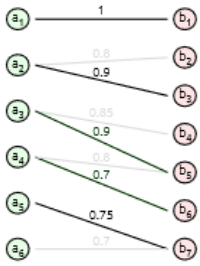


Auswahl von 1:1 Matches

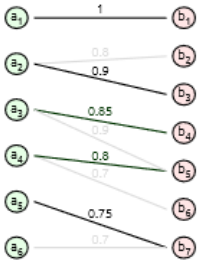
- einfacher Ansatz (Coma) - Symmetric Best Match (*Max-Both*)
  - Best-Match S2.b1 zu S1.a1 wird nur berücksichtigt, wenn S1.a1 auch Best-Match von S2.b1 ist

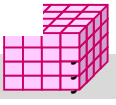


- Stable Marriage
  - Mapping M (Menge an Matches zw. S1 und S2) ist stabil, wenn es kein Paar (S1.ai, S2.bj) gibt, mit einer höheren Ähnlichkeit als die der aktuellen Matches in M sowohl für S1.ai und S2.bj

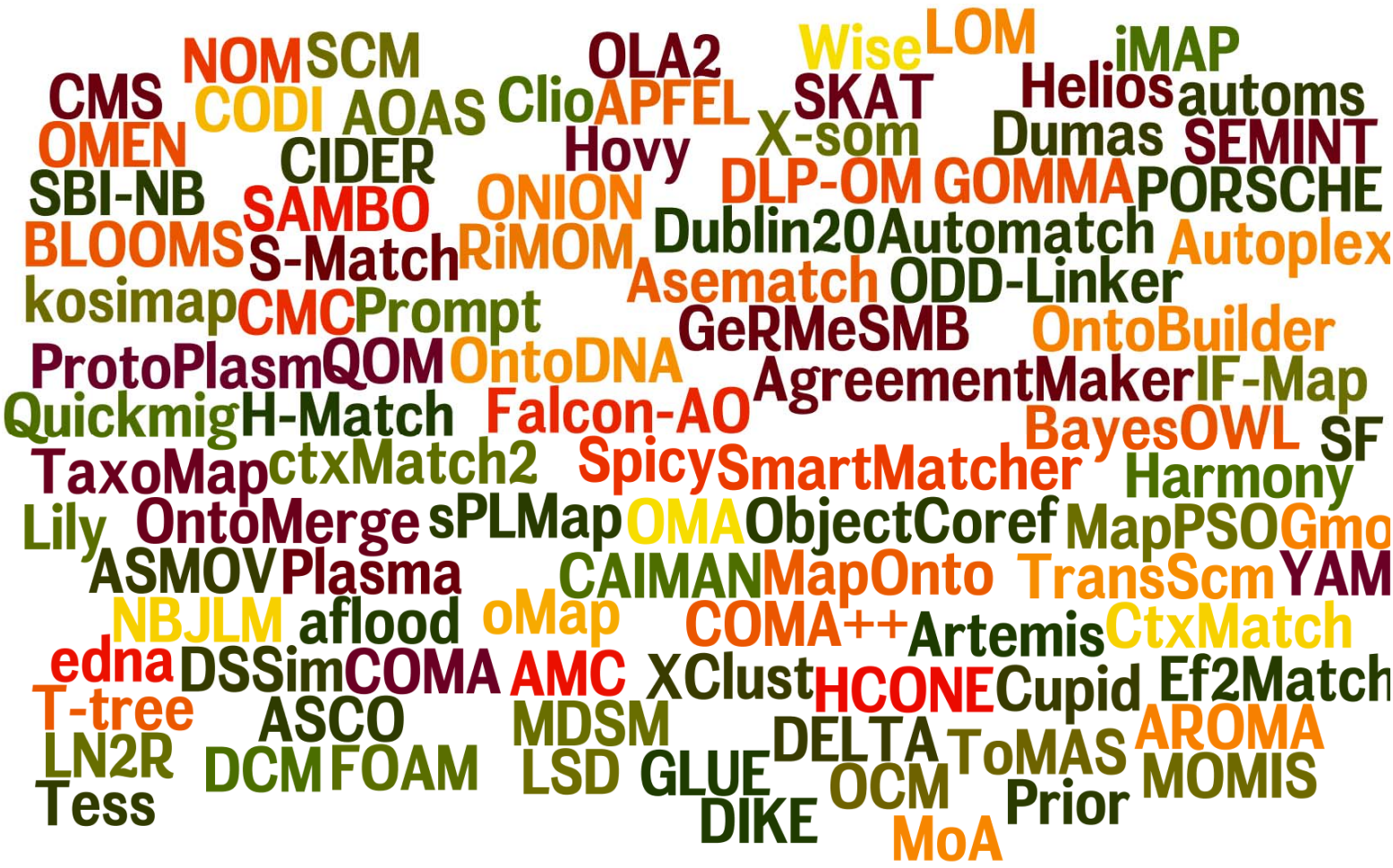


- Maximal Weight Mapping
  - 1:1 Mapping mit maximaler Summe der Ähnlichkeiten
  - Mapping muss nicht stabil sein





Match-Prototypen



Welche Tools am besten?

Die Wahl des richtigen Werkzeugs hängt stark von den Anforderungen ab:

ChatGPT

- Für **Ontologien und semantische Daten**: Tools wie **LogMap** oder **COMA++**.
- Für **ETL- und kommerzielle Datenintegration**: **Informatica**, **Talend** oder **IBM InfoSphere**.
- Für **Cloud- und Big-Data-Integration**: **Google Cloud Data Fusion**, **Amazon Glue**, oder **Apache Spark** / **Delta Lake**

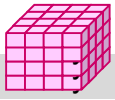
- Delta Lake
  - automatische Schemaextraktion aus strukturierten und teilstrukturierten Quellen (CSV, JSON, …)

5. Fazit: Möglichkeiten in Delta Lake

ChatGPT

Delta Lake allein bietet keine native automatische Unterstützung für Schema Matching. Es ist jedoch möglich:

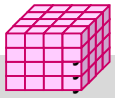
1. **Vergleiche** von Schemata auf technischer Ebene (z. B. Spaltennamen, Datentypen) durchzuführen.
2. **Benutzerdefinierte Mappings** oder **Regeln** zu implementieren.
3. **Externe Tools** und Frameworks zu integrieren, um semantische Korrespondenzen zu identifizieren.



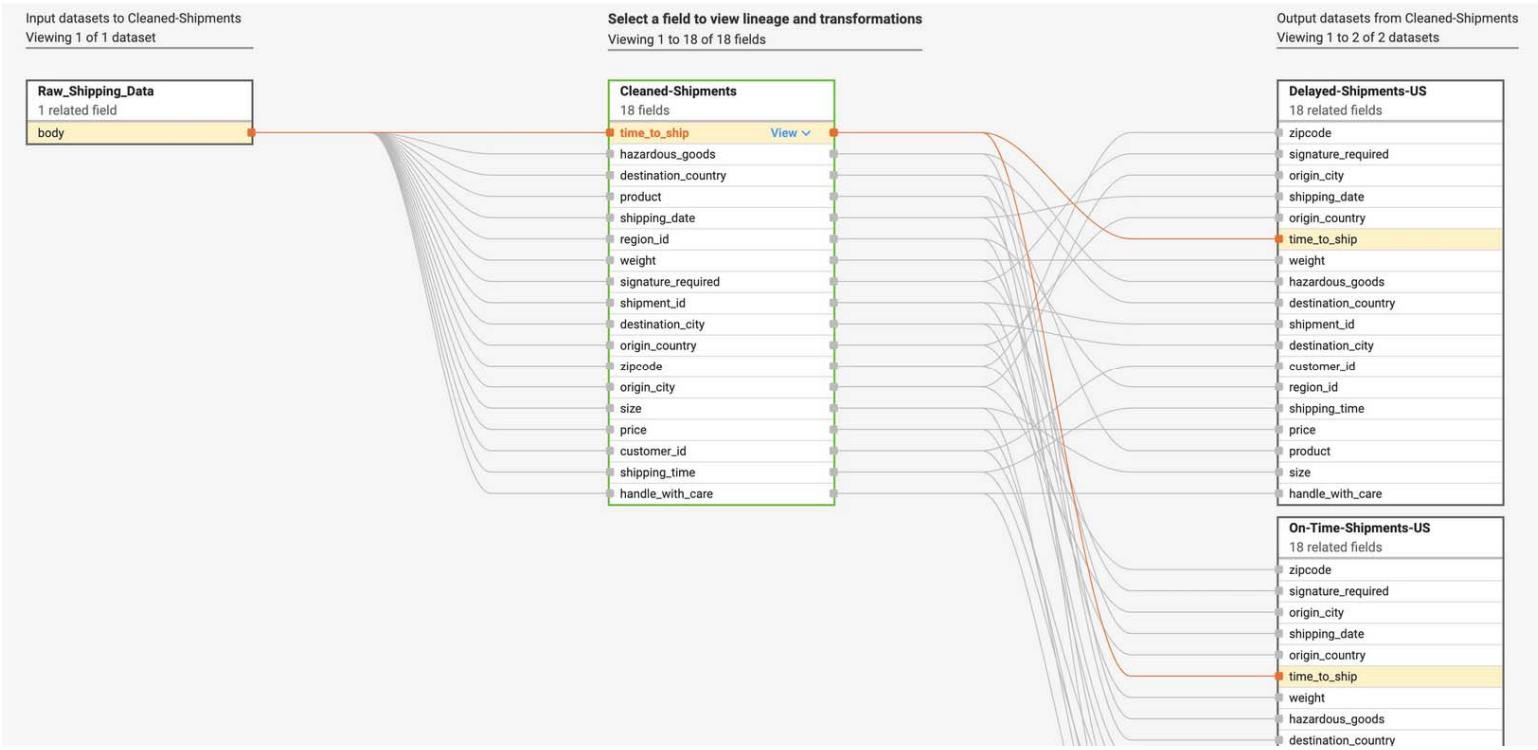
Google Cloud Data Fusion

- visuelle Definition von ETL-Pipelines mit Unterstützung von Schema-Mapping
  - automatische Schemaeerkennung
  - primär manuelles Schema-Mapping, jedoch können automatische Ansätze integriert werden
  - automatische Erkennung von Attributpaarn mit identischer Namen

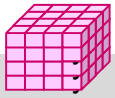
https://cloud.google.com/application-integration/docs/data-mapping-overview



Google Cloud Data Fusion (2)

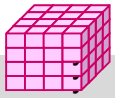


https://medium.com/gft-engineering/my-journey-with-google-cloud-data-fusion-dbe9e6f34924



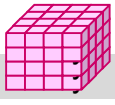
4. ETL: Schemaintegration + Data Cleaning

- ETL-Überblick
- Semantische Heterogenität
- Schema Matching
  - Verfahren
  - Prototypen / Tools
- Data Cleaning
  - Probleme
  - Teilaufgaben
- Entity Matching (record linkage)
  - Blocking, Matching, Clustering
  - Systeme/Prototypen: MS SQL-Server, Dedoop, Famer



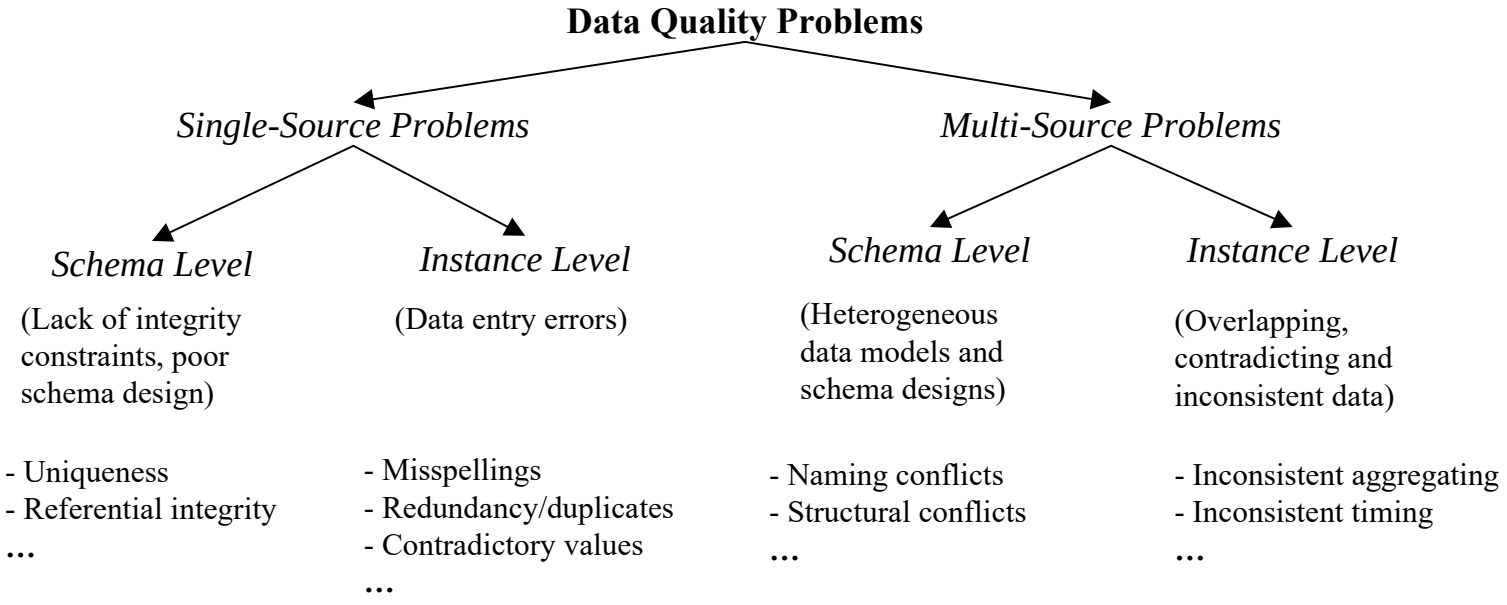
Data Cleaning

- Datenanalyse / Profiling
  - Entdeckung von Datenfehlern und -inkonsistenzen
  - manuell bzw. Einsatz von Analyse-Tools
- Definition von Mapping-Regeln und Transformations-Workflows
  - Datentransformationen auf Schemaebene
  - Cleaning-Schritte zur Behandlung von Instanzdaten
- Transformation
  - regelmäßige Ausführung der geprüften Transformationsschritte
- ggf. Rückfluss korrigierter Daten in operative Quellsysteme



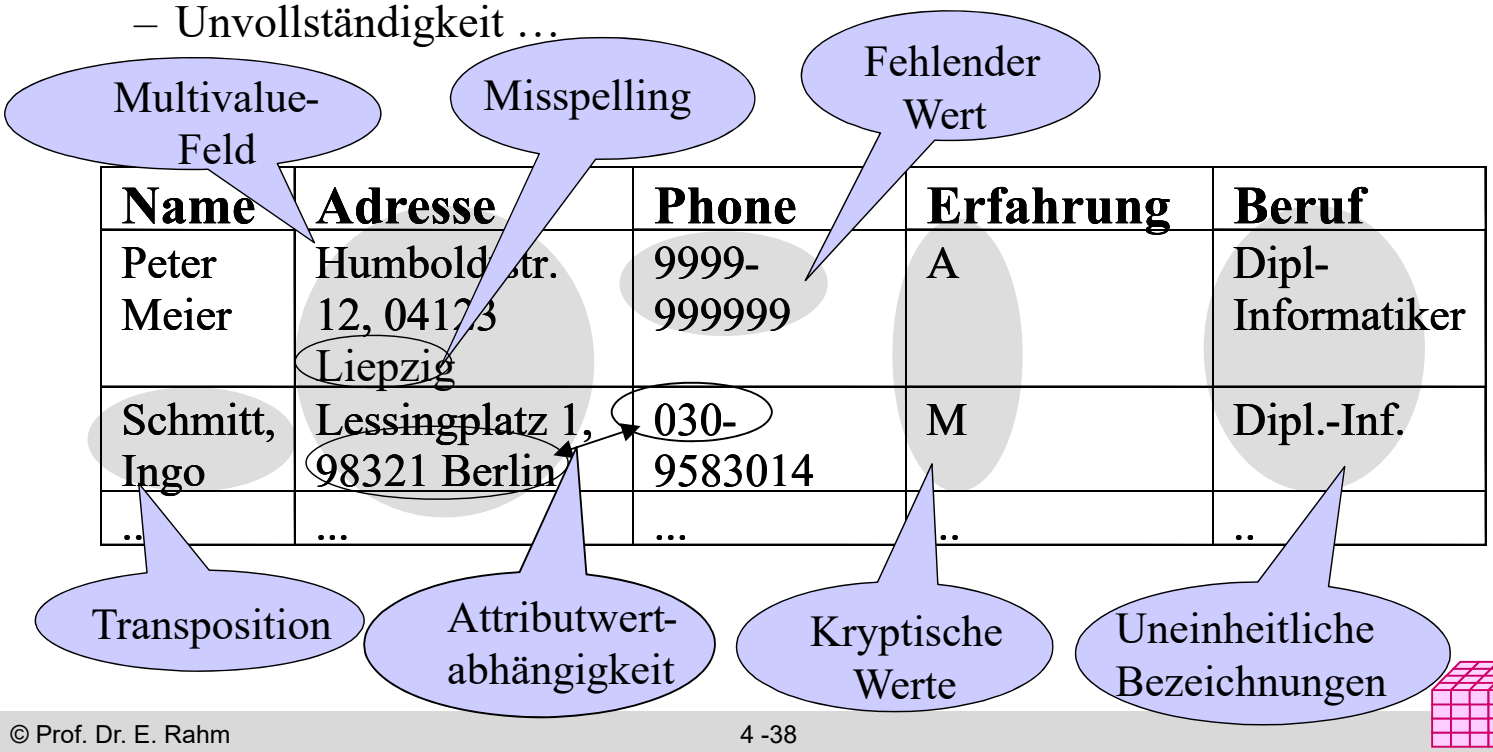
Probleme bezüglich Datenqualität

- Probleme auf Schema- und auf Instanzebene
- Probleme bezüglich einer oder mehrerer Datenquellen (Single-Source vs. Multi-Source)



Single-Source Probleme

- Ursachen:
  - Fehlen von Schemata (z.B. bei Dateien) und von Integritäts-Constraints
  - Eingabefehler
  - unterschiedliche Änderungsstände
  - Unvollständigkeit …



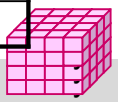
Datenanalyse / Profiling

- Entdeckung von Fehlern / Verifikation korrekter Werte
- Ableitung von (wirklichen) Metadaten
- Berechnung der Statistiken zu Attributen auf Basis ihrer Instanzen
  - Datentyp, Länge, Maximum und Minimum, Null-, Default-Werte, Kardinalität, ...
  - Ermitteln von Wertebereichen, Häufigkeiten und Mustern von Attributwerte
- Erkennung von Ausreißern, funktionalen Abhängigkeiten
  - SQL-Abfragen für Basis-Checks, zB

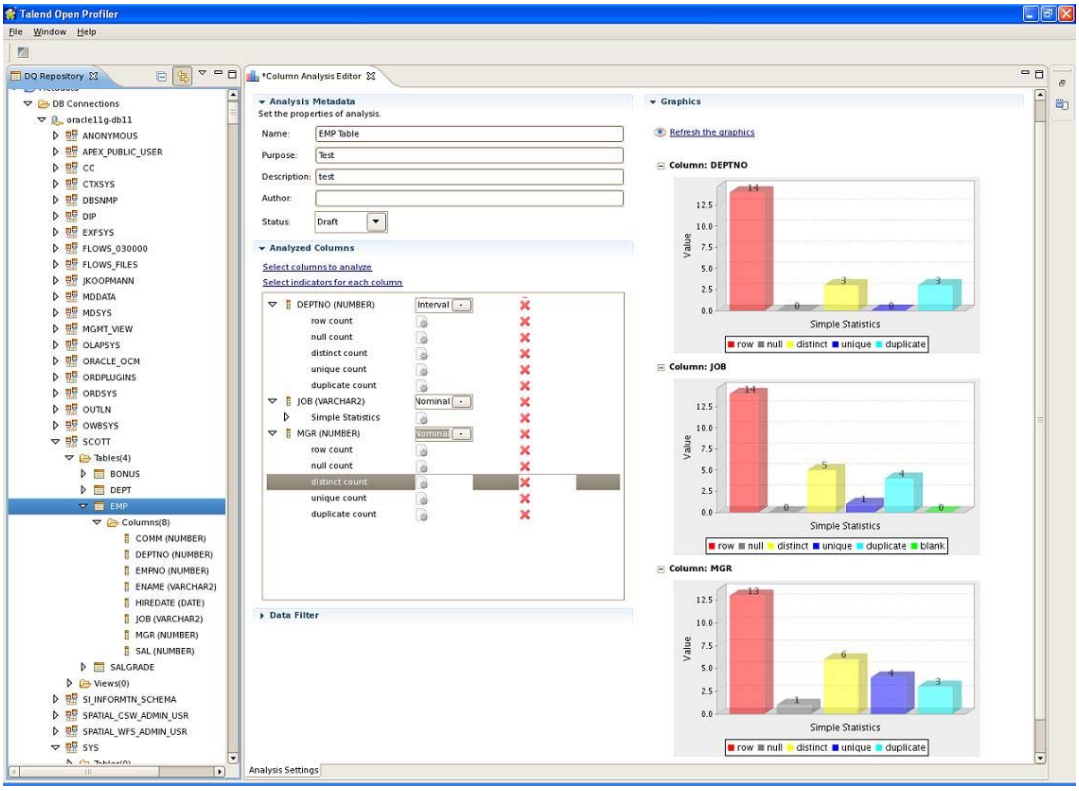
```
SELECT Stadt,count(*) as Anzahl From Student Group By Stadt order by 2
```

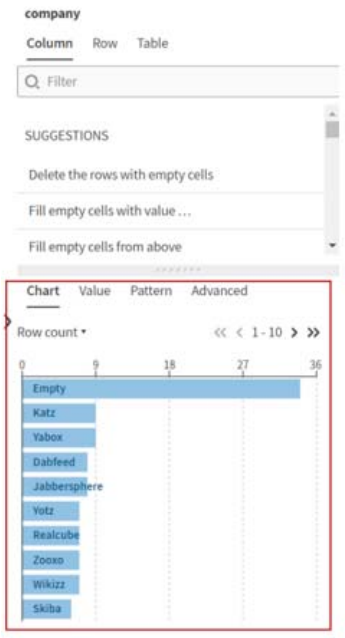
Attribute Values	#occurences
IBM	3000
I.B.M.	360
Intel Bus Mach	213
International Business Machine	36

Instanzwerte	Pattern	Identifizierte Datenkategorie
(978) 555-1212	(nnn) nnn-nnnn	Telefonnummer
036-55-1234	nnn-nn-nnnn	Social Security Number
abc@web.de	aaa@aaa.aa	Email-Addresse
12.03.2020	nn.nn.nnnn	Datum

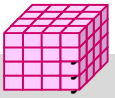


Bsp.: Talend Data Profiling





https://www.linux.com/news/explore-your-database-talend-open-profiler/

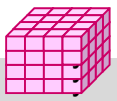


Behandlung von Single-Source-Problemen

- Definition und Einführung von Standardrepräsentationen
  - einheitliches Format für Datums-/Zeit-Angaben
  - einheitliche Groß/Kleinschreibungsform für Namen / String-Attribute
  - einheitliche Abkürzungen, Kodierungsschemas
- Bereitstellung von (Konversions-)Tabellen zur expliziten Werteabbildung

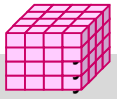
Legacy Value	New Value
IBM	IBM
I.B.M	IBM
Intel Bus Mach	IBM
...	...

- Extraktion von individuellen Werten aus Freiform-Attributen
  - Parsing und Attribut-Splitting, *z.B. Name -> Vorname / Nachname*
  - Reorganisierung der Wortreihenfolge
- Validierung / Korrektur mit Hintergrundwissen
  - Überprüfung/Spell checking mit Wörterbüchern, Datenbanken mit Adressen, Produktbezeichnungen, Akronymen/Abkürzungen, etc.
  - Nutzung bekannter Attributabhängigkeiten zur Korrektur von fehlenden / falschen Attributwerten, z.B. PLZ -> Ort



Multi-Source-Probleme

- unterschiedliche Repräsentationen der Instanzdaten
  - versch. Wertebereiche (z.B. *Geschlecht = {1,2} vs. Gender = {m,w}*)
  - verschiedene Einheiten (z.B. *Verkauf in EUR vs. Verkauf in Tsd.EUR*)
  - verschiedene Genauigkeiten
- unterschiedliche Änderungsstände und Aggregationsstufen
- überlappende, widersprüchliche bzw. inkonsistente Daten
- Hauptproblem: Behandlung überlappender Daten / Duplikate
  - Beschreibung einer Instanz der realen Welt durch mehrere Datensätze unterschiedlicher Quellen
  - oft nur teilweise Redundanz (einzelne Attribute, nur in Teilmenge der Datenquellen)
  - Behandlung durch Datenintegration
    - Duplikaterkennung, Record Linkage, Entity Matching
    - ggf. Fusion der sich entsprechenden Instanzen



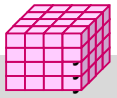
Multi-Source-Dateninkonsistenzen: Beispiel

Source1: Customer

CID	*Name*	*Street*	*City*	*Sex*
11	Kristen Smith	2 Hurley Pl	South Fork, MN 48503	0
24	Christian Smith	Hurley St 2	S Fork MN	1

Source2: Client

Cno	*LastName*	*FirstName*	*Gender*	*Address*	*Phone/Fax*
24	Smith	Christoph	M	23 Harley St, Chicago IL, 60633-2394	333-222-6542 / 333-222-6599
493	Smith	Kris L.	F	2 Hurley Place, South Fork MN, 48503-5998	444-555-6666



Beispiel (2)

CID	*Name*	*Street*	*City*	*Sex*
11	Kristen Smith	2 Hurley Pl	South Fork, MN 48503	0
24	Christian Smith	Hurley St 2	S Fork MN	1

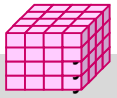
Source1: Customer

No	*LName*	*FName*	*Gender*	*Street*	*City*	*State*	*ZIP*	*Phone*	*Fax*	*CID*	*Cno*
1	Smith	Kristen L.	F	2 Hurley Place	South Fork	MN	48503-5998	444-555-6666		11	493
2	Smith	Christian	M	2 Hurley Place	South Fork	MN	48503-5998			24	
3	Smith	Christoph	M	23 Harley Street	Chicago	IL	60633-2394	333-222-6542	333-222-6599		24

Customers ***(Integrierte und bereinigte Daten)***

Cno	*LastName*	*FirstName*	*Gender*	*Address*	*Phone/Fax*
24	Smith	Christoph	M	23 Harley St, Chicago IL, 60633-2394	333-222-6542 / 333-222-6599
493	Smith	Kris L.	F	2 Hurley Place, South Fork MN, 48503-5998	444-555-6666

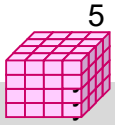
Source2: Client



Semistrukturierte Datensätze (Kameras)

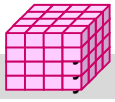
property	value
"35mm equivalent"	"25-300mm"
"<page title>"	"Nikon Coolpix S6800 Digital Camera (Black) | UK Digital Cameras"
"brand"	"Nikon"
"camera resolution"	"16 Megapixels"
"colour"	"Black",
"features"	"Slimline"
"hd video"	"Full HD (1080P)"
"lcd size"	"3.0""
"lens tele mm"	"300"
"lens wide mm"	"25"
"mpn"	"VNA520E1"
"optical zoom"	"23"
"optical zoom range"	"18x and higher"

property	value
"<page title>"	"Nikon Coolpix S6800 Price in India with Offers, Reviews & Full Specifications | PriceDekho.com"
"color"	"Black",
"amazon"	"Infibeam Ebay Homeshop18 Snapdeal Flipkart"
"digital zoom"	"4x"
"bangalore"	"Hyderabad Chennai Mumbai Delhi Pune"
"approx resolution"	"16 MP"
"external memory"	"Yes"
"face detection"	"NA"
"gps"	"NA"
"hdmi"	"NA"
"maximum shutter speed"	"1/2000 sec"
"metering"	"NA"
"minimum shutter speed"	"1 sec"
"optical zoom"	"18x"
"screen size"	"3 Inches"
"usb"	"Yes",
"video display resolution"	"NA"
"wifi"	"Yes; Wi-Fi 802.11 b/g/n"

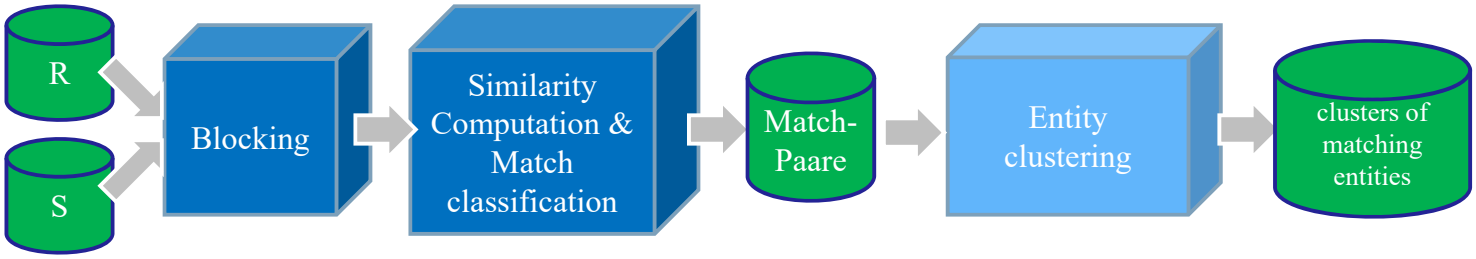


4. ETL: Datenvorverarbeitung und -integration

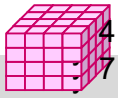
- ETL-Überblick
- Schemaintegration
  - Bottom-Up- vs. Top-Down-Integration
  - Semantische Heterogenität
- Schema Matching
  - Verfahren
  - Prototypen / Tools
- Data Cleaning
  - Probleme
  - Teilaufgaben
- Entity Matching (record linkage)
  - Blocking, Matching, Clustering
  - Systeme/Prototypen: MS SQL-Server, Dedoop, Famer



Entity Matching Workflow

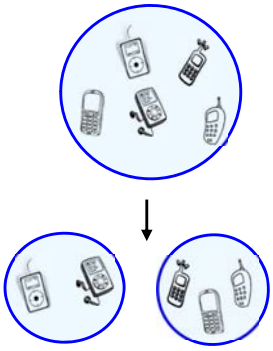


- Eingabe: 1, 2 or n Datenquellen mit Sätzen
- Ausgabe:
  - Paare von Satzmatches oder
  - Menge von Entity-Clustern, in denen alle Matches zu einer Entität gruppiert sind
- n>=2: saubere Datenquellen (ohne Duplikate) oder nicht
  - für saubere Quellen kann höchstens 1 Match pro Entität vorliegen: maximale Clustergröße n

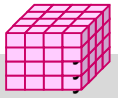


Blocking

- naives Matching: paarweiser Vergleich aller Sätze
  - quadratische Komplexität: 1 Quelle mit n Sätzen: n (n-1)/ 2 Vergleiche
  - bei 1 Million Vergleichen pro Sekunde dauert Matching für $n=10^7$ 19 Monate
  - skalierbare Lösungen erfordern Reduzierung der Vergleichszahl
- Blocking
  - Gruppierung ähnlicher Entitäten in Partitionen/Blöcke über Attributwerte (z.B. Produkttyp)
  - nur Entitäten eines Blocks werden miteinander verglichen

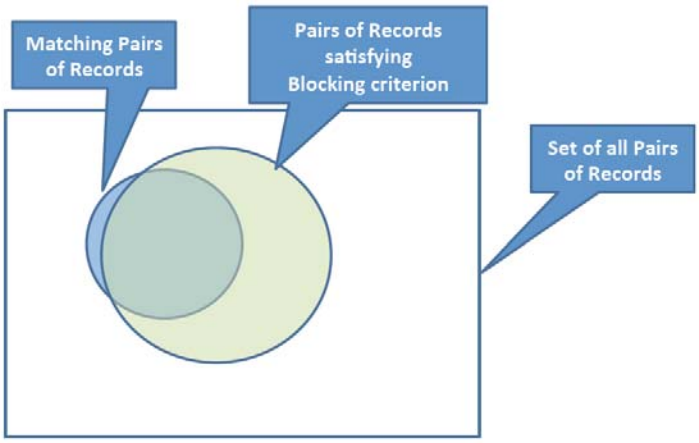


- Blockgröße wichtig für Leistung und Match-Qualität
  - kleine Blöcke: wenige Vergleiche, evtl werden Matches aber nicht gefunden
  - bei schmutzigen Daten ggf. mehrere Partitionierungen nötig (**multi-pass blocking**)
- Vielzahl an Blocking-Alternativen
  - Standard-Blocking
  - Sorted Neighborhood …

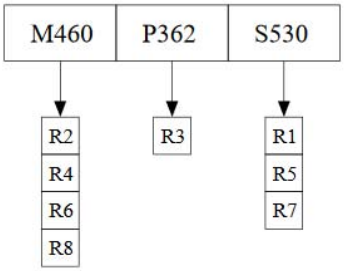


Standard Blocking

- Partitionierung des Suchraums über Blocking Key Values (BKV)
- BKV-Berechnung über Funktion auf Attributwerten, z.B.
  - Soundex (Nachname)
  - Postleitzahl
  - Präfix(Produkthersteller, 4), …
- Beispiel Soundex-Blocking:

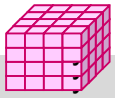


Identifiers	Surnames	BKVs (Soundex encoding)
R1	Smith	S530
R2	Miller	M460
R3	Peters	P362
R4	Myler	M460
R5	Smyth	S530
R6	Millar	M460
R7	Smyth	S530
R8	Miller	M460



8 Sätze: 28 Vergleiche ohne Blocking

P. Christen: A survey of indexing techniques for scalable record linkage and deduplication. IEEE TKDE, 2012



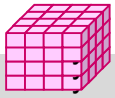
Sorted Neighborhood

- meist für 1 Eingabequelle
- Sortierung der Sätze gemäß *Sort Key*
- Matching nur auf benachbarten Sätzen (Fenster fester Länge *w*)
- lineare Komplexität
- Beispiel (w=3)

Window positions	BKVs (Surname)	Identifiers
1	Millar	R6
2	Miller	R2
3	Miller	R8
4	Myler	R4
5	Peters	R3
6	Smith	R1
7	Smyth	R5
8	Smyth	R7

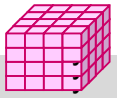
Window range	Candidate record pairs
1 – 3	(R6,R2), (R6,R8), (R2,R8)
2 – 4	(R2,R8), (R2,R4), (R8,R4)
3 – 5	(R8,R4), (R8,R3), (R4,R3)
4 – 6	(R4,R3), (R4,R1), (R3,R1)
5 – 7	(R3,R1), (R3,R5), (R1,R5)
6 – 8	(R1,R5), (R1,R7), (R5,R7)

P. Christen: A survey of indexing techniques for scalable record linkage and deduplication. IEEE TKDE, 2012



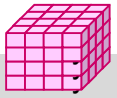
Metriken für Blocking

- Blocking sollte möglichst viele Vergleiche einsparen, ohne Matches zu verlieren
  - hohe Reduction Rate und sehr hohe Pairs Completeness erforderlich
- Metrik Reduction Rate
  - Reduction Rate (RR) = $1 - \frac{Anzahl\ der\ Vergleiche\ mit\ Blocking}{Anzahl\ der\ Vergleiche\ ohne\ Blocking}$
  - Reduktion auf 1% der Vergleiche: RR=0,99
- Metrik Pairs Completeness
  - Pairs Completeness (PC) = $\frac{Anzahl\ der\ Match-Paare,die\ verglichen\ werden}{Anzahl\ der\ Match-Paare}$
  - Erreichbarer Recall für Matching ist durch PC begrenzt
- Multi-Pass Blocking verbessert PC zu Lasten der RR



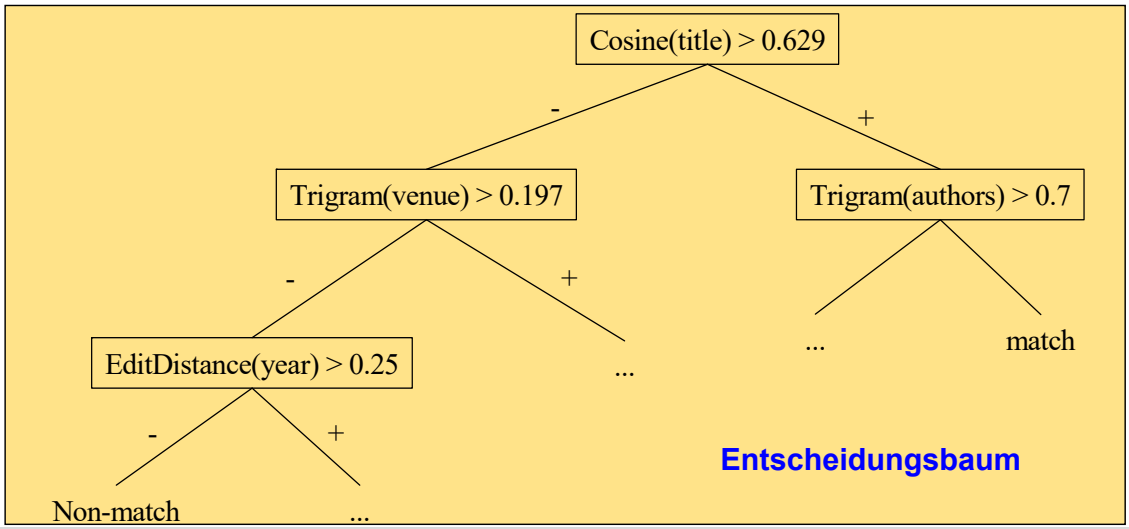
Matching

- kombinierte Nutzung mehrerer Ähnlichkeiten
  - Ähnlichkeit verschiedener Attributwerte, z.B. gemäß Stringähnlichkeit (Edit-Ditsance, Q-gram etc,) oder numerische Ähnlichkeit
  - kontext-basiertes Matching (z.B. Personen ähneln sich, wenn sie die selben Eltern haben)
- Verwendung von manuell definierten Match-Regeln
  - Personen-Match falls Namensähnlichkeit > 80% + gleiches Geburtsdatum
  - Publikations-Match: Titelähnlichkeit > 85% + Autorähnlichkeit > 50%
  - durchschnittliche Ähnlichkeit mehrerer Attribute > 75% …
- Nutzung überwachter Klassifikationsverfahren (machine learning)
  - z.B. mit Entscheidungsbäumen, SVM, neuronalen Netzen, etc.
  - erfordern Trainingsdaten mit Matches/Non-Matches
  - Nutzung von Embeddings (representation learning) kann Matchbarkeit von Attributwerten verbessern



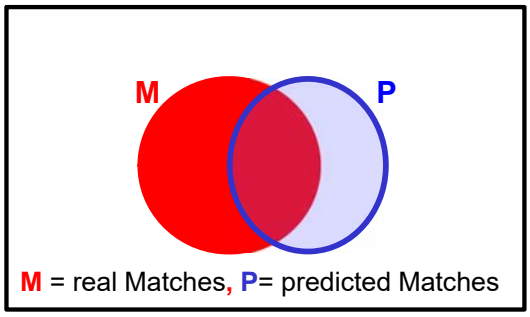
Beispiel ML-basiertes Matching

- Finden effektiver Match-Einstellungen ist schwierig
  - Auswahl der Attribute, Matcher, Einstellungen
- gelerntes Klassifikationsmodell (z.B. Entscheidungsbaum) bestimmt automatisch
  - zu prüfende Attribute (z.B. Publikatonstitel, Autoren, …)
  - Ähnlichkeitsmetriken (EditDistance, Cosine, ..)
  - Schwellwerte



Metriken für Matching

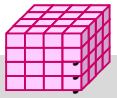
- möglichst alle tatsächlichen Matches zu finden (guter Recall) und möglichst wenige falsche Matches (hohe Precision)
- Begriffe
  - A = all Pairs
  - True Positives (TP) = P ∩ M
  - False Positives (FP)= P – M
  - True Negatives (TN) =A-FP
  - False Negatives (FN)= M - P



- Metriken
  - Recall $= \frac{|TP|}{|M|}$ Precision $= \frac{|TP|}{|P|}$
  - F-Measure (F1-Metrik) $= \frac{2\,Recall * Precision}{Recall + Precision}$

Entity Clustering

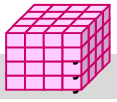
- Gruppierung der Matches einer Entität in Cluster
  - Cluster mit k Sätzen repräsentiert k (k-1)/2 Match-Paare: kompaktere Repräsentation für k>2
  - für n duplikatfreie Quellen gilt k <= n
  - Fusionierung der Cluster-Mitglieder zur Datenintegration
- Entity clustering
  - Input: paarweise Match-Kandidaten mit ihrer Ähnlichkeit (bilden sog. *Similarity Graph*)
  - Ziel: Maximierung der Ähnlichkeit innerhalb der Cluster und Minimierung der Ähnlichkeit zwischen Cluster
- Basisansatz: Connected Component (transitive Hülle) über paarweise Match-Links
  - kann leicht zu großen Clustern mit unähnlichen Sätzen führen



MS SQL-Server: Match-Operatoren

- Bestandteil von SQL-Server Integration Services (SSIS)*
  - ermöglicht Definition komplexer ETL-Workflows
  - zahlreiche Operatoren
- Fuzzy Lookup
  - "Fuzzy Match" zwischen Eingaberelation und sauberen Sätzen einer Referenztabelle: *inkrementelles Matching*
  - Parameter: Schwellwerte bzgl. String-Ähnlichkeit (Edit Distance) sowie Gewichte zur Kombination von Ähnlichkeiten
- Fuzzy Grouping
  - Gruppierung ähnlicher Sätze (potenzielle Duplikate) innerhalb einer Tabelle über String-Matching (Edit Distance)

* http://msdn.microsoft.com/en-us/library/ms345128.aspx

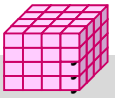


Match-Prototypen

- DEDOOP (U Leipzig, 2012)
  - paralleles paarweises Entity Matching auf Hadoop
  - Unterstützung für Blocking, lernbasiertes Matching und Lastbalancierung

- FAMER (U Leipzig, seit 2017)
  - paralleles Matching für viele (>2) Datenquellen
  - Unterstützung für **Entity Clustering**

- Magellan
  - Open-Source Python Framework mit Unterstützung von Blocking und Matching
  - Unterstützung für ML-Matching, inkl. Deep Learning
  - visuelle Oberfläche und gut dokumentierte API



Duplikate in Webdaten: Beispiel

Canon VIXIA HF S10 Camcorder - 1080p - 8.59 MP - 10 x optical zoom
Flash card, 32 GB, 1y warranty, F/1.8-3.0
The VIXIA HF S10 delivers brilliant video and photos through a Canon exclusive 8.59 megapixel CMOS image sensor and the latest version of Canon's advanced image processor, ...
★★★★★ 12 reviews - Add to Shopping List

$975 new
from 52 sellers
Compare prices

Canon (VIXIA) HF S10 VIS Dual Flash Memory Camcorder
Canon HF S10 iVIS Dual Flash Memory CamcorderSPECIAL SALE PRICE: $899 Display both English/Japanese + we supplu all English manuals in English as PDF. ....
Add to Shopping List

$899.00 new
Made in Japan Online

Canon VIXIA HF S10
Dual Flash Memory High Definition Camcorder The Next Step Forward in HD Video Canon has a well-known and highly-regarded reputation for optical excellence, ...
Add to Shopping List

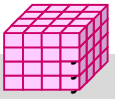
$999.00 new
Performance Audio
2 seller ratings

Canon VIXIA HF S100 Flash Memory Camcorder
***Canon Video HF S100 Instant Rebate Receive $200 with your purchase of a new Canon VIXIA HF S100 Flash Memory Camcorder. (Price above includes $200 ....
Add to Shopping List

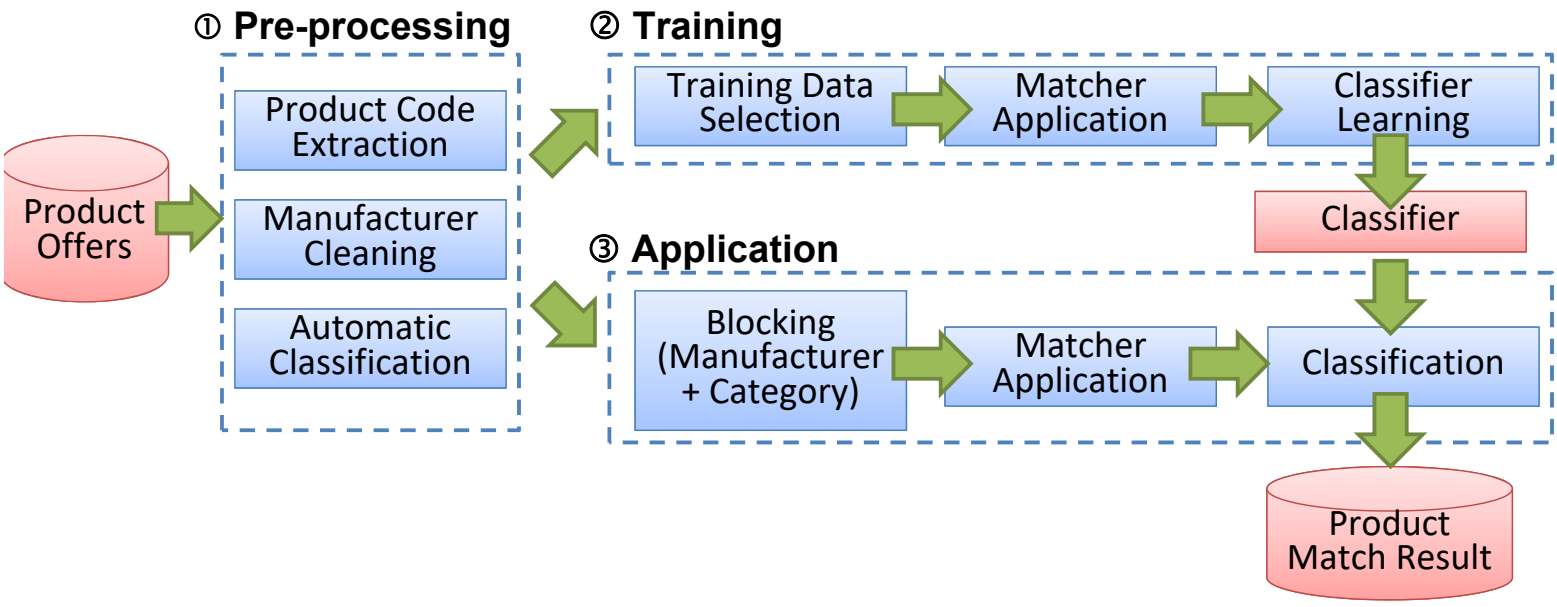
$899.95 new
Arlingtoncamera.com
5 seller ratings

Canon Vixia Hf S10 Care & Cleaning
Care & Cleaning Digital Camera/Camcorder Deluxe Cleaning Kit with LCD Screen Guard Canon VIXIA HF S10 Camcorders Care & Cleaning.
Add to Shopping List

$2.99 new
shop.com
★★½☆☆ 38 seller ratings



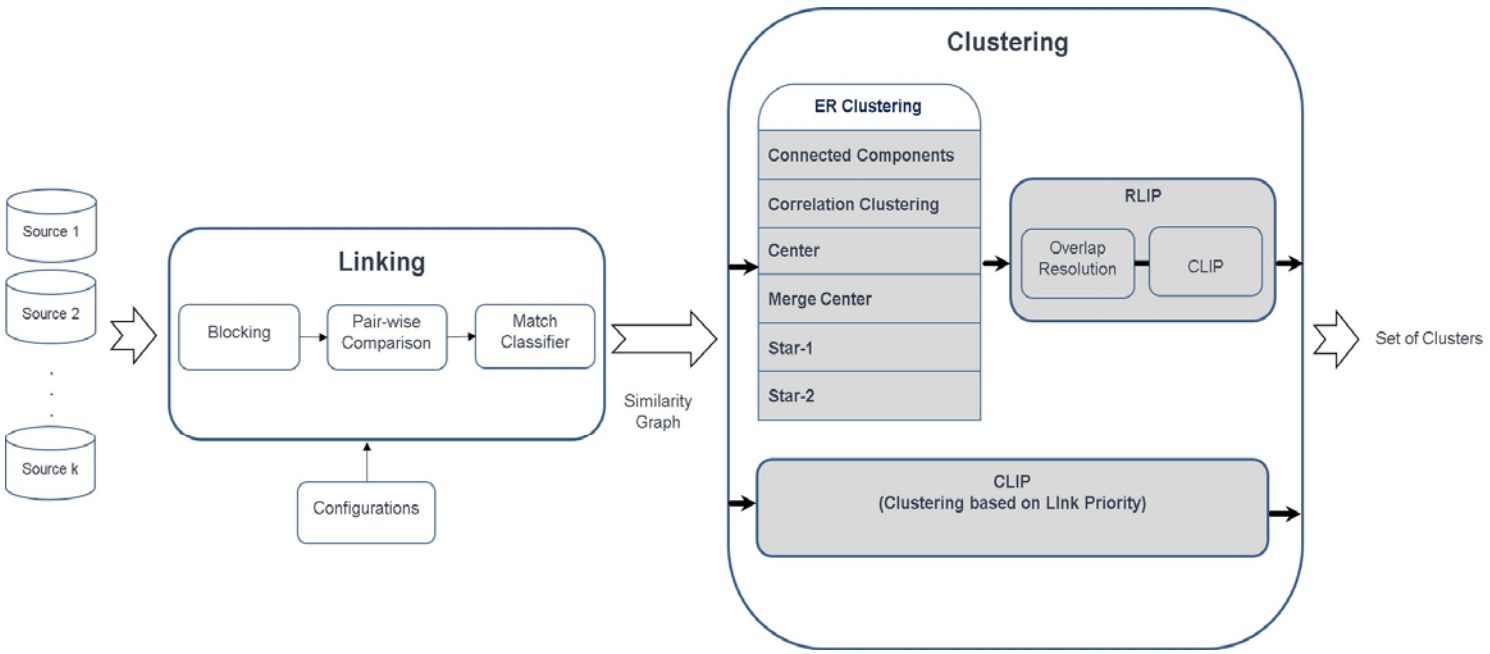
Workflow Product Matching



Koepcke, Thor, Thomas, Rahm: *Tailoring entity resolution for matching product offers*. Proc. EDBT, 2012

FAMER

- FAst Multi-source Entity Resolution system
  - Annahme: duplikatfreie Eingabequellen



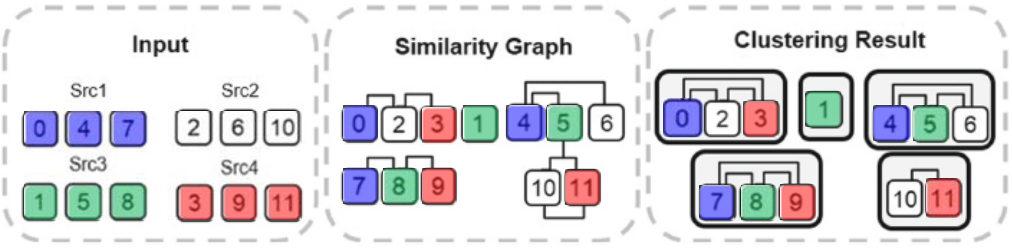
https://dbs.uni-leipzig.de/research/projects/object_matching/famer

FAMER-Beispiel

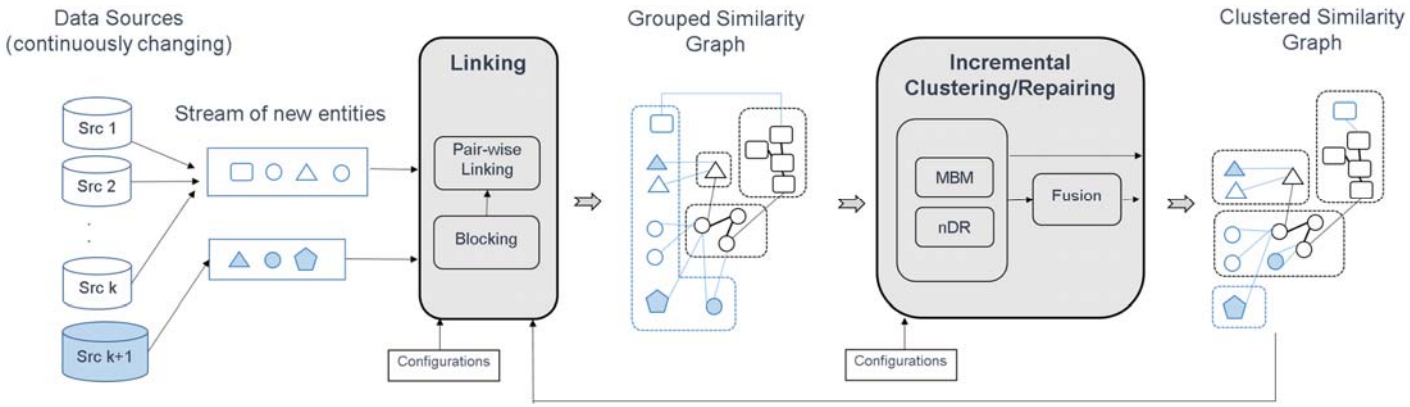
Eingabedaten aus 4 Quellen

Id	Name	Surname	Suburb	Post code	SourceId
0	ge0rge	Walker	winston salem	271o6	Src1
1	George	Alker	winstom salem	27106	Src2
2	George	Walker	Winstons	27106	Src3
3	Geoahge	Waker	Winston	271oo	Src4
4	Bernie	Davis	pink hill	28572	Src1
5	Bernie	Daviis	Pinkeba	2787z	Src2
6	Bernii	Davs	pink hill	28571	Src3
7	Bertha	Summercille	Charlotte	28282	Src1
8	Bertha	Summeahville	Charlotte	2822	Src2
9	Brtha	Summerville	Charlotte	28222	Src4
10	Bereni	dan'lel	Pinkeba	27840	Src3
11	Bereni	Dasniel	Pinkeba	2788o	Src4

FAMER-Anwendung



Inkrementelles Matching/Clustering mit FAMER*

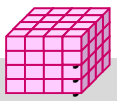


- Integration neuer Entitäten/Datenquellen
- Matching neuer Entitäten mit existierendem Similarity Graph
  - Zuweisung eines neues Entity zu ähnlichstem Cluster bzw. neues Cluster
- optionale Anpassung (Reparatur) bisheriger Cluster

*A. Saeedi, E. Peukert, E. Rahm: Incremental Multi-source Entity Resolution for Knowledge Graph Completion. Proc. ESWC 2020

Zusammenfassung (1)

- ETL als komplexer, aufwändiger Integrationsprozess
- Schema- und Datenintegration / Data Cleaning
  - zahlreiche Schema- und Datenkonflikte
  - Unterscheidung quell-lokaler und -übergreifender Datenkonflikte
  - Data Profiling erster Schritt zur Datenbereinigung
- Fokussierung auf Data Warehouse-spezifisches Zielschema erleichtert Schemaintegration
  - Top-Down-Schemaintegration
  - keine vollständige Integration aller Quell-Schemata erforderlich
- wichtiges Teilproblem: Schema-Matching
  - Bestimmung semantischer Korrespondenzen
  - Nutzung und Kombination mehrerer Matcher, u.a. linguistischer, struktureller und instanzbasierter Verfahren



Zusammenfassung (2)

- zentrales Problem: Duplikat-Identifikation und –Behandlung (Entity Matching)
  - hohe Effizienzanforderungen (-> Nutzung von Blockingverfahren und ggf. Parallelisierung)
  - kombinierte Nutzung mehrerer Match-Verfahren
  - trainingsbasierte Klassifikationsansätze erleichtern Konfiguration des Match-Schrittes
  - Entity Clustering gruppiert alle Matches (Varianten) pro Entity
- inkrementelles Matching / Clustering erforderlich
  - Abgleich neuer mit bereits existierenden Datensätzen
  - Unterstützung u.a. in FAMER-Prototyp
- zahlreiche Tools / Produkte mit stark unterschiedlicher Funktionalität
  - zunehmend ML-Unterstützung für Entity-Matching