

# Informationsintegration

## Antrittsvorlesung

am Tag der Informatik, 5.5.2004

Felix Naumann

naumann@informatik.hu-berlin.de

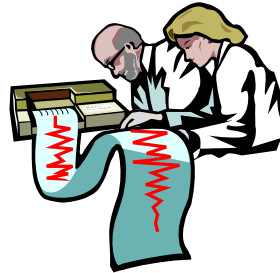


## Arbeitsgruppe Informationsintegration

- Juniorprofessur
  - 2 x 3 Jahre
- Gefördert durch DFG
  - „Aktionsplan Informatik“
  - Seit 5. Mai 2003

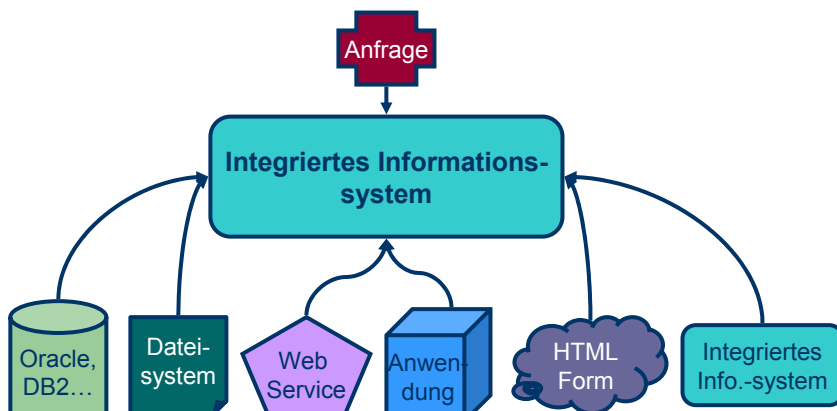
## Einige Untertitel

- Content Merging
- Objekt/Data Fusion
- Data Amalgamation
- Data Consolidation
- Intelligent Information Integration
- Data Cleansing
- Datenintegration
- Datenverschmelzung



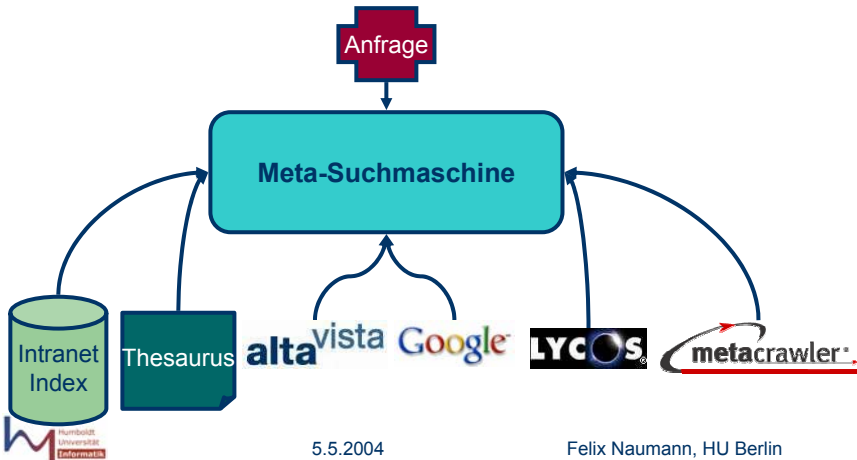
3

## Integrierte Informationssysteme



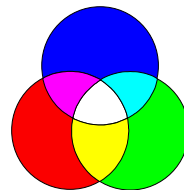
4

# Integrierte Suchmaschinen

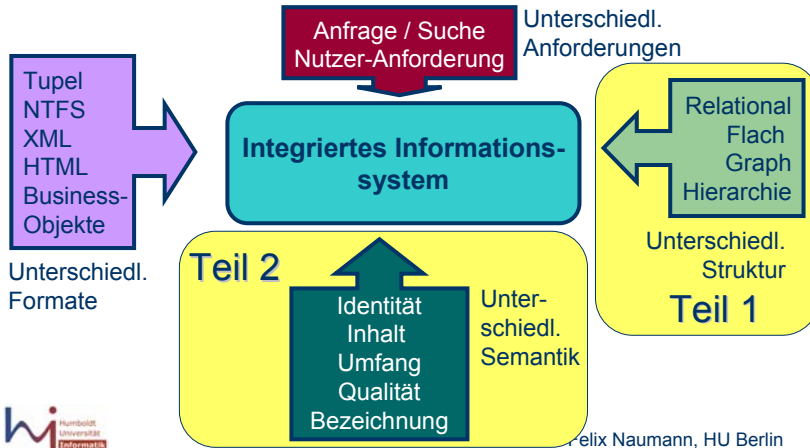


# Redundanz – pro & contra

- Ist gut, denn sie bietet
  - mehr Informationen,
  - detaillierter Informationen,
  - und verifizierbare Informationen.
  - Deshalb sollten wir integrieren!
- Ist problematisch, denn
  - Redundanz herrscht nur konzeptionell.
  - Technische und strukturelle Schwierigkeiten
  - Konflikte und Alternativen
  - Deshalb ist Informationsintegration interessant.



# Integrierte Informationssysteme

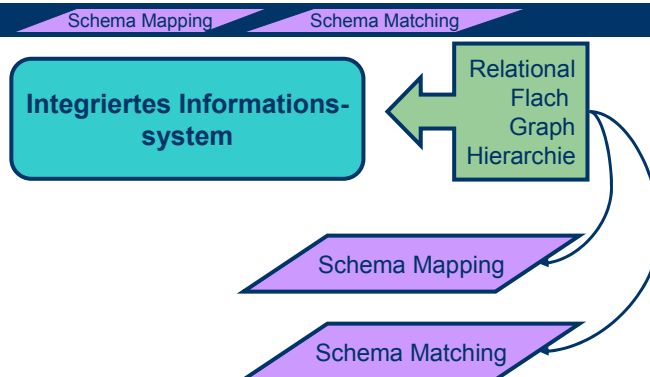


7



Felix Naumann, HU Berlin

# Lösung unterschiedlicher Struktur



8



5.5.2004

Felix Naumann, HU Berlin

# Schematische Heterogenität



Schema Mapping

Schema Matching

```
Männer( Id, Vorname, Nachname)
Frauen( Id, Vorname, Nachname)
```

Relation vs. Attribut

```
Person( Id, Vorname,
        Nachname, männlich,
        weiblich)
```

Relation vs. Wert

```
Person( Id, Vorname,
        Nachname, Geschlecht)
```

Attribut vs. Wert

9



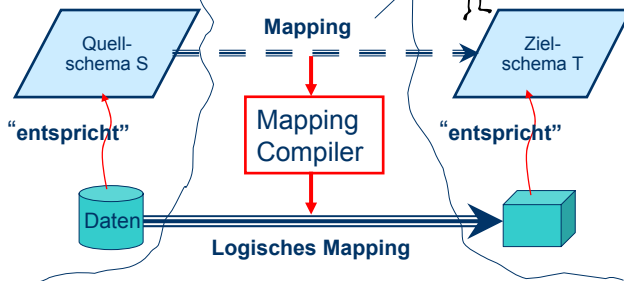
5.5.2004

Felix Naumann, HU Berlin

# Schema Mapping im Kontext

Schema Mapping

Schema Matching



- Möchte Daten aus S
- Versteht/Kennt T
- Versteht nicht immer S

10



5.5.2004

Felix Naumann, HU Berlin

# Motivation: Warum ist Schema Mapping nützlich?

Schema Mapping

Schema Matching

- Datentransformation zwischen heterogenen Schemata
  - Üblicherweise schreiben Experten komplexe Anfragen oder Programme
    - Zeitintensiv
    - Experte für die Domäne, für Schemata und für Anfrage
    - XML macht alles noch schwieriger
- Idee: Automatisierung
  - Gegeben: Zwei Schemata und ein high-level Mapping dazwischen
  - Gesucht: Anfrage zur Datentransformation

11



5.5.2004

Felix Naumann, HU Berlin

# Motivation: Warum ist Schema Mapping schwierig?

Schema Mapping

Schema Matching

- Generierung der “richtigen” Anfrage unter Berücksichtigung
  - des Quell und Ziel-Schemas,
  - des Mappings,
  - und der Nutzer-Intention.
  - Fagin et al. 2002
- Garantie, dass die transformierten Daten dem Zielschema entsprechen.
- Effiziente Datentransformation
  - Für materialisierte Integration
  - Für virtuelle Integration

12



5.5.2004

Felix Naumann, HU Berlin

# Schema Mapping Beispiel

Schema Mapping

Schema Matching

- ARTICLE
  - artPK
  - title
  - pages
- AUTHOR
  - artFK
  - name
- PUBLICATION
  - pubID
  - title
  - date
  - author

```
SELECT artPK AS pubID UNION SELECT null AS pubID
       title AS title   null AS title
       null AS date     null AS date
       null AS author   name AS author
FROM   ARTICLE          FROM   AUTHOR
```

13



5.5.2004

Felix Naumann, HU Berlin

# Schema Mapping Beispiel

Schema Mapping

Schema Matching

- ARTICLE
  - artPK
  - title
  - pages
- AUTHOR
  - artFK
  - name
- PUBLICATION
  - pubID
  - title
  - date
  - author

```
SELECT artPK AS pubID
       title AS title
       null AS date
       name AS author
FROM   ARTICLE, AUTHOR
WHERE  ARTICLE.artPK = AUTHOR.artFK
```

14



5.5.2004

Felix Naumann, HU Berlin

# Schema Matching – Motivation

Schema Mapping

Schema Matching

- Große Schemata
  - > 100 Tabellen, viele Attribute
  - Bildschirm nicht lang genug
- Unübersichtliche Schemata
  - Tiefe Schachtelungen
  - Fremdschlüssel
  - Bildschirm nicht breit genug
  - XML Schema
- Fremde Schemata
  - Unbekannte Synonyme
- Irreführende Schemata
  - Unbekannte Homonyme
- Fremdsprachliche Schemata
- Kryptische Schemata
  - |Attributnamen| ≤ 8 Zeichen
  - |Tabellennamen| ≤ 8 Zeichen

15



5.5.2004

Felix Naumann, HU Berlin

The screenshot shows a software interface for schema mapping. The main window is divided into two panes: 'Source schemas' on the left and 'Target Schema' on the right. The 'Source schemas' pane displays a highly nested tree structure of schema elements, including records, sets, and attributes. The tree is very deep and wide, illustrating the complexity mentioned in the text. Two callout boxes are present: one pointing to a vertical scrollbar on the right side of the source schema tree, and another pointing to the depth of the nesting in the tree structure.

Man beachte die Scrollbar!

Man beachte die Schachtelungstiefe!



# Schema Matching Klassifikation

Schema Mapping

Schema Matching

- Schema Matching basierend auf
  - Namen der Schemaelemente (*label-based*)
  - Darunterliegende Daten (*instance-based*)
  - Struktur des Schemas (*structure-based*)
  - Mischformen, Meta-Matcher
  - (Rahm und Bernstein 2001)
- Kernalgorithmus des Schema Matching:
  - Gegeben zwei Schemata mit Attributmengen A und B.
  - Bilde Kreuzprodukt aller Attribute aus A und B.
  - Für jedes Paar vergleiche Ähnlichkeit.
  - Ähnlichste Paare sind Matches.

17



5.5.2004

Felix Naumann, HU Berlin

# Schema Matching – Label-basiert

Schema Mapping

Schema Matching

- Ähnlichkeitsmaß: Ähnlichkeit der Attributnamen
- Beispiele:
  - $\text{Ähnlichkeit}(\text{'Name'}, \text{'Name'}) = 1$
  - $\text{Ähnlichkeit}(\text{'Titel'}, \text{'title'}) = 0.8$
- Edit-distance
  - Anzahl der Edit-Operationen um String 1 in String 2 umzuwandeln.
- Probleme:
  - Effizienz
  - Auswahl der besten Matches
    - Iterativ,
    - Stable Marriage, etc.
  - Synonyme und Homonyme werden nicht erkannt

18



5.5.2004

Felix Naumann, HU Berlin

# Schema Matching – Instanzbasiert

Schema Mapping

Schema Matching

- Ähnlichkeitsmaß: Ähnlichkeit der Attributwerte
  - Typische Eigenschaften
    - Länge,
    - Existenz eines Trennzeichens,
    - Mischung Groß- und Kleinschreibung,
    - Buchstabenverteilung, etc.
- Probleme
  - Auswahl und Gewichtung der Eigenschaften
  - Datenmenge: Sampling
  - Vergleichsmethode, z.B. Naive Bayes

19



5.5.2004

Felix Naumann, HU Berlin

# Schema Matching – Strukturbasiert

Schema Mapping

Schema Matching

- Ähnlichkeitsmaß: Elemente der „Nachbarschaft“ von Attributen
- Kernidee
  - Nutze (komplexe) Struktur der Schemata aus.
    - Hierarchieebene
    - Elementtyp (Attribut, Relation, Label,...)
    - Nachbarschaftsbeziehungen
  - Wenn Element  $x$  ähnliche Nachbarn hat wie Element  $y$ , sind  $x$  und  $y$  ähnlich.
- Effektiv durch Kombination mit anderen Methoden

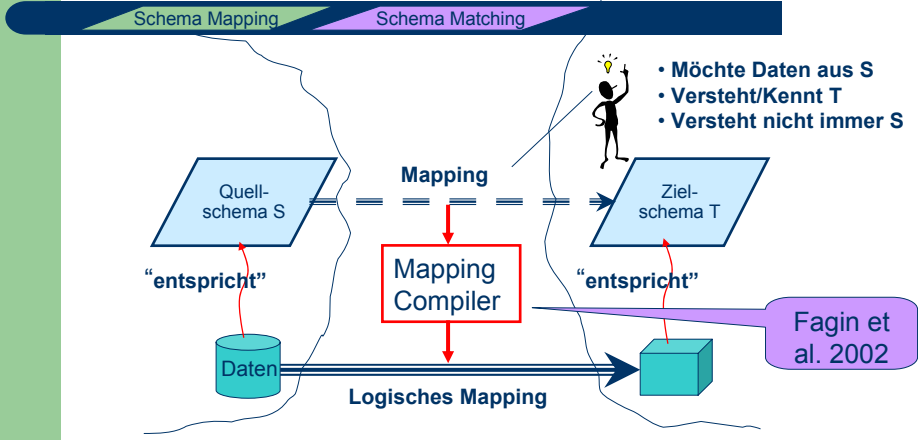
20



5.5.2004

Felix Naumann, HU Berlin

# Schema Mapping im Kontext



21



5.5.2004

Felix Naumann, HU Berlin

Clio      Clio

File Database Mappings Help      File Database Mappings Help

Source Schemas      Target Schema

expenseDB: Record      statisticsDB: Set

Set of (company)      cityStatistics: Record

company: Record      city (string)

cid (string)      Set of (organization)

cname (string)      cid (string)

city (string)      cname (string)

Set of (grant)      Set of (funding)

grant: Record      gid (string)

cid (string)      proj (string)

amount (string)      fald (string)

project (string)      recv (string)

Set of (project)      Set of (financial)

projects: Record      financial: Record

name (string)      sid (string)

year (string)      amount (string)

     date (string)

     Set of (project)

     project: Record

     name (string)

     year (string)

Schema Mapping      Transformations-anfrage

```

XQuery
XSL
SQL Query

xmlns:
xmlns:
xmlns:

$xmlL1/project/text() = $xmlL1/name/text() AND
$xmlL2/cid/text() = $xmlL2/cid/text() AND
$xmlL1/cid/text() = $xmlL1/cid/text() AND
$xmlL2/city/text() = $xmlL2/city/text()
RETURN
<organization>
<cid> $xmlL1/cid/text() </cid>,
<cname> $xmlL1/cname/text() </cname>,
distinct (
FOR
$xmlL2 IN $doc/expenseDB/grant,
$xmlL1 IN $doc/expenseDB/project,
$xmlL2 IN $doc/expenseDB/company
WHERE
$xmlL2/project/text() = $xmlL2/name/text() AND
$xmlL2/cid/text() = $xmlL2/cid/text() AND
$xmlL1/cname/text() = $xmlL2/cname/text() AND
$xmlL1/city/text() = $xmlL2/city/text() AND
$xmlL1/cid/text() = $xmlL2/cid/text()
RETURN
<funding>
<gid> $xmlL2/gid/text() </gid>,
<proj> $xmlL2/project/text() </proj>,
<fald> "Sk267(", $xmlL2/project/text(), ",
</funding>
)
</organization>
),
distinct (
FOR
$xmlL1 IN $doc/expenseDB/grant,
$xmlL1 IN $doc/expenseDB/project,
$xmlL1 IN $doc/expenseDB/company
WHERE

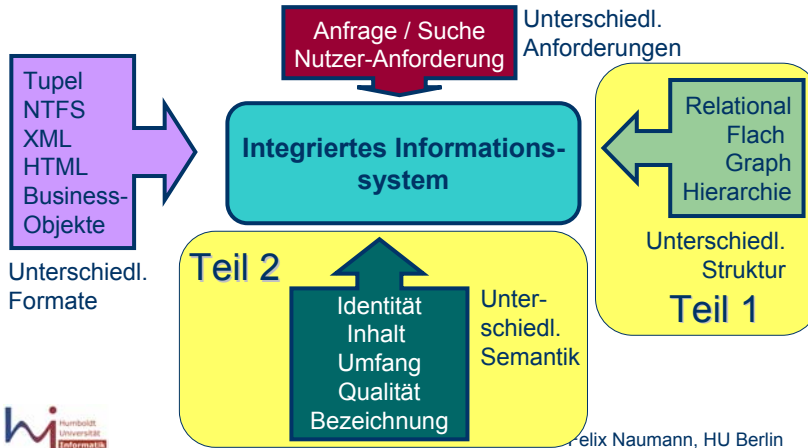
```

Preview      Execute Query      Copy to Clipboard

No File

22      Humboldt Universität Informatica      5.5.2004

# Integrierte Informationssysteme

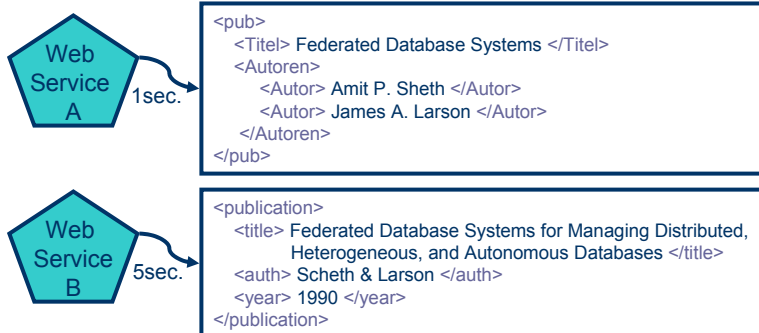


23



Felix Naumann, HU Berlin

# Beispiel der Informationsintegration



Identifikation

Integration

Optimierung

Visualisierung

24



5.5.2004

Felix Naumann, HU Berlin

# Objektidentifikation

Identifikation   Integration   Optimierung   Visualisierung

- Erkennung mehrere (ähnlicher) Darstellungen des gleichen Objekts
- Auch
  - Duplicate Detection
  - Data Cleansing
  - Record Linkage
- Domänenspezifische Algorithmen
  - Adressdaten
  - Mikrobiologische Daten
- Manchmal gibt es IDs
  - Bücher: ISBN
  - Personen: SSN / Personalausweisnummer
  - Webseiten: URL



25



5.5.2004

Felix Naumann, HU Berlin

# Objektidentifikation

Identifikation   Integration   Optimierung   Visualisierung

- Das klassische Problem
  - Finde Duplikate innerhalb einer Tabelle.
  - Sehr große Datenmenge
    - kein quadratischer Algorithmus
    - kein Hauptspeicher-Algorithmus
- Forschung
  - Sorted-Neighborhood Methode (Hernandez, Stolfo 1998)
  - Record Linkage (Fellegi, Sunter, 1969), etc.
- Industrie
  - Trillium, Vality, ETI, et al.

Vorname	Nachname	Adresse	ID
Sal	Stolpho	123 First St.	456780
Mauricio	Hernandez	321 Second Ave	123456
Felix	Naumann	Hauptstr. 11	987654
Sal	Stolfo	123 First Street	456789



26



5.5.2004

Felix Naumann, HU Berlin

# Objektidentifikation in XML Daten

Identifikation Integration Optimierung Visualisierung

- Data Warehouse Duplicates
  - Ausnutzung hierarchischer Daten
  - Nur Star-Schema
  - (Ananthkrishna, Chauduri & Ganti 2002)
- XML Duplikate
  - Kinder verschiedener Typen (Snowflake-Schema)
  - Was ist ein Duplikat?
  - Semistrukturierte Daten
  - Schemalose Daten

27



5.5.2004

Felix Naumann, HU Berlin

# Objektidentifikation in XML Daten

Identifikation Integration Optimierung Visualisierung

```
- <author>
  <name>Bernd Amann</name>
- <publication>
  <title>XML Repository and Active Views Demonstration.</title>
  <year>1999</year>
</publication>
- <publication>
  <title>Active Views for Electronic Commerce.</title>
  <year>1999</year>
</publication>
</author>
- <author>
  <name>Sophie Cluet</name>
- <publication>
  <title>XML Repository and Active Views Demonstration.</title>
  <year>1999</year>
</publication>
- <publication>
  <title>Active Views for Electronic Commerce.</title>
  <year>1999</year>
</publication>
- <publication>
  <title>Views in a Large Scale XML Repository.</title>
  <year>2001</year>
</publication>
</author>
```

- Vergleiche <author>
  - Mit Subelementen (<publication>)?
  - Wie tief?
- Vergleiche <publication>
  - Mit parallelen Elementen (<year>)?
  - Schema, oder Daten?
- Kurz: Was ist ein Duplikat?

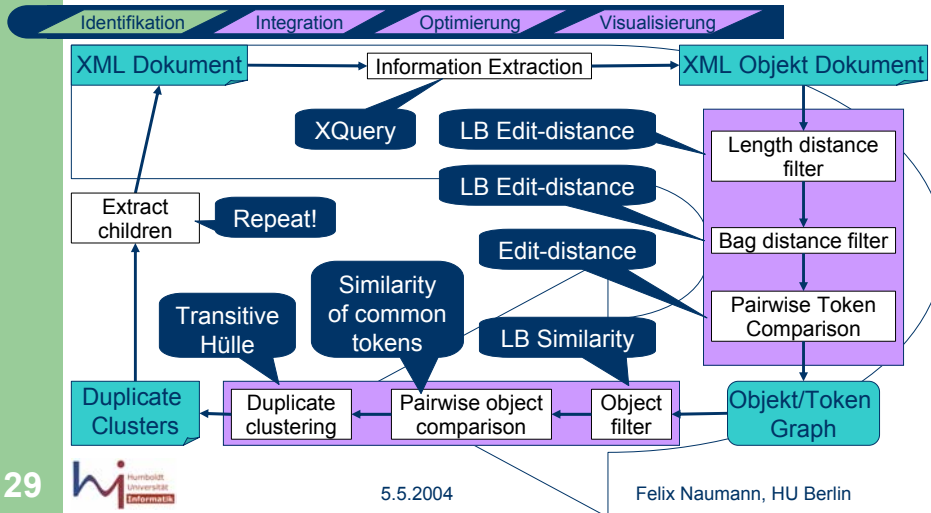
28



5.5.2004

Felix Naumann, HU Berlin

# Objektidentifikation in XML Daten (Melanie Weis)



# Objektidentifikation in XML Daten

The diagram shows two XML snippets with arrows indicating object identification. The left snippet is a list of publications, and the right snippet is a list of authors and publications. The arrows point from the left snippet to the right snippet, indicating the identification of objects.

```

- <author>
  <name>Bernd Amann</name>
- <publication>
  <title>XML Repository and Active Views Demonstration.</title>
  <year>1999</year>
</publication>
- <publication>
  <title>Active Views for Electronic Commerce.</title>
  <year>1999</year>
</publication>
</author>
- <author>
  <name>Sophie Cluet</name>
- <publication>
  <title>XML Repository and Active Views Demonstration.</title>
  <year>1999</year>
</publication>
- <publication>
  <title>Active Views for Electronic Commerce.</title>
  <year>1999</year>
</publication>
- <publication>
  <title>Views in a Large Scale XML Repository.</title>
  <year>2001</year>
</publication>
</author>
- <inproceedings key="conf/vldb/AbiteboulAAACHMMMMSTV9">
  <author>Serge Abiteboul</author>
  <author>Vincent Aguilera</author>
  <author>Sébastien Ailleret</author>
  <author>Bernd Amann</author>
  <author>Sophie Cluet</author>
  <author>Brendan Hills</author>
  <author>Frédéric Hubert</author>
  <title>XML Repository and Active Views Demonstration.</t
  <pages>742-745</pages>
  <year>1999</year>
  <booktitle>VLDB</booktitle>
  <url>db/conf/vldb/vldb99.html#AbiteboulAAACHMMMMM
  <crossref>conf/vldb/99</crossref>
  <ee>db/conf/vldb/AbiteboulAAACHMMMMSTV99.html</e
  <cdrom>VLDB99/P73.pdf</cdrom>
  <cite>conf/edbt/SantosAD94</cite>
  <cite>www/org/w3/dom</cite>
</inproceedings>
  
```

5.5.2004 Felix Naumann, HU Berlin

# Duplikat-gesteuertes Schema Matching (Alexander Bilke)

Identifikation    Integration    Optimierung    Visualisierung

- Umgekehrte Idee
- Schema Matching
- Herkömmliche Lösungen (siehe zuvor)
  - Namen der Schemaelemente (*label-based*)
  - Darunterliegende Daten (*instance-based*)
    - Insbesondere Eigenschaften von Spalten
  - Struktur des Schemas (*structure-based*)
- Nun
  - Finde Duplikate (trotz mangelnder Schemata)
  - Korrespondenzen zwischen gleichen Attributwerten

31



5.5.2004

Felix Naumann, HU Berlin

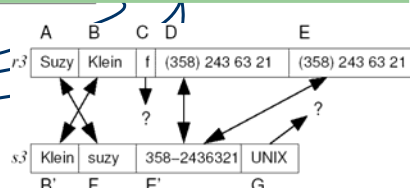
# Duplikat-gesteuertes Schema Matching

Identifikation    Integration    Optimierung    Visualisierung

R	A	B	C	D	E
r <sub>1</sub>	John	Doe	m	(408) 757 33 39	(408) 757 33 38
r <sub>2</sub>	Joe	Smith	m	(249) 361 56 16	(249) 234 23 66
r <sub>3</sub>	Suzy	Klein	f	(358) 243 63 21	(358) 243 63 21
r <sub>4</sub>	Sam	Adams	m	(541) 812 71 00	(541) 812 11 64
r <sub>5</sub>	Mark	Spitz	m	(901) 831 93 11	(901) 861 23 82
r <sub>6</sub>	Jim	Beam	-	(782) 123 89 57	(781) 188 37 44

Jetzt wissen wir, WAS integriert werden soll. Bloß WIE?

S	B'	F	E'	G
s <sub>1</sub>	Doe	jdoe	408-9182043	XP
s <sub>2</sub>	Deen	jdean	369-3663625	XP
s <sub>3</sub>	Klein	suzy	358-2436321	UNIX
s <sub>4</sub>	Adams	adams	541-8121164	W2000
s <sub>5</sub>	Wong	howard	923-6363443	Linux
s <sub>6</sub>	Kurz	itsme	-	UNIX



32



5.5.2004

Felix Naumann, HU Berlin



# Objektintegration

Identifikation   Integration   Optimierung   Visualisierung



0766607194	H. Melville		\$3.98	
------------	-------------	--	--------	--



0766607194	Herman Melville	Moby Dick	\$5.99	
------------	-----------------	-----------	--------	--



33

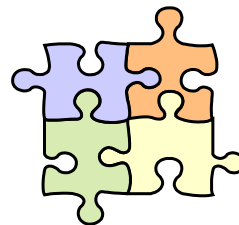
5.5.2004

Felix Naumann, HU Berlin

# Relationale Objektintegration

Identifikation   Integration   Optimierung   Visualisierung

- **Union**  $\cup$ 
  - Duplikat-Eliminierung
- **Outer union**  $\odot$  (Codd 1979)
  - Union bei heterogenen Schemata
- **Minimum union**  $\oplus$  (Ullmann 1989, Galindo-Legaria 1994)
  - Eliminierung subsummierter Tupel
- **Merge union**
  - Duplikatintegration
  - Konfliktlösung

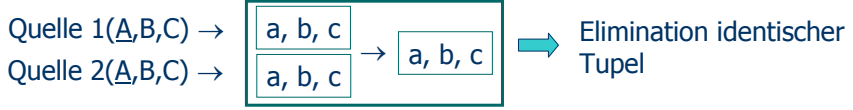


34

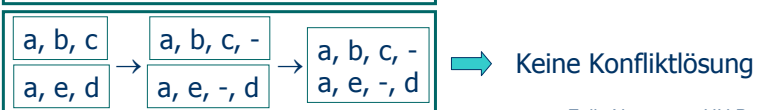
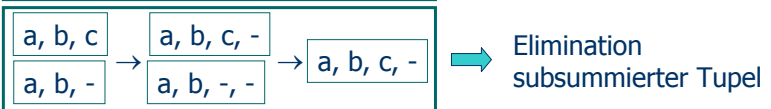
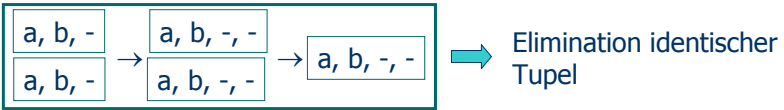
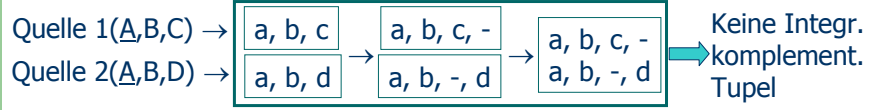
5.5.2004

Felix Naumann, HU Berlin

# Wie Union integriert

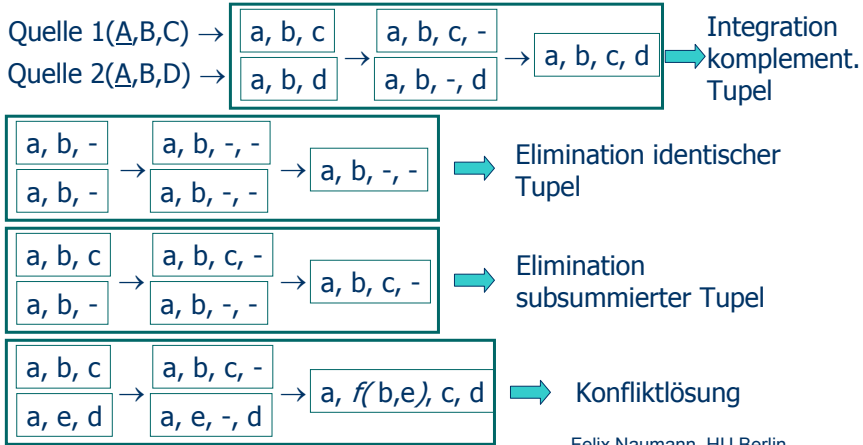


# Wie Minimum Union integriert



# Wie Merge Union integrieren sollte (Jens Bleiholder)

Identifikation   Integration   Optimierung   Visualisierung



37

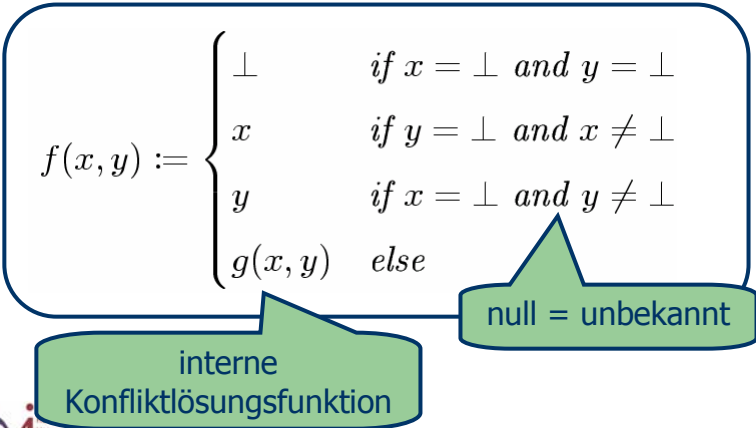
Informatica

5.5.2004

Felix Naumann, HU Berlin

# Konfliktlösung

Identifikation   Integration   Optimierung   Visualisierung



38

Informatica

5.5.2004

Felix Naumann, HU Berlin

# Konfliktlösungsfunktionen

Identifikation   Integration   Optimierung   Visualisierung

- Numerisch: SUM, AVG, MAX, MIN, ...
- Nicht-numerisch:  

Jetzt wissen wir, WIE integriert werden soll.  
Aber WIE GUT schaffen wir das?
- Spezialfunktionen: RANDOM, COUNT, CHOOSE, FAVOR, MaxIQ, ...
- Domänen-spezifisch ...

39

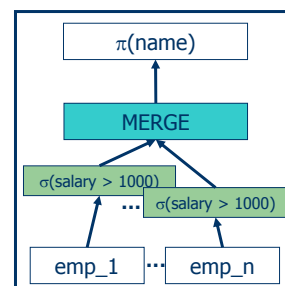
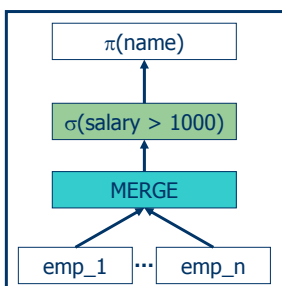


5.5.2004

Felix Naumann, HU Berlin

# Anfrageoptimierung mit Merging

Identifikation   Integration   Optimierung   Visualisierung



- Korrektheit?
- Vollständigkeit?
- Effizienz?

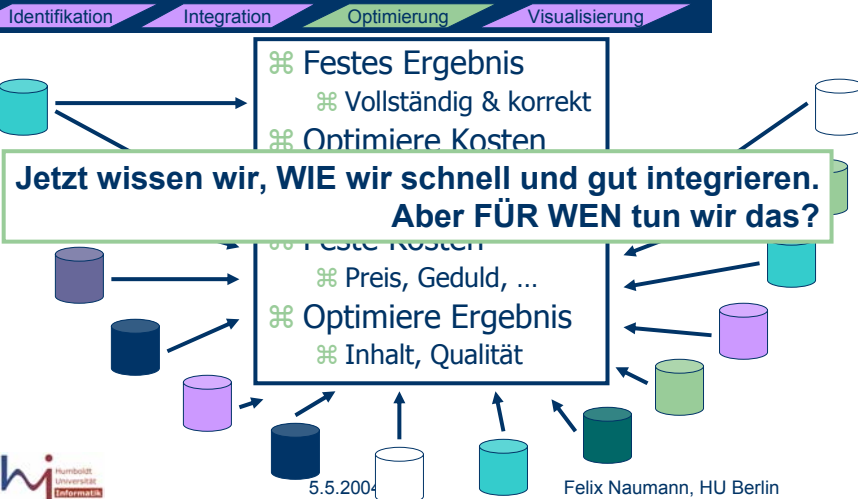
40



5.5.2004

Felix Naumann, HU Berlin

## Optimierung – Ein Paradigmenwechsel



## Visualisierung – Ziele

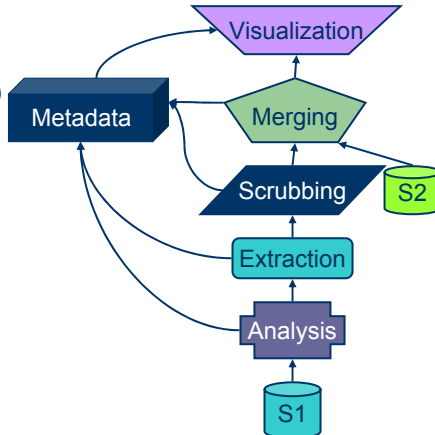
Identifikation Integration **Optimierung** Visualisierung

- Darstellung integrierter Ergebnisse
  - Viele Quellen
  - Viele Operationen / Transformationen
- Nutzerfreundliche Darstellung
- Drill-Down
  - Metadaten
- Kein kleinster gemeinsamer Nenner!

# Metadaten der Integration

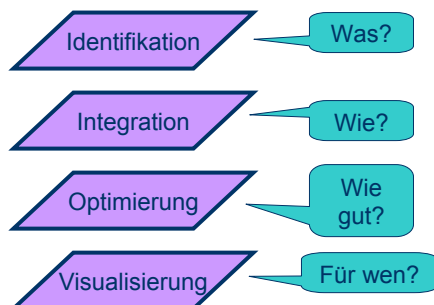
Identifikation   Integration   Optimierung   Visualisierung

- Herkunft
  - Data lineage
    - z.B. (Cui & Widom 2001)
  - Durch Operatoren hindurch
- Transformationen
  - Aggregation / Integration
- Konflikte
- Datenqualität
- Ranking



43

# Informationsintegration - Zusammenfassung



<http://www.informatik.hu-berlin.de/mac/naumann@informatik.hu-berlin.de>

44

## Dank an

- Doktoranden
  - Jens Bleiholder
  - Melanie Weis
  - Alexander Bilke (TU-Berlin)
  - Armin Roth (DaimlerChrysler)
- Studenten
  - Christoph Böhm
  - Karsten Draba



## Literatur

- Fellegi & Sunter 1969
  - A theory of record linkage. Journal of the American Statistical Association, 64: 1183-1210.
- Codd 1979
  - Extending the Relational Database Model to Capture more Meaning. ACM Transactions on Database Systems (TODS), 4(4): 397-434.
- Galindo-Legaria 1994
  - Outerjoins as Disjunctions. ACM SIGMOD Conference, 348-358.
- Hernandez & Stolfo 1998
  - Real-world data is dirty: Data cleansing and the merge/purge problem. Data Mining and Knowledge Discovery, 2(1): 9-37.
- Rahm & Bernstein 2001
  - A survey of approaches to automatic schema matching. VLDB Journal 10(4), 334-350.
- Cui & Widom 2001
  - Lineage tracing for general data warehouse transformations. VLDB Journal 12(1): 41-58.
- Ananthakrishna, Chauduri & Ganti 2002
  - Eliminating Fuzzy Duplicates in Data Warehouses. VLDB Conference: 586-597.
- Erhard Rahm and Philip Bernstein 2001
  - A survey of approaches to automatic schema matching, VLDB Journal 10(4), 2001.
- Ron Fagin, Mauricio Hernandez, Lucian Popa, Renee Miller, and Yannis Velegarakis,
  - Translating Web Data, VLDB 2002, Hong Kong, China.