Hasso Plattner Institut

IT Systems Engineering | Universität Potsdam

Data Consolidation
in Three Steps

Milano, May 9, 2008

Felix Naumann

The HPI – Hasso Plattner Institut

- Founded in 1998 as a Public Private Partnership
- Hasso Plattner, co-founder of SAP, endowed over 200 Mio. Euro.
- Adjoined with the University of Potsdam
  - Capital of Brandenburg, bordering Berlin
- 400 students – Bachelor, Master, and PhD

The Information Systems Department

Paul **Führing**

project **ViQTOR**

Patricia **Hobro**

Prof. Felix **Naumann**

Jens **Bleiholder**

project **fusem**

Karsten **Draba**

Data Fusion

DQ Assessment

Information Integration

project **HumMer**

Information Quality

Duplicate Detection

Melanie **Weis** & Sascha **Szott**

Data Cleaning

Armin **Roth**

Peer Data Management Systems

Matching

Data Integration for Life Science Data Sources

project **Aladin**

Service-Oriented Systems

project **System P**

Mohammed **AbuJarour**

Ontologies

Frank **Kaufer**

Jana **Bauckmann**

Alexander **Albrecht**

project **XClean**

ETL Process Management

Data Profiling for Schema Management

Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
- Summary

Data Fusion in Three Steps:
Resolving Inconsistencies at Schema-, Tuple-, and Value-level

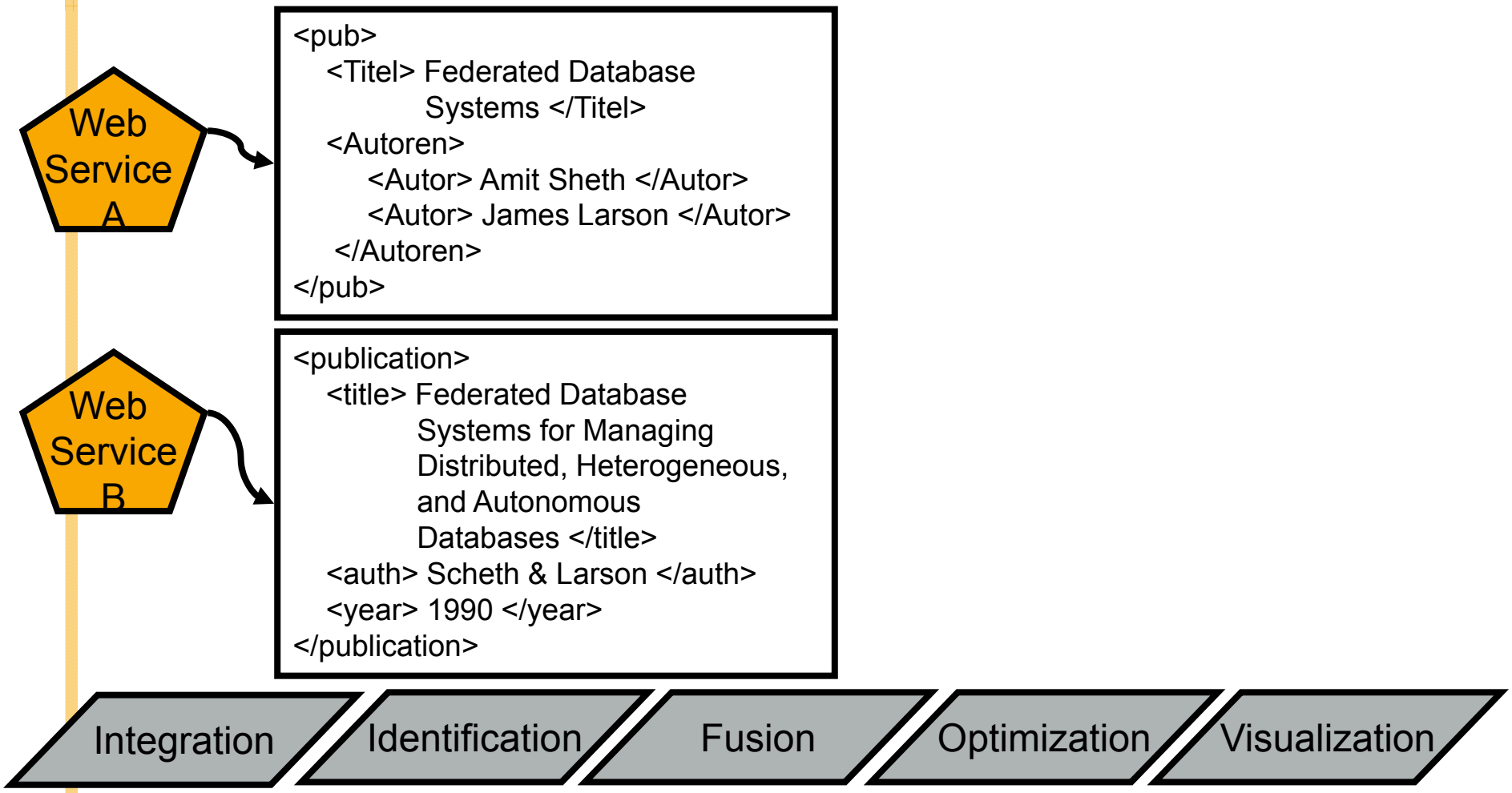
Felix Naumann[1] Alexander Bilke[2] Jens Bleiholder[1] Melanie Weis[1]
[1] Humboldt-Universität zu Berlin
Unter den Linden 6, 10099 Berlin, Germany
{naumann|bleiho mweis}@informatik.hu-berlin.de
[2] Technische Universität Berlin
Strasse des 17. Juni 135, 10623 Berlin, Germany
bilke@cs.tu-berlin.de

Data Fusion

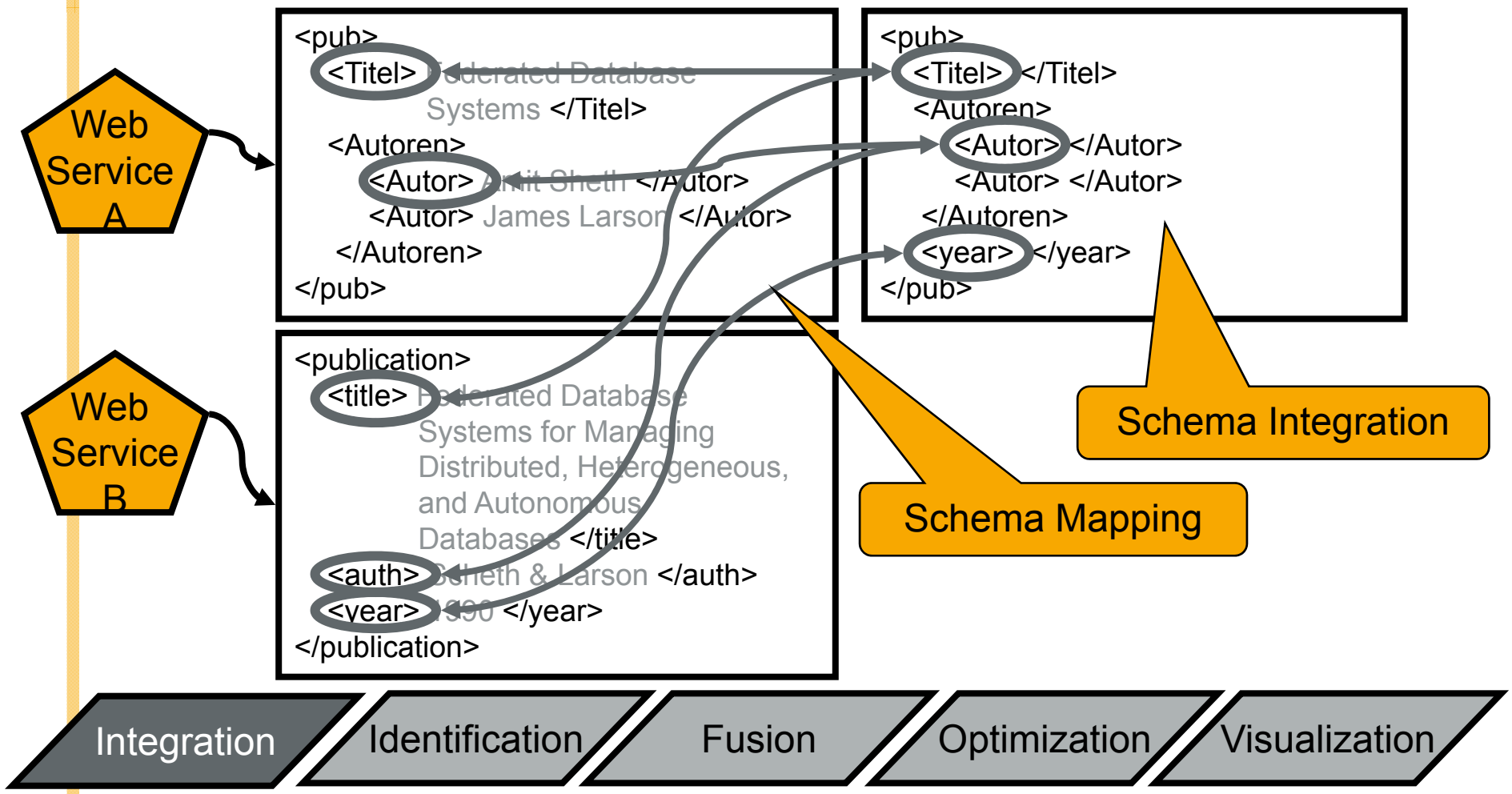
JENS BLEIHOLDER and FELIX NAUMANN
Hasso-Plattner-Institut, Potsdam, Germany

acm Computing Surveys

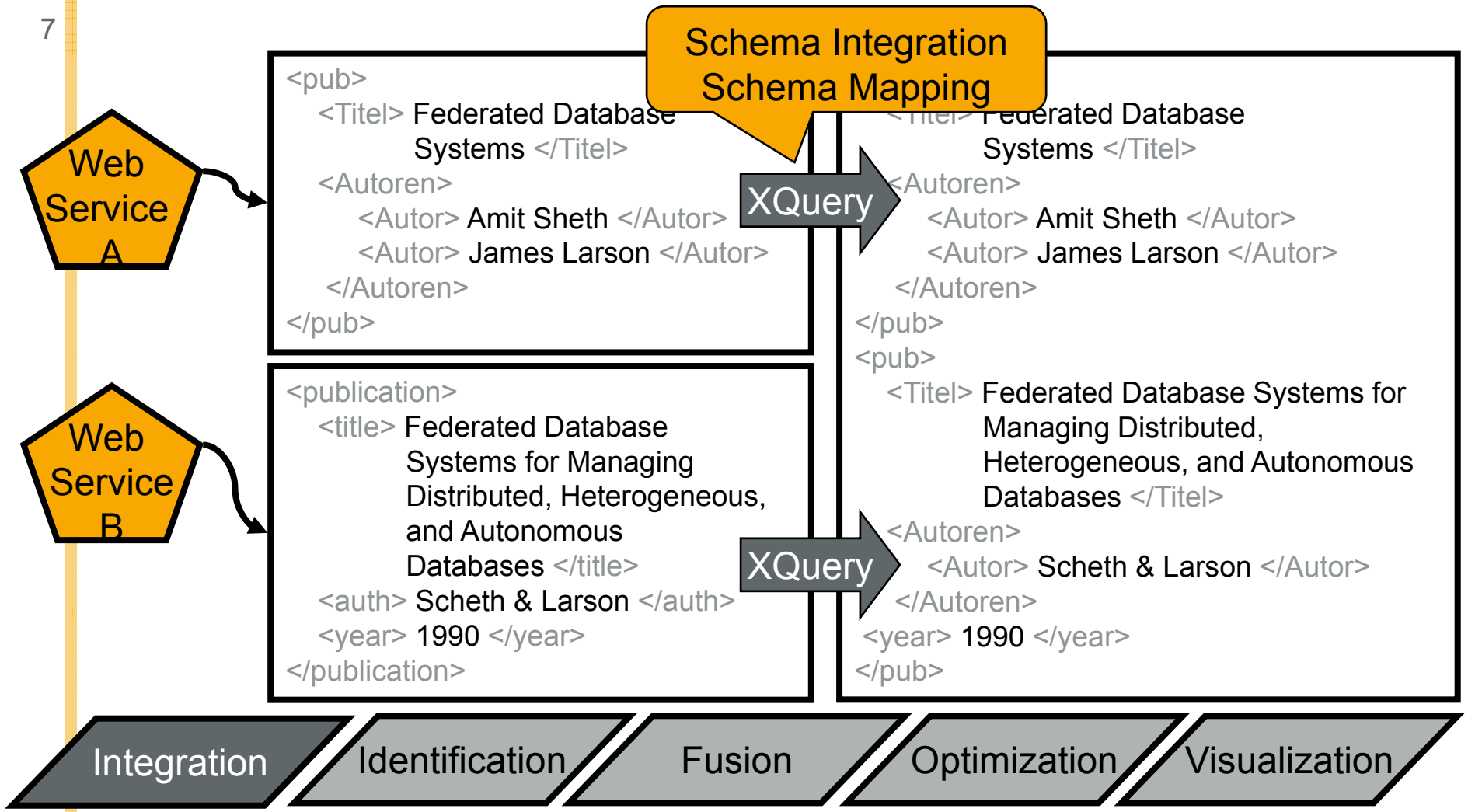
Information Integration



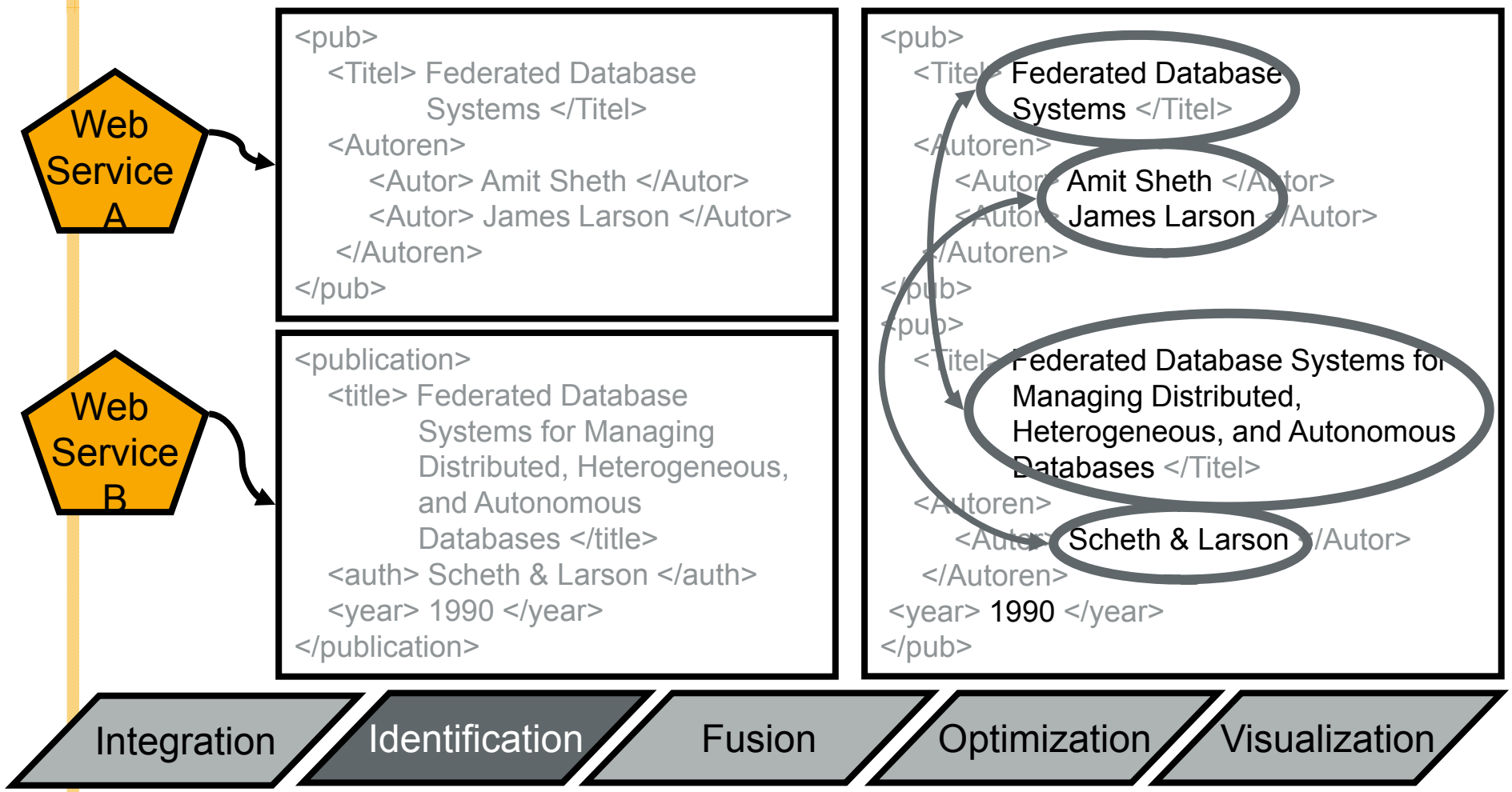
Information Integration



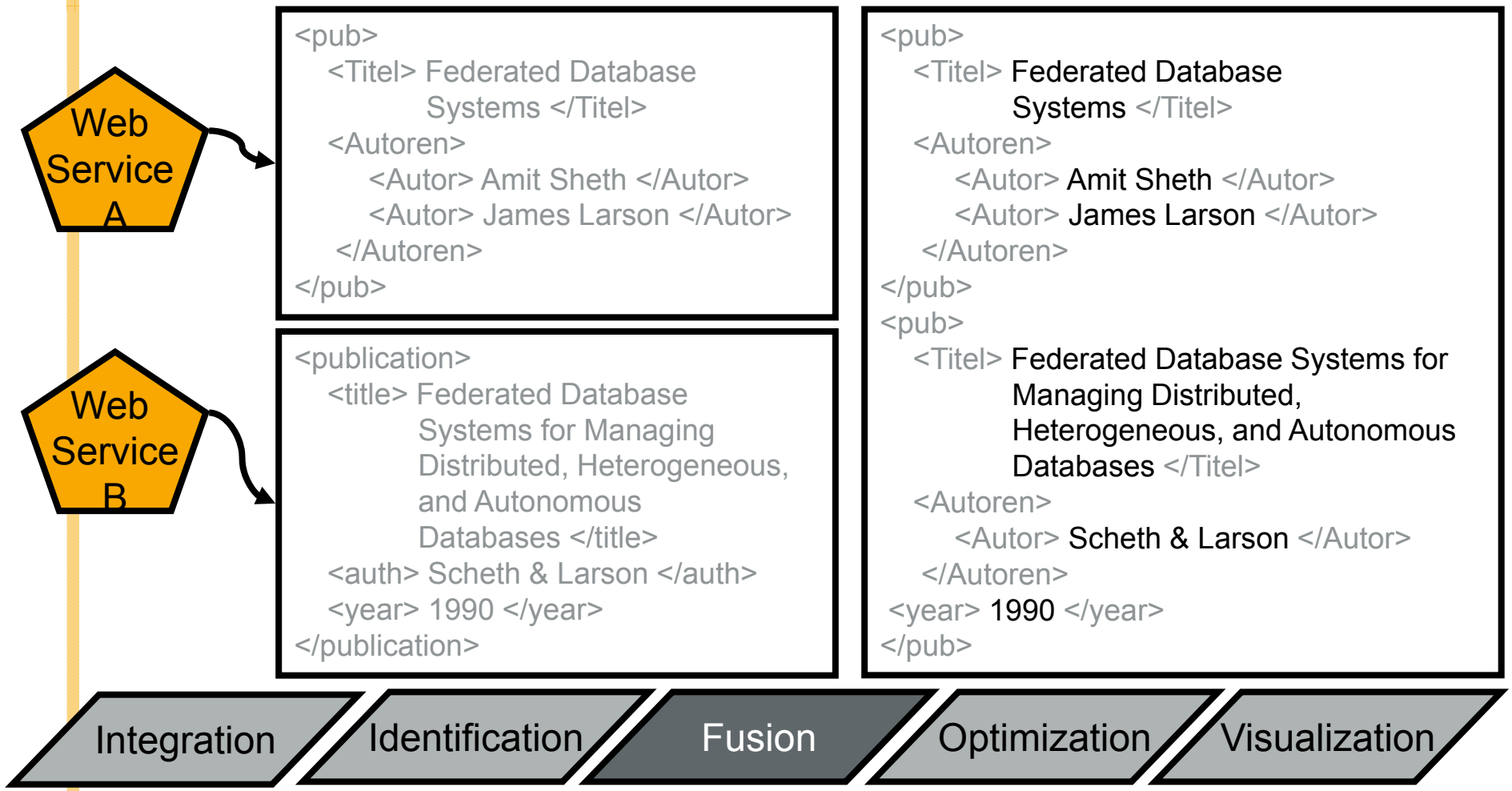
Information Integration



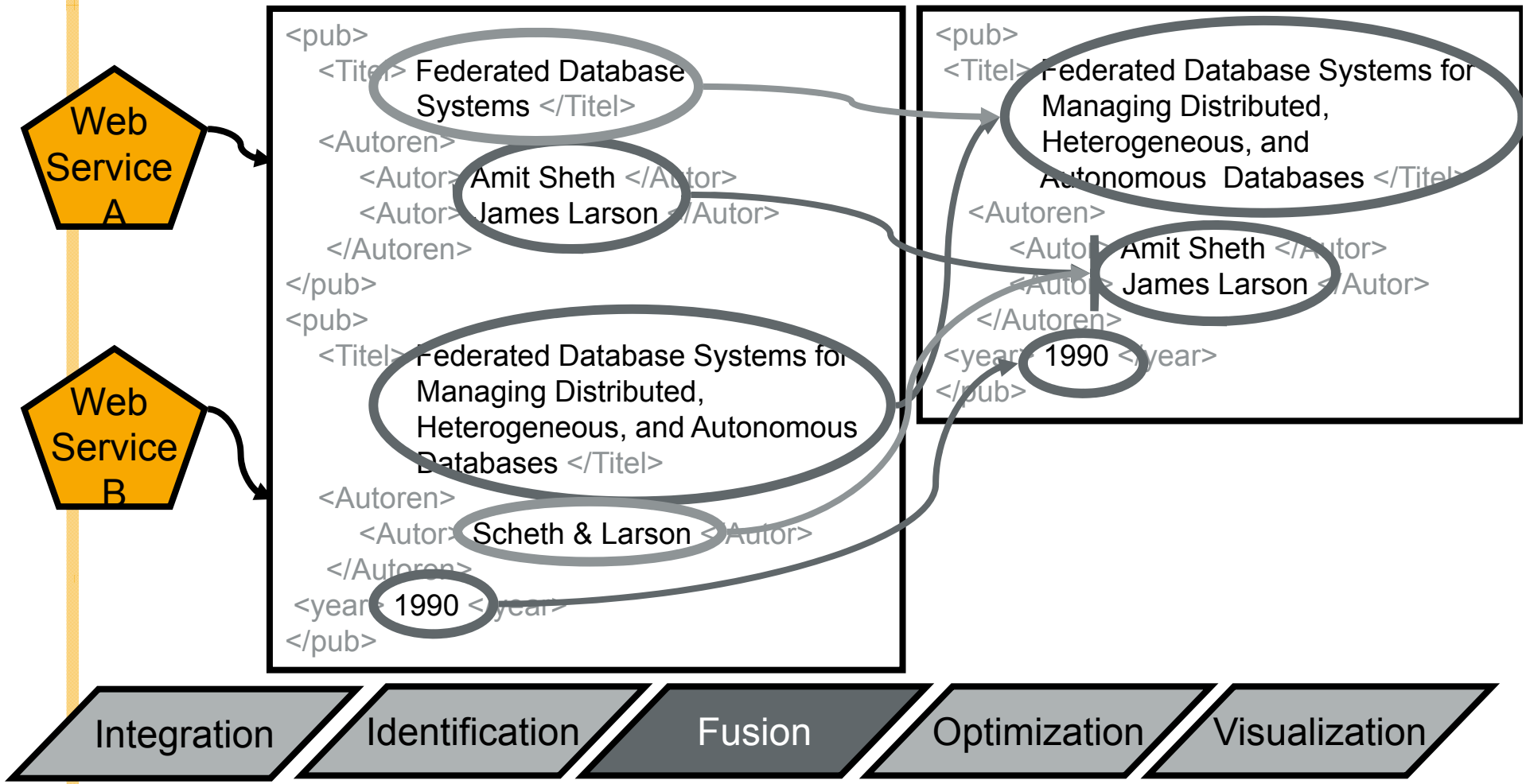
Information Integration



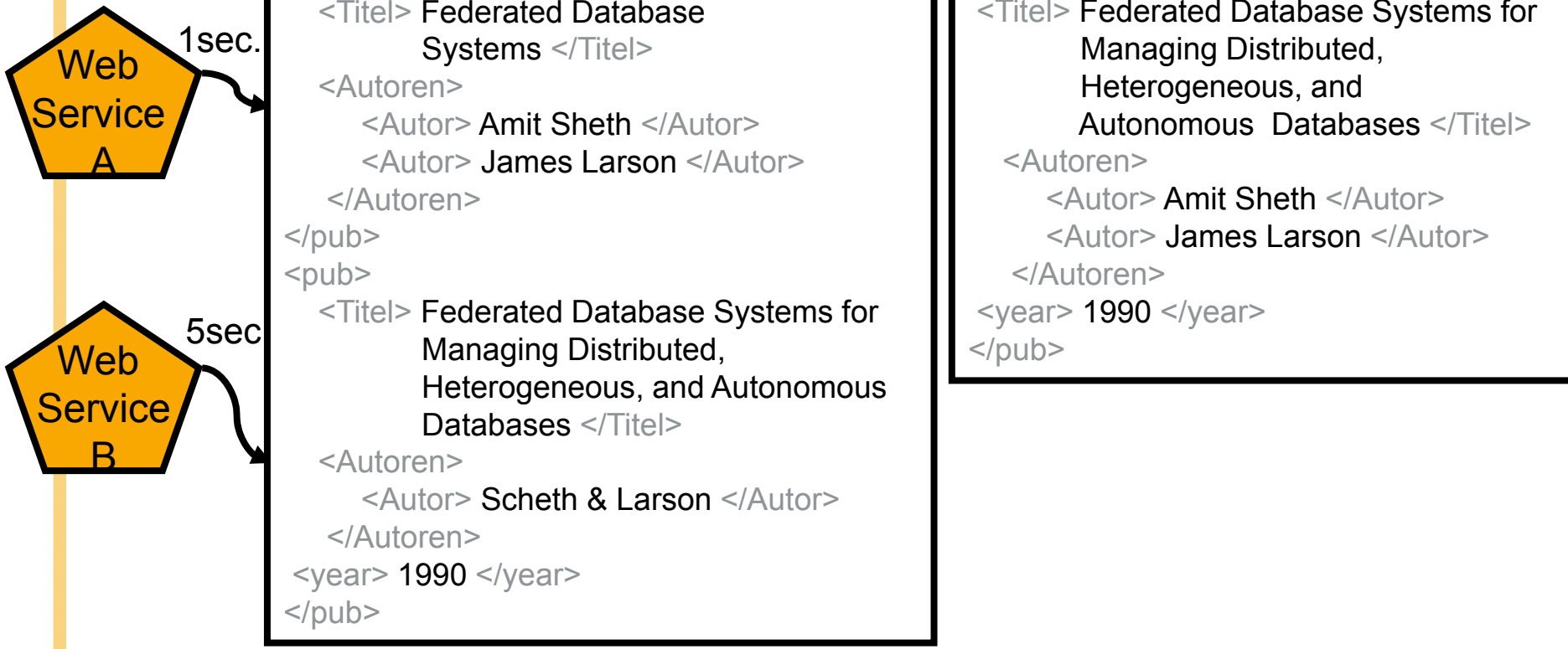
Information Integration



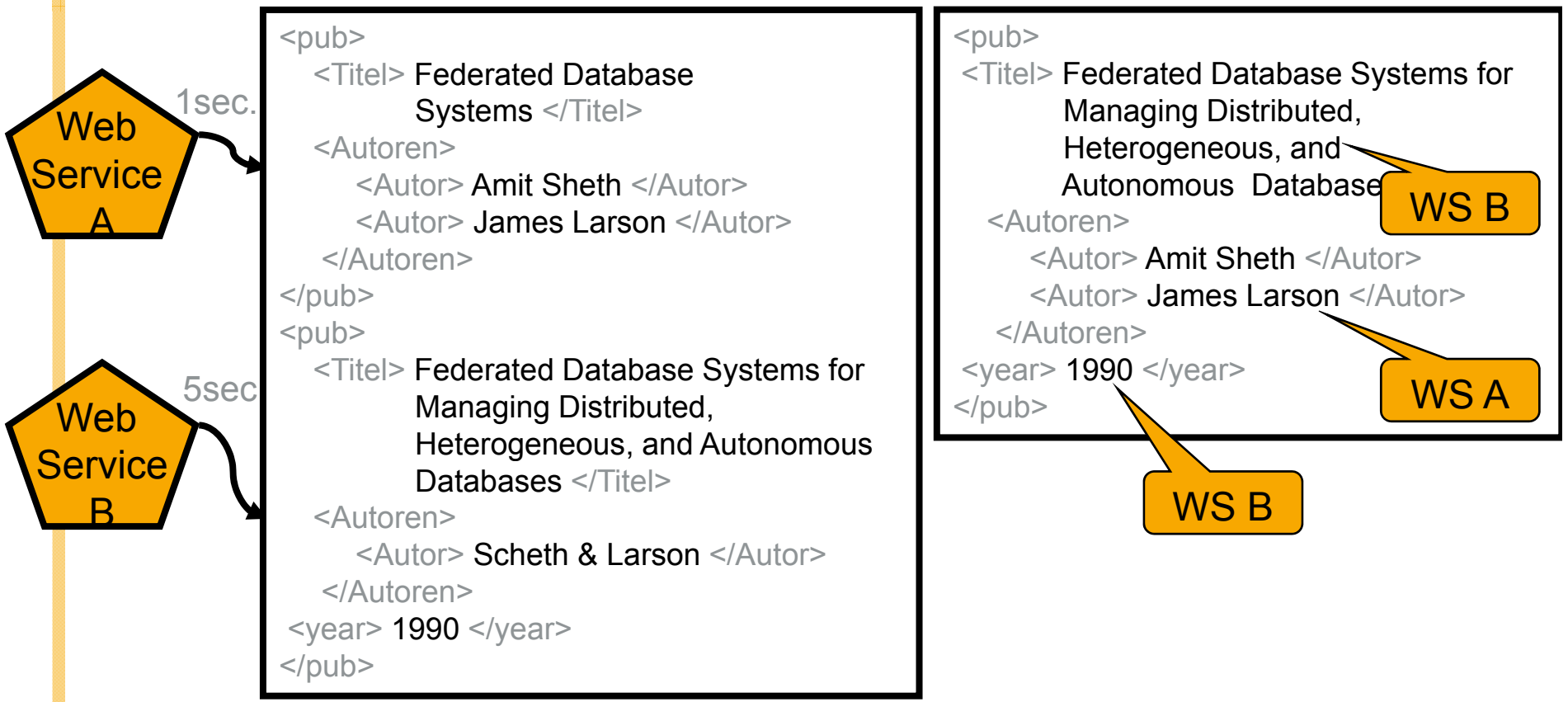
Information Integration



Information Integration



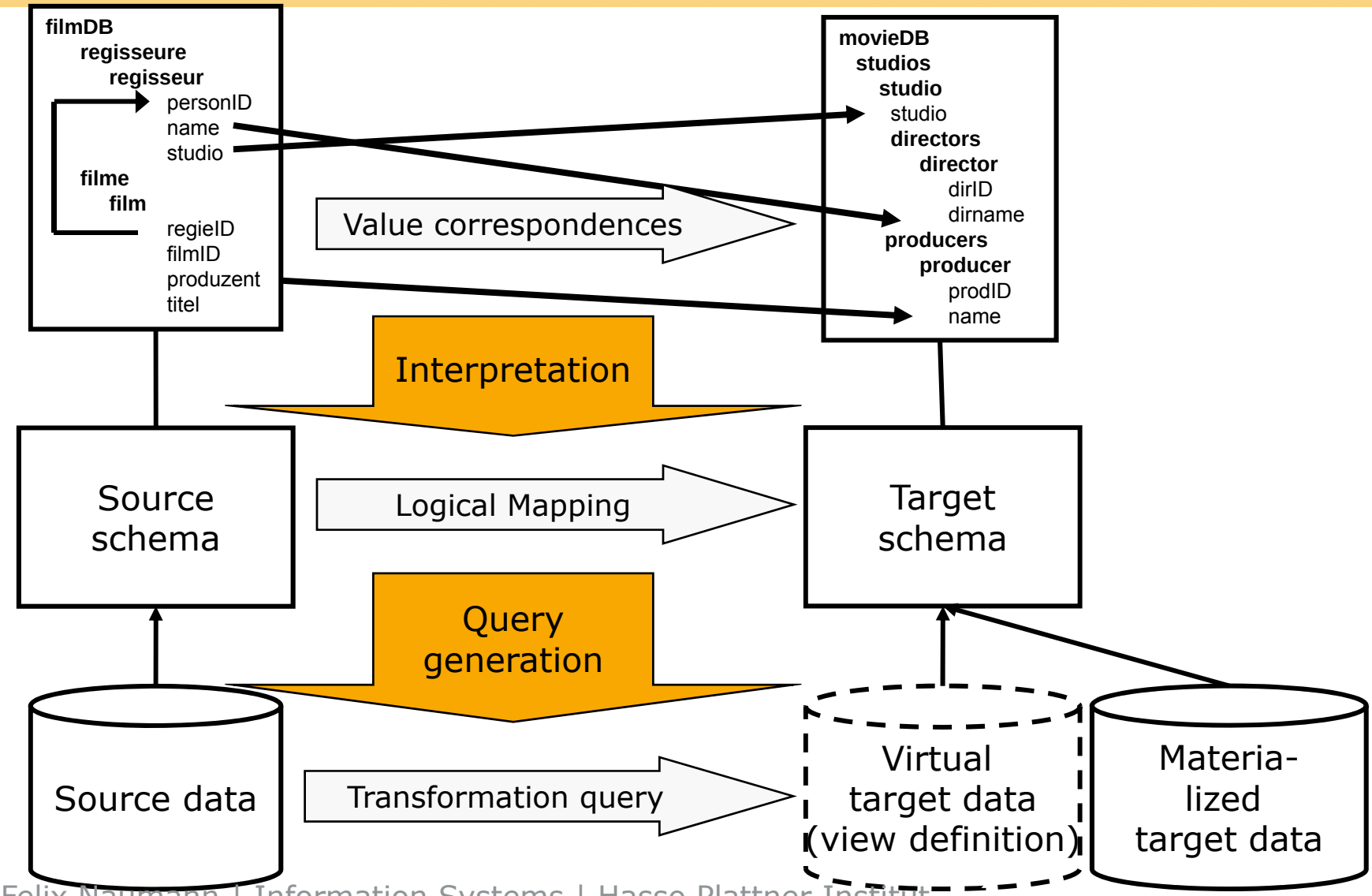
Information Integration



Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
- Summary

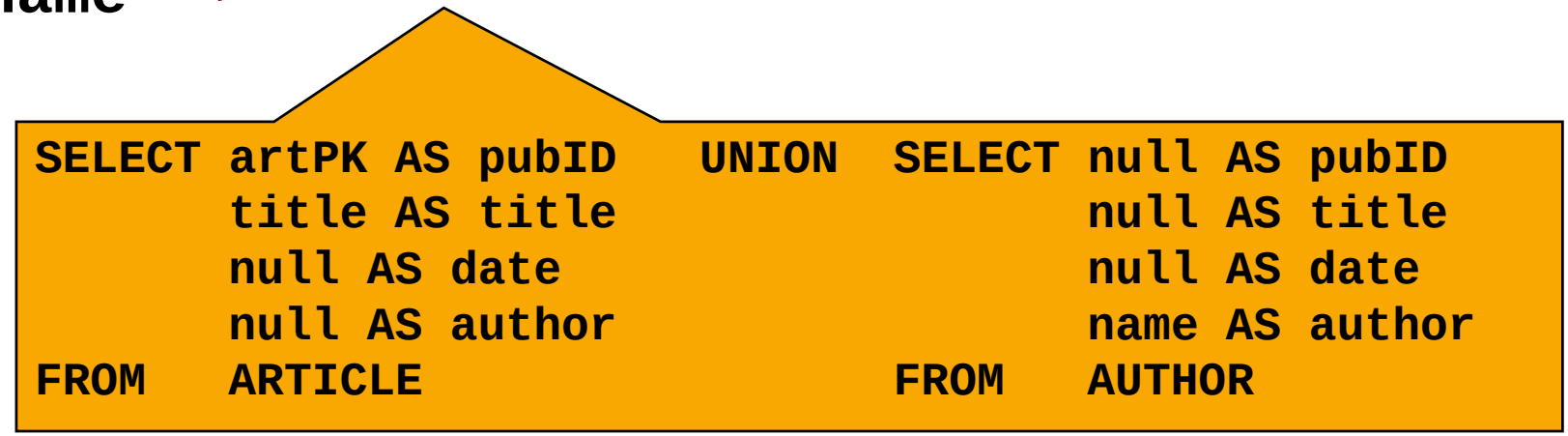
Schema Mapping in Context



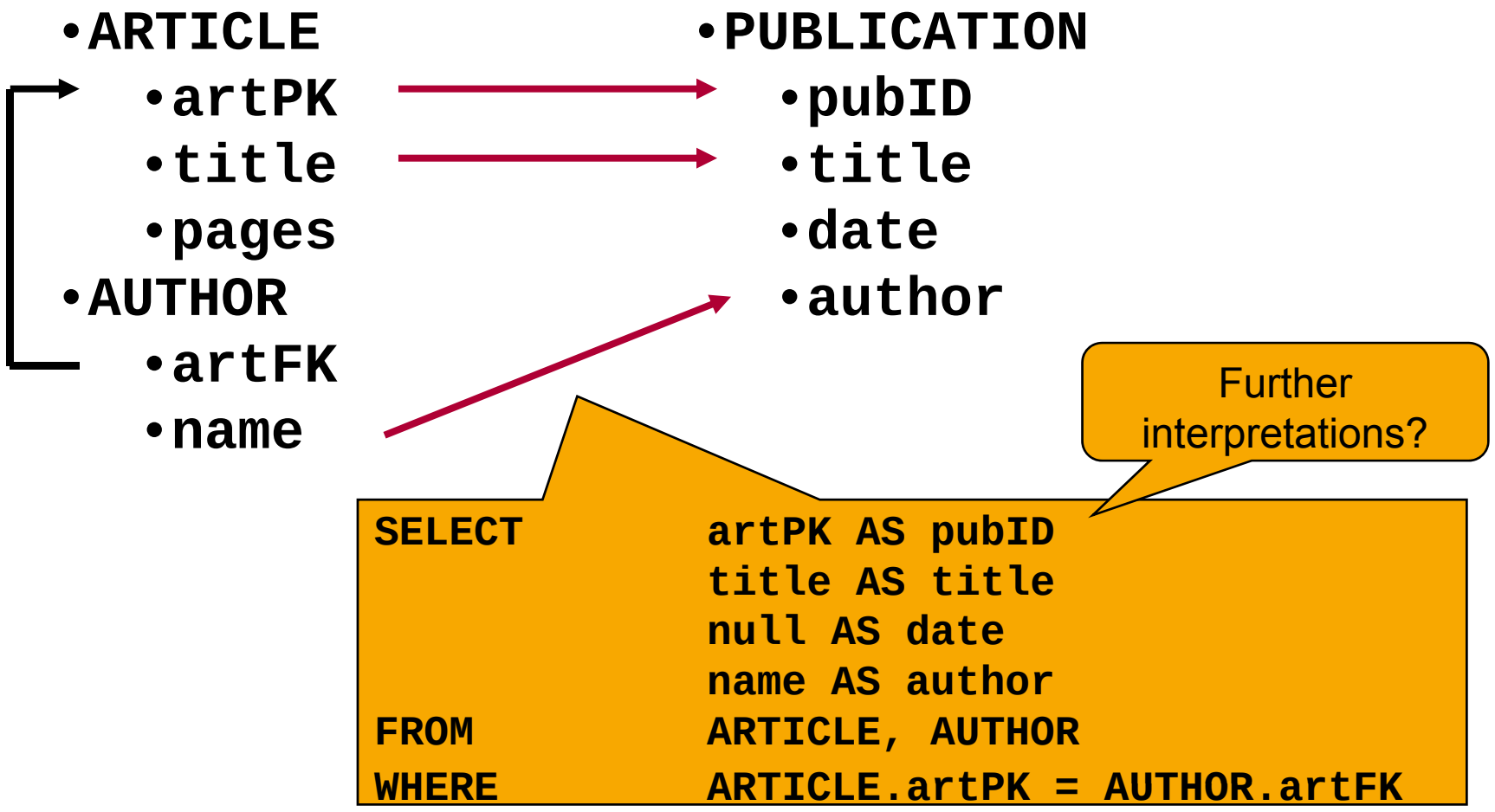
Schema Mapping Example

- **ARTICLE**
  - artPK
  - title
  - pages
- **AUTHOR**
  - artFK
  - name

- **PUBLICATION**
  - pubID
  - title
  - date
  - author



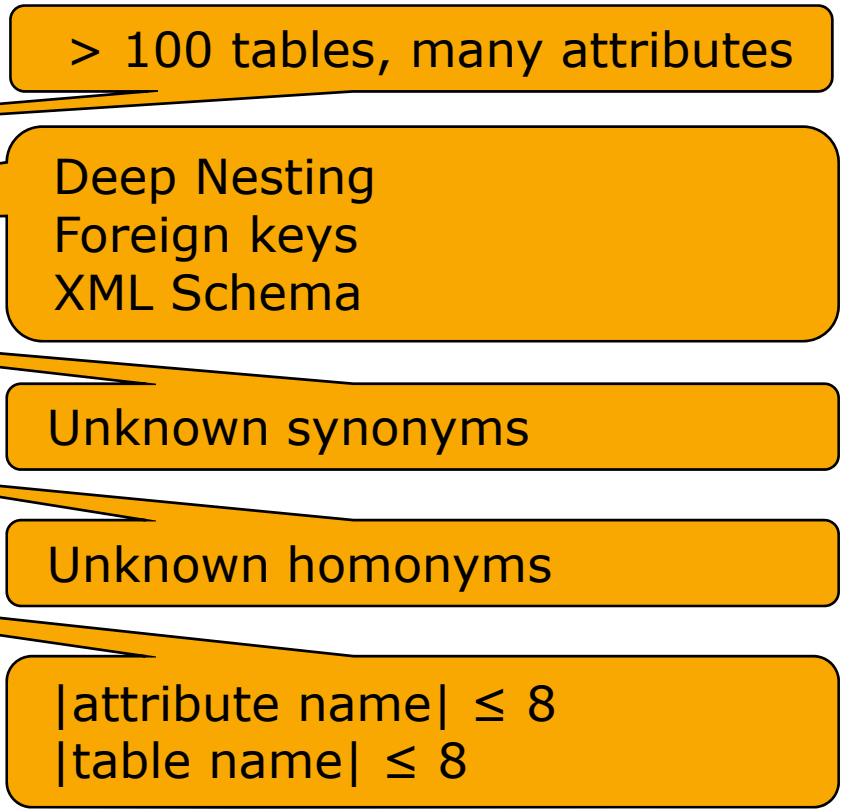
Schematic heterogeneity – solutions

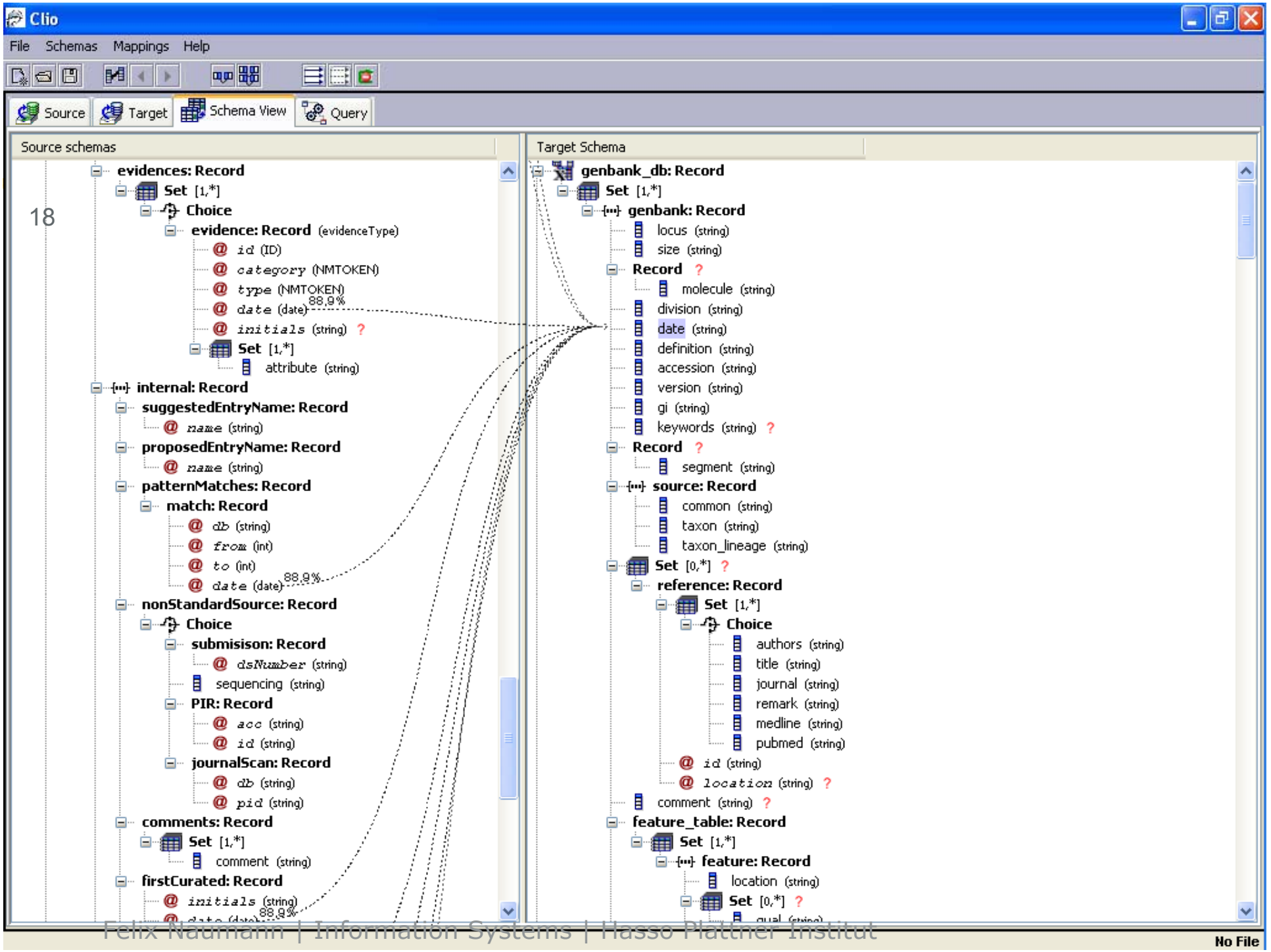


Schema Matching – Motivation

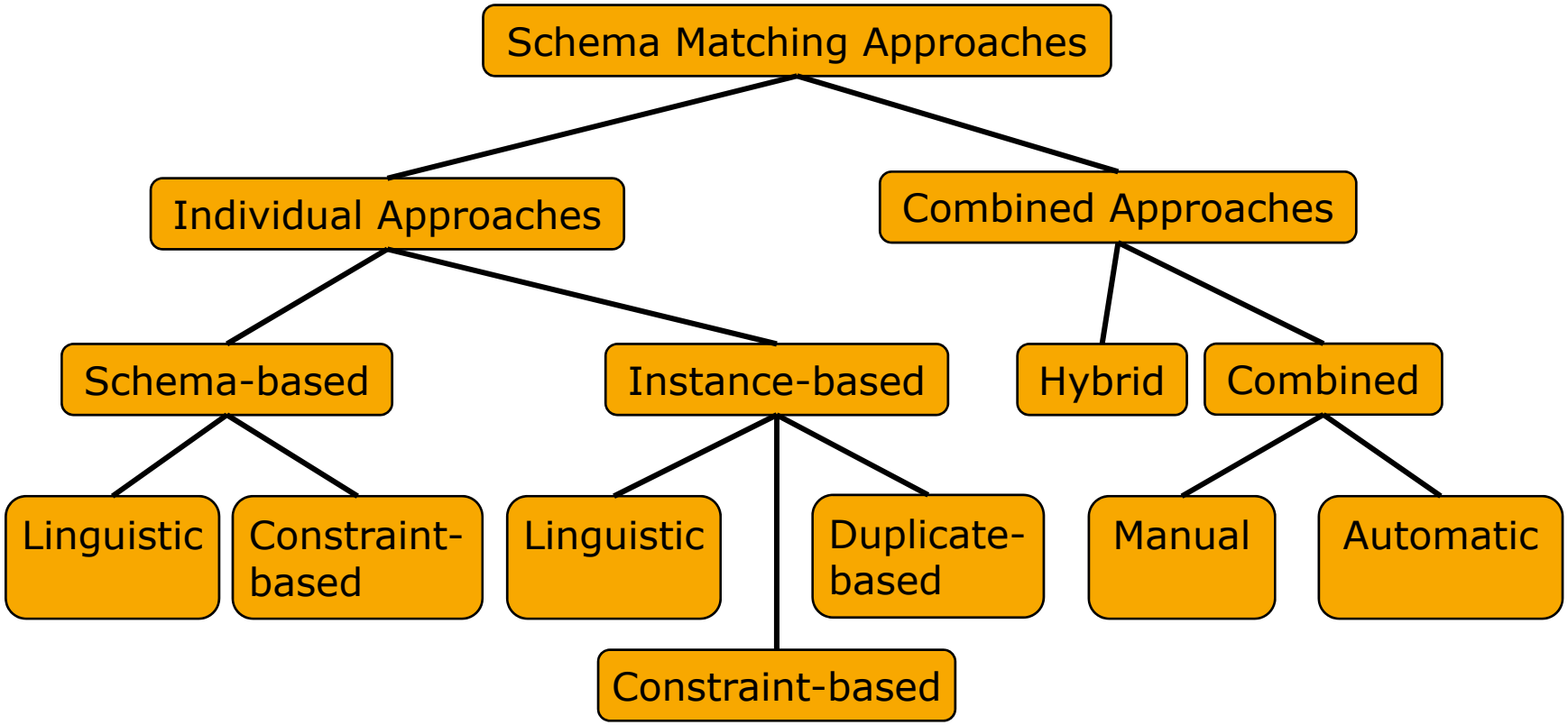
Schemata are

- large
- complex
- foreign
- confusing
- different language
- cryptic




Clio
File Schemas Mappings Help
Source Target Schema View Query
Source schemas
evidences: Record
Set [1,*]
Choice
evidence: Record (evidenceType)
id (ID)
category (NMTOKEN)
type (NMTOKEN)
date (date)
initials (string) ?
Set [1,*]
attribute (string)
internal: Record
suggestedEntryName: Record
name (string)
proposedEntryName: Record
name (string)
patternMatches: Record
match: Record
db (string)
from (int)
to (int)
date (date)
nonStandardSource: Record
Choice
submisison: Record
dsNumber (string)
sequencing (string)
PIR: Record
acc (string)
id (string)
journalScan: Record
db (string)
pid (string)
comments: Record
Set [1,*]
comment (string)
firstCurated: Record
initials (string)
88,9%
Target Schema
genbank_db: Record
Set [1,*]
genbank: Record
locus (string)
size (string)
Record ?
molecule (string)
division (string)
date (string)
definition (string)
accession (string)
version (string)
gi (string)
keywords (string) ?
Record ?
segment (string)
source: Record
common (string)
taxon (string)
taxon_lineage (string)
Set [0,*] ?
reference: Record
Set [1,*]
Choice
authors (string)
title (string)
journal (string)
remark (string)
medline (string)
pubmed (string)
id (string)
location (string) ?
comment (string) ?
feature_table: Record
Set [1,*]
feature: Record
location (string)
Set [0,*] ?
No File

Schema Matching Classification [RB01]



Skip

Instance-based Schema Matching

Instance-based Schema Matching:

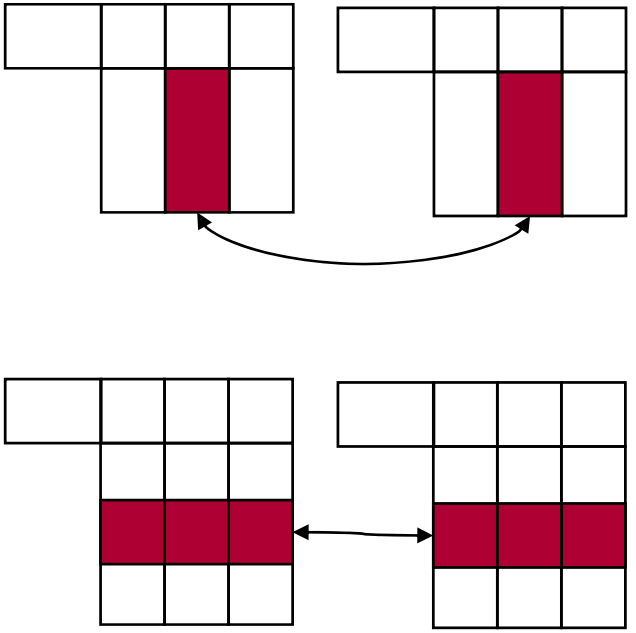
- Correspondences based on similar data values or their properties

Conventional solution: Vertical

- Comparison of columns
- = Attribute classification

Our solution: Horizontal

- Comparison of rows
- = Duplicate detection (despite missing attribute correspondences)



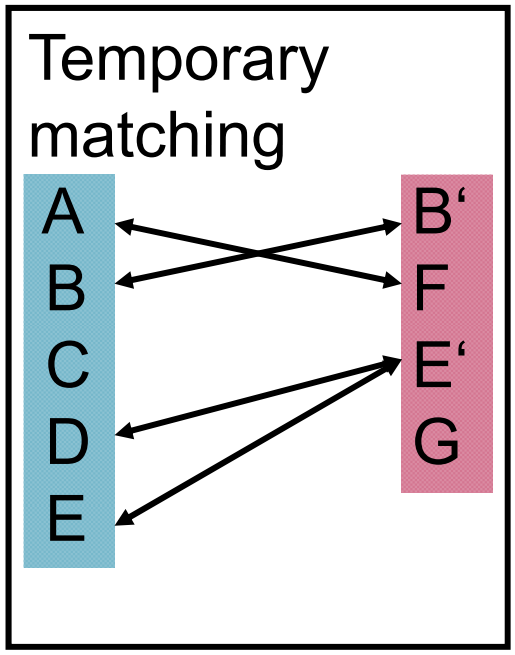
DUMAS Matcher

A	B	C	D	E
Max	Michel	m	601- 4839204	601- 4839204
...	...	...	...	...

?

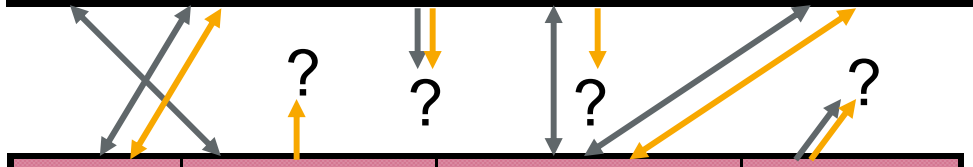
?

B‘	F	E‘	G
Michel	maxm	601- 4839204	UNIX
...	...	...	...

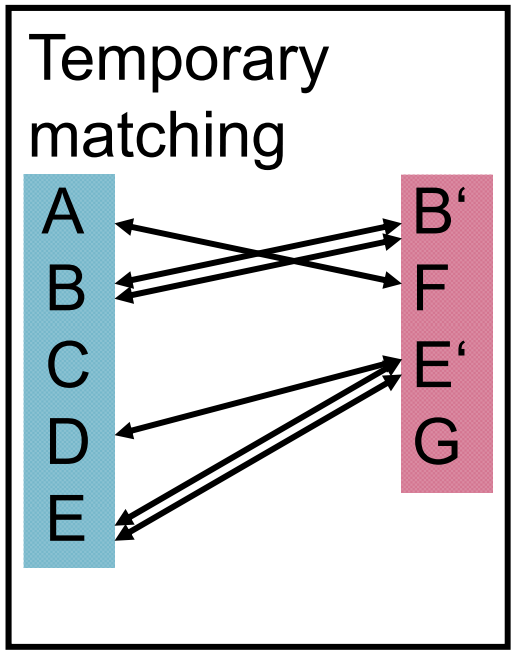


DUMAS Matcher

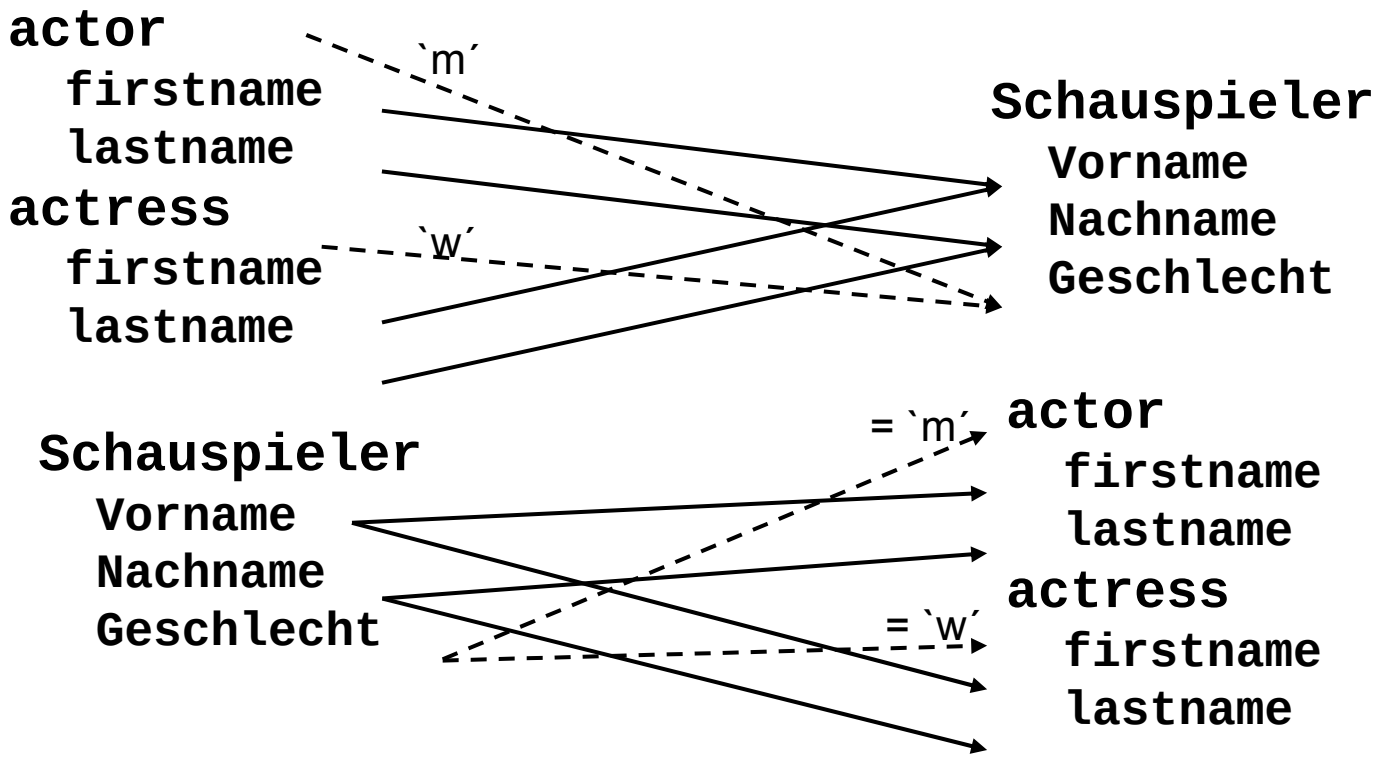
A	B	C	D	E
Max	Michel	m	601- 4839204	601- 4839204
Sam	Adams	m	541- 8127100	541- 8121164



B‘	F	E‘	G
Michel	maxm	601- 4839204	UNIX
Adams	beer	541- 8127164	WinXP



Schema Matching – High-level Matching



Schema Matching – Extensions

n:1 und 1:n matches

- Many combinations
- Many functions
- Parsing
- iMap

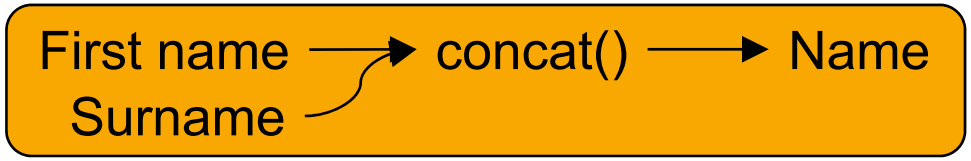
Matching in complex schemata

- Find mapping, not only correspondences
- Unions and joins

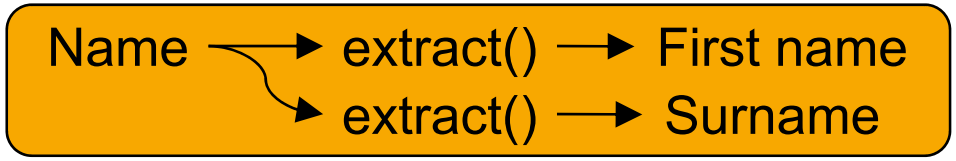
Global Matching

- Match Table and Schema, not just Attributes

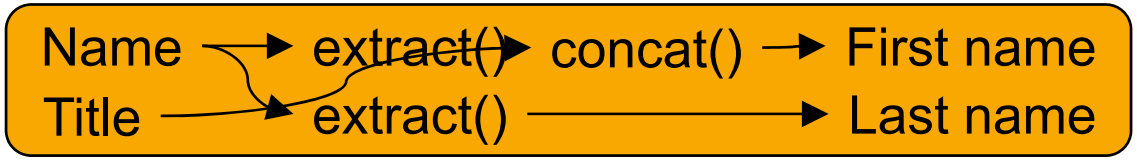
n:1 Matching

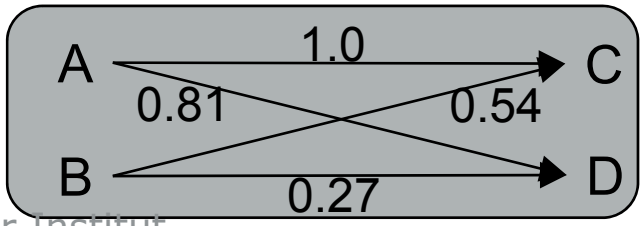


1:n Matching



m:n matching





Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
- Summary

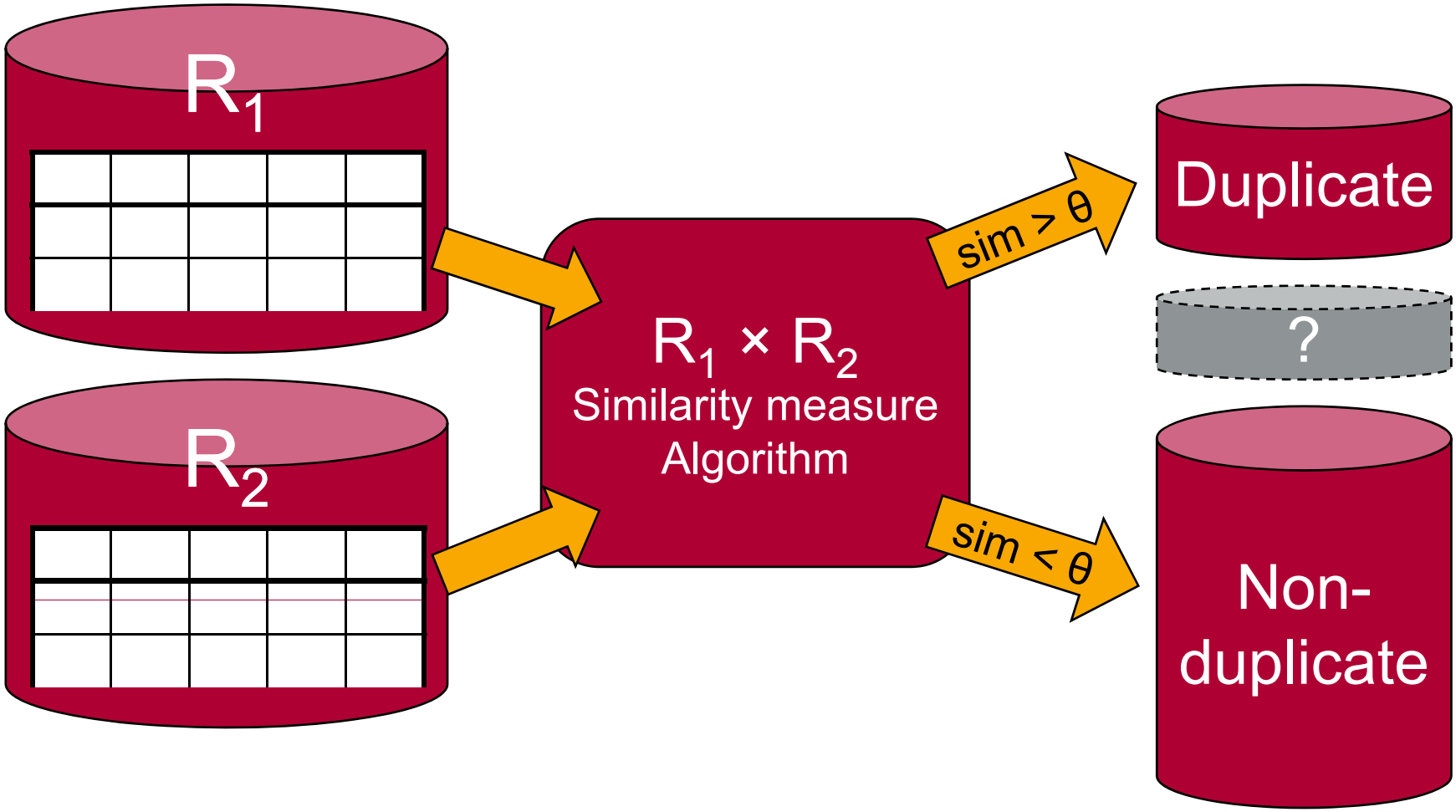
Duplicate Detection

Duplicate detection is the discovery of multiple representations of the same real-world object.

- Problem 1: Representations are not identical.
  - *Fuzzy duplicates*
- Solution: Similarity measures
  - Value- and record-comparisons
  - Domain-dependent or domain-independent

- Problem 2: Data sets are large.
  - Quadratic complexity: Comparison of every pair of records.
- Solution: Algorithms
  - E.g., avoid comparisons by partitioning.

Duplicate Detection



Motivation

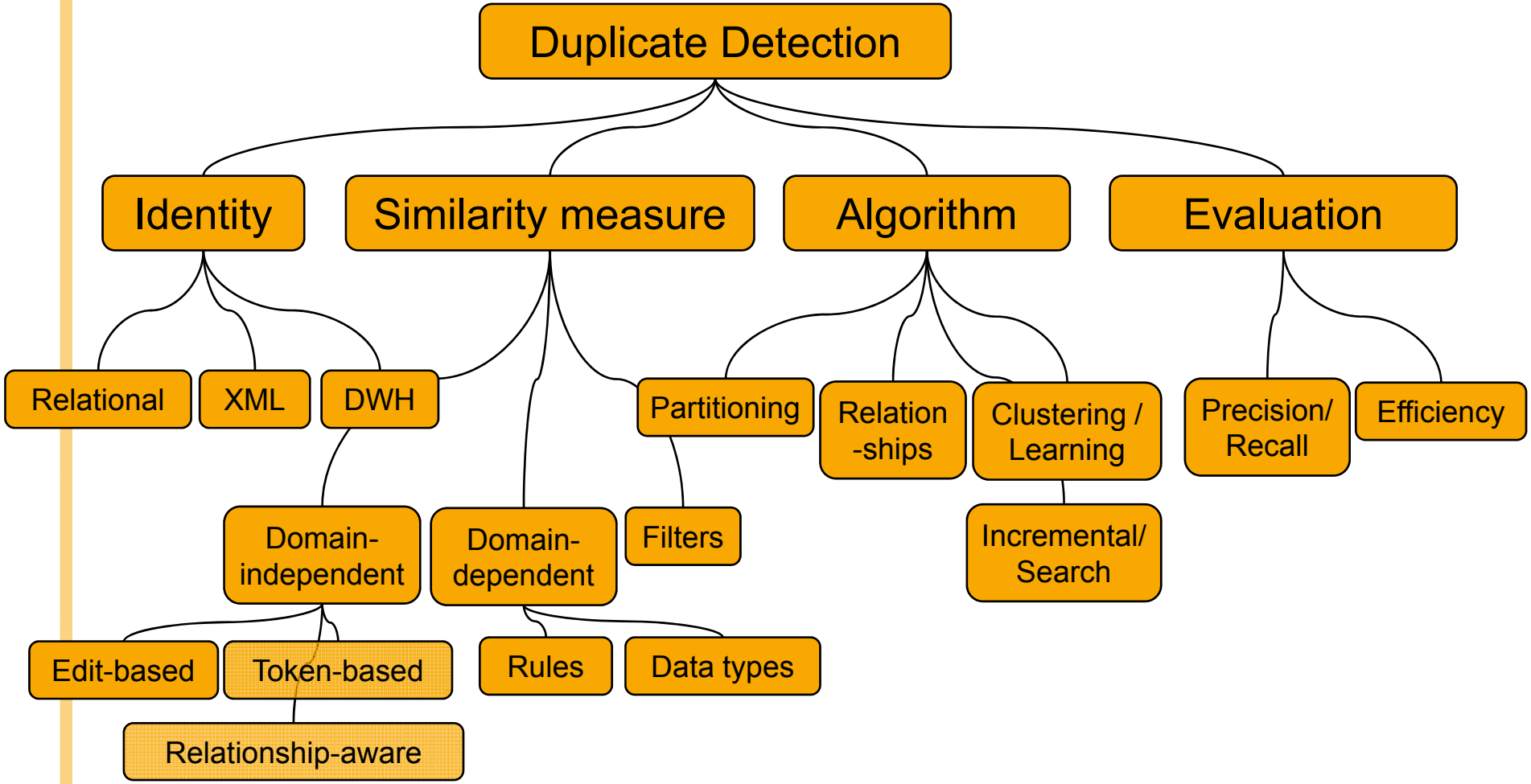
- Possible effects
  - Example: Portfolio Management Offers
  - Credit maximum not detected
  - Too low inventory levels
  - No quantity discount for multiple orders
  - Total revenue of preferred customers unknown
  - Multiple mailings of same catalog to same household
- General problems
  - Additional, unnecessary IT expenses
  - Low customer satisfaction
  - Potentials and dangers not detected
  - Poor quality financial data

Customer	Revenue
BMW	20.000
BaMoWe	5.000.000
Bayerische Motorenwerke	300.000
…	…

Ironically, “Duplicate Detection” has many Duplicates

- Household matching
- Doubles
- Duplicate detection
- Mixed and split citation problem
- Record linkage
- Object identification
- Match
- Deduplication
- Fuzzy match
- Object consolidation
- Entity resolution
- Entity clustering
- Approximate match
- Identity uncertainty
- Reference reconciliation
- Merge/purge
- Hardening soft databases
- Householding
- Reference matching

Duplicate Detection – Research

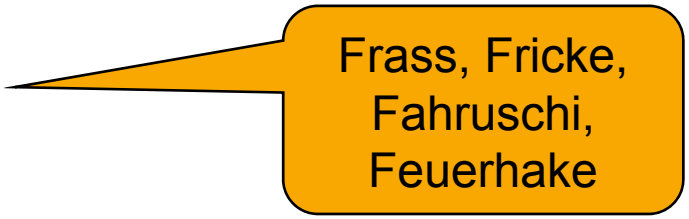


Token-based Similarity Measures

- Tokens
  - Words / Terms
  - n-grams
- Jaccard
  - |{common tokens}| / |{all tokens}|
- TFIDF [Cohen et al. 2003]
  - Term frequency: *tf*
  - Inverse document frequency: *idf*
  - TFIDF: log (*tf*+1) x log (*idf*)
  - Common words have low weight
  - Similarity measure: Cosine similarity of term vectors weighted by TFID
- And many more
  [Koudas Srivastavasa 2005]

Edit-based Similarity Measures

- Jaro [Jaro 1989] / Jaro-Winkler [Winkler 1999]
  - Common letters within ½ string length
  - Transposed letters
- Edit-distance / Levenshtein-distance [Levenshtein 1965]
  - Minimum number of edits from one word to the other
  - Domain-specific costing
  - Dynamic Programming
- Soundex
  - 4-letter code for each word
  - SOUNDEX('Farwick ') = F620
- …

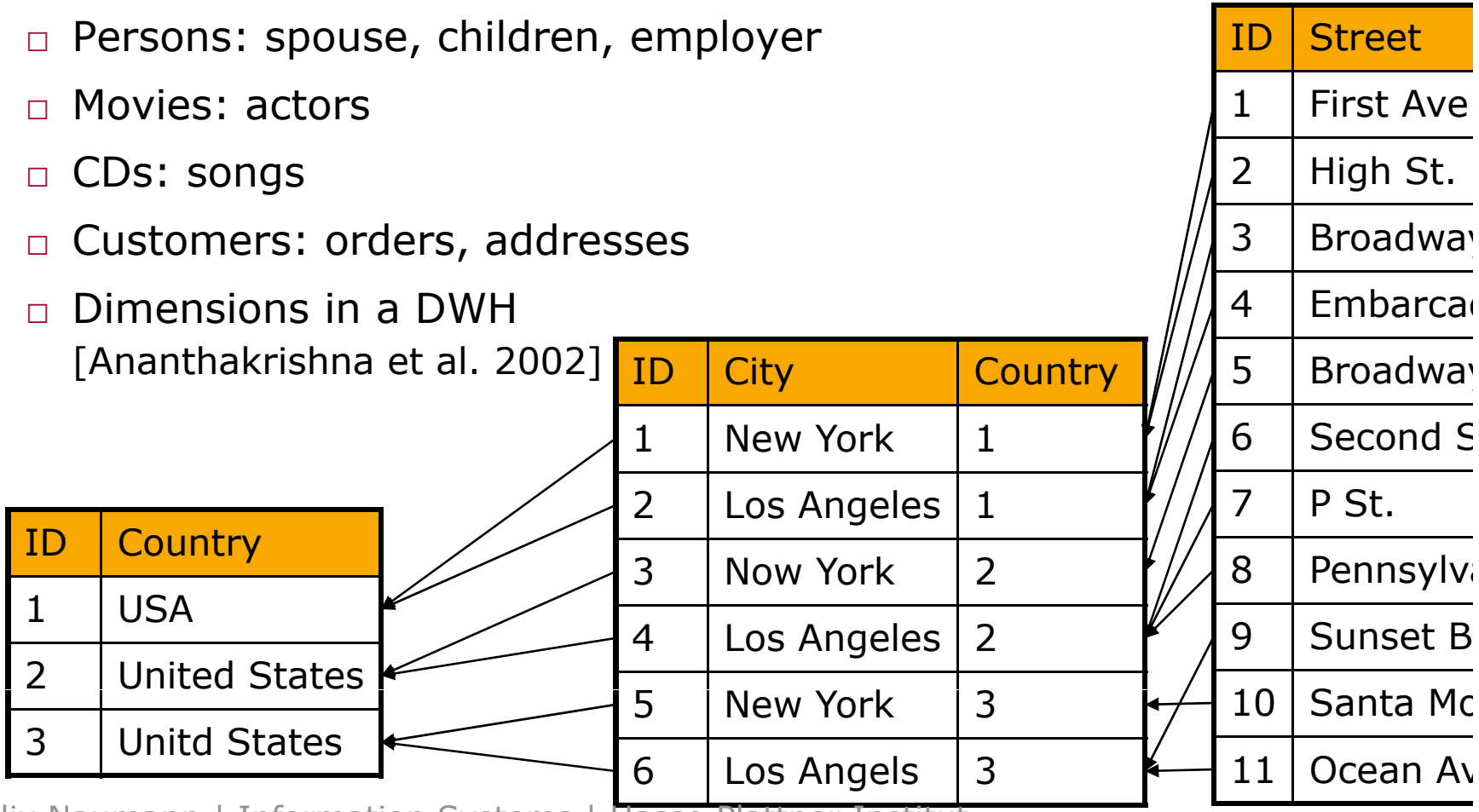


Domain-dependent Similarity Measures

- Data Types
  - Special similarity for dates
  - Special similarity for numerical attributes
  - ...
- Rules
  - [Hernandez Stolfo 1998], [Lee et al. 2000]
  - **Given two records, r1 and r2.**
    IF last name of r1 = last name of r2,
    AND first names differ slightly,
    AND address of r1 = address of r2
    THEN r1 is equivalent to r2.

Relationship-aware Similarity Measures

- Idea: Not only values of the records, but values of related records are relevant for similarity.
  - Persons: spouse, children, employer
  - Movies: actors
  - CDs: songs
  - Customers: orders, addresses
  - Dimensions in a DWH
    [Ananthakrishna et al. 2002]

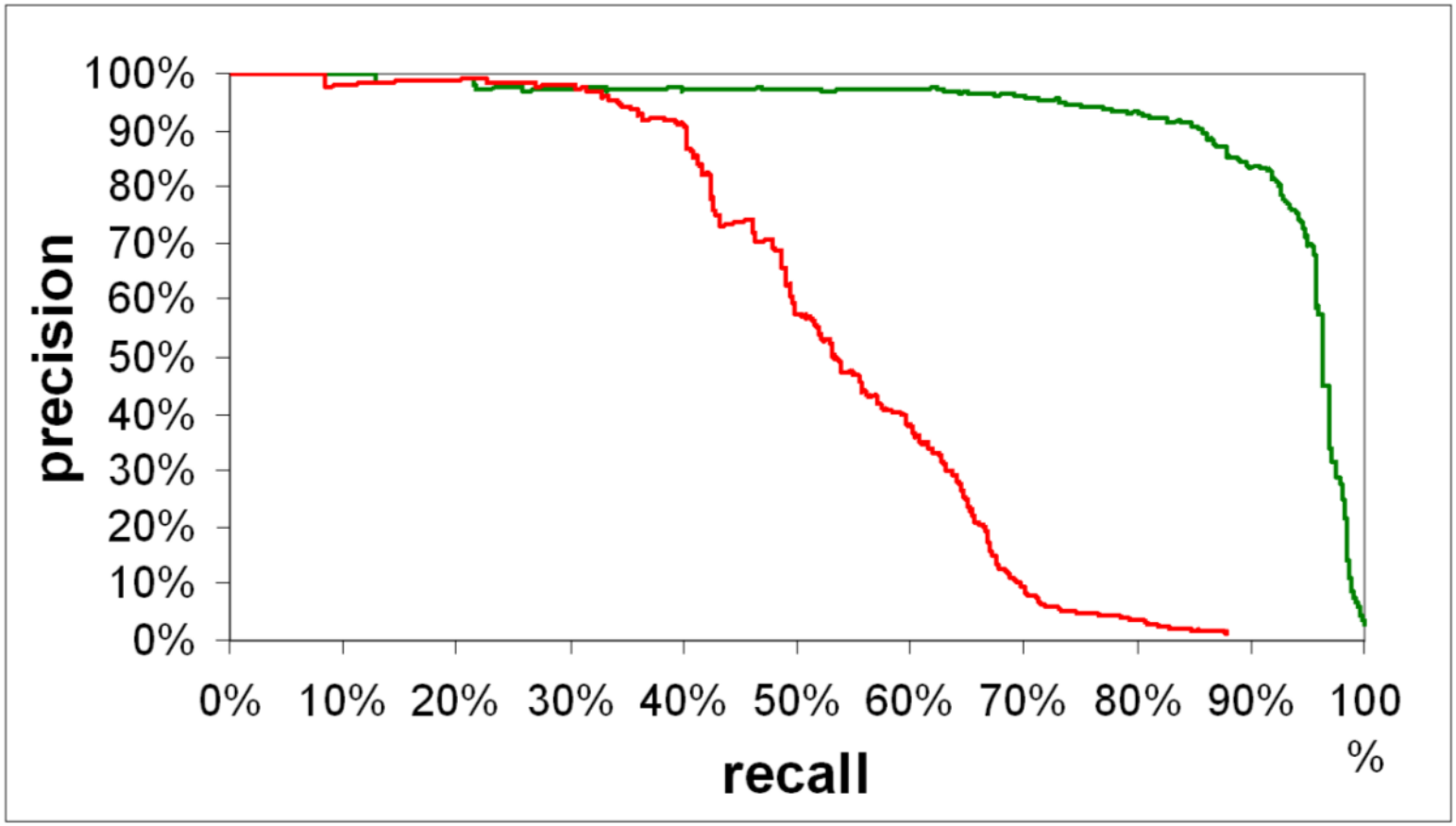


ID	Country
1	USA
2	United States
3	Unitd States

ID	City	Country
1	New York	1
2	Los Angeles	1
3	Now York	2
4	Los Angeles	2
5	New York	3
6	Los Angels	3

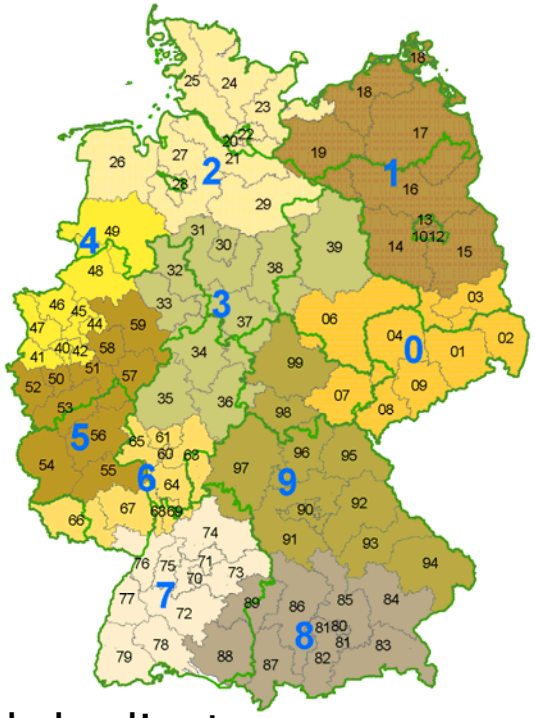
ID	Street
1	First Ave
2	High St.
3	Broadwa
4	Embarca
5	Broadwa
6	Second S
7	P St.
8	Pennsylv
9	Sunset B
10	Santa Mo
11	Ocean Av

Relationship-aware Similarity Measures – Evaluation



Partitioning / Blocking

- Partition the records (horizontally) and compare pairs of records only within a partition.
  - Partitioning by first two zip-digits
    - Ca. 100 partitions in Germany
    - Ca. 100 customers per partition
    - => 495.000 comparisons
  - Partition by first letter of surname
  - …



Source: wikipedia.de

- Idea: Partition multiple times by different criteria.
  - Then apply transitive closure on discovered duplicates.

Sorted Neighborhood
[Hernandez Stolfo 1998]

- Idea
  - Sort tuples so that similar tuples are close to each other.
  - Only compare tuples within a small neighborhood (window).

1. Generate key
   - E.g.: SSN+"first 3 letters of name" + ...
2. Sort by key
   - Similar tuples end up close to each other.
3. Slide window over sorted tuples
   - Compare all pairs of tuples within window.

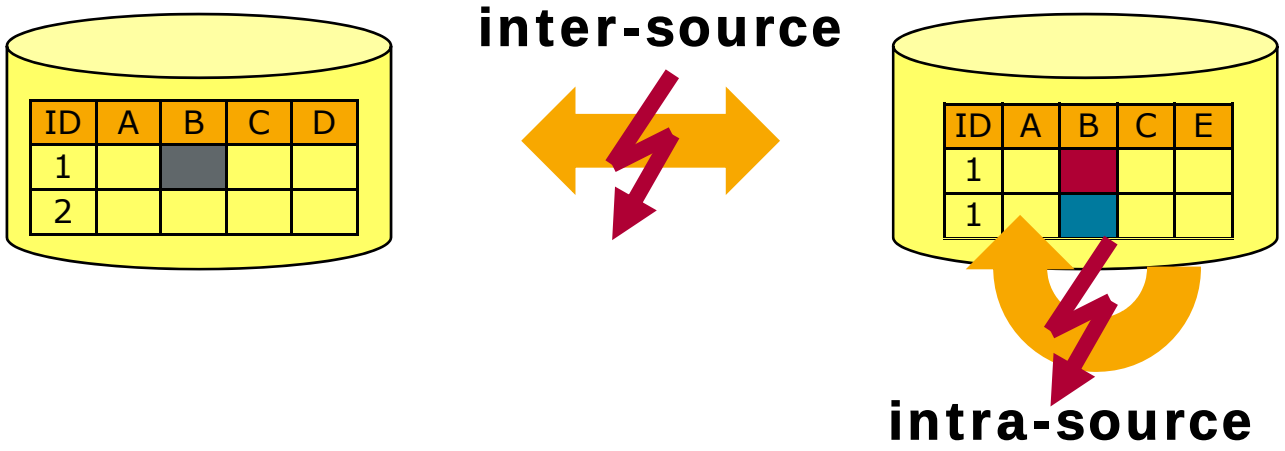
- Problems
  - Choice of key
  - Choice of window size
- Complexity: At least 3 passes over data
  - Sorting!

Overview

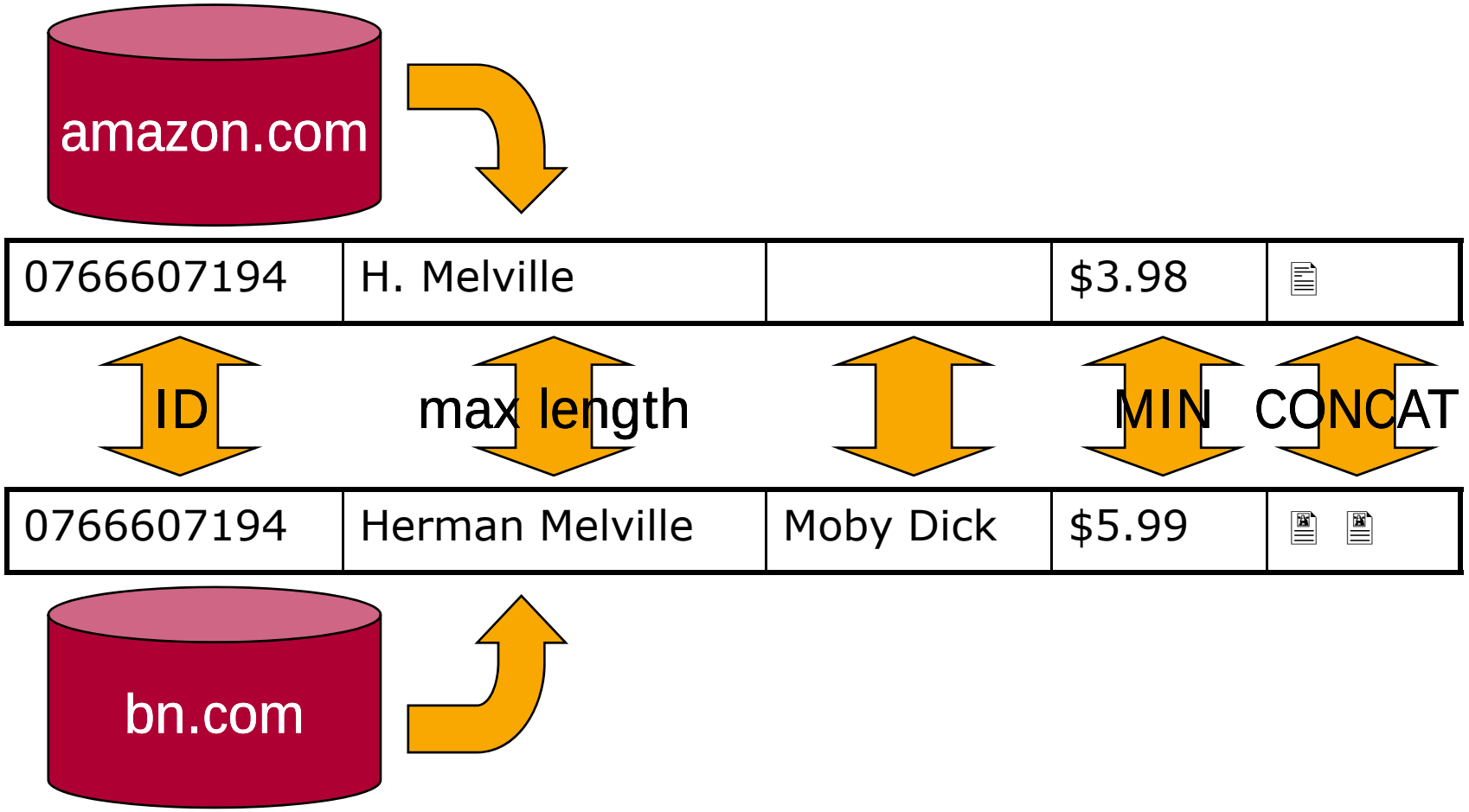
- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
  - Data Conflicts
  - Relational Operators
  - Conflict Resolution
  - Tools
- Summary

Data conflicts

Two duplicates have different values for semantically same attribute.



Data Fusion


amazon.com
0766607194
H. Melville
$3.98
ID
max length
MIN
CONCAT
0766607194
Herman Melville
Moby Dick
$5.99
bn.com

Data conflicts – origins

- No integrity or consistency checks
- Redundant schemata
- Typos, transmission errors, incorrect calculations
- Variants
  - Kantstr. / Kantstrasse / Kant Str. / Kant Strasse
  - Kolmogorov / Kolmogoroff / Kolmogorow
- Typical confusion (OCR)
  - U<->V, 0<->o, 1<->l, etc.
- Obsolete values
  - Different update frequencies, forgotten update

Within information system

Data conflicts – origins

- Locally consistent but globally inconsistent
- Duplicates
- Different data types
- Local spelling variations and conventions

Across information systems

Examples for errors

- Addresses
  - Str. → Straße, Ch. → Chaussee, etc.
  - R.-Breitscheid-Str. 72 a → Rudolf-Breitscheid.-Str. 72A
  - 128 spellings for Frankfurt am Main
    - Frankfurt a.M., Frankfurt/M Frankfurt, Frankfurt a. Main, …
- Names
  - Dr. Ing. h.c. F. Porsche AG
  - Hewlett-Packard Development Company, L.P.
- Numerical data
  - 10.000 € = 10T EURO = 10k EUR = 10.000,00€ = 10,000.- €
- Phone numbers
- Birth dates

Numerical data - consequences

Southwest

Published on Chanhassen Villager (http://www.chanvillager.com)

Property mistakenly valued at $189 million

By rcraw
Created 12/03/2007 - 4:46pm

Property mistakenly valued at $189 million results in tax adjustments in county

An $18,900 Waconia property that was mistakenly valued at $189 million is "throwing a wrench" into property tax statements and the Carver County budget. County officials issued a press release Monday detailing the problem that came to light last week.

An error was identified in the estimated market valuations used to calculate Pay 2008 Proposed Property Taxes, according to the release. The County Assessor's Office placed an incorrect estimated market value on a parcel located in the city of Waconia, apparently resulting in extra zeroes being added to the value.

The mistake results in an imbalance in the amount of property taxes the county was expecting to collect. The mistake added about $900,000 in expected revenue, according to County Administrator David Hemze.

The county is planning to consider recommendations to cut the 2008 budget by $900,000 so that proposed property taxes will match tax notices sent to residents in November.

"It kind of threw a wrench into everything," said Hemze. "It's unfortunate. It's a mistake and we're concentrating on responding to the mistake and trying to ensure that it doesn't happen again." If the county does not cut the budget by $900,000, the county portion of property taxes would go up for all properties in the county. The effect would be greatest in Waconia, but Hemze said the average-valued home outside of Waconia would also experience a $29 increase on top of the number indicated on the November tax notices.

Numerical data - consequences

SPIEGEL ONLINE

28. Januar 2008, 11:27 Uhr

FRANKREICH

Telefonkundin erhält Rechnung über 63 Millionen Euro

Als eine Französin aus Lothringen unlängst ihre Telefonrechnung bekam, blieb ihr buchstäblich die Spucke weg: 63 Millionen Euro sollte sie begleichen. Dabei hatte sie ursprünglich nur um Korrektur einer Abrechnung in Höhe von 67 Euro gebeten.

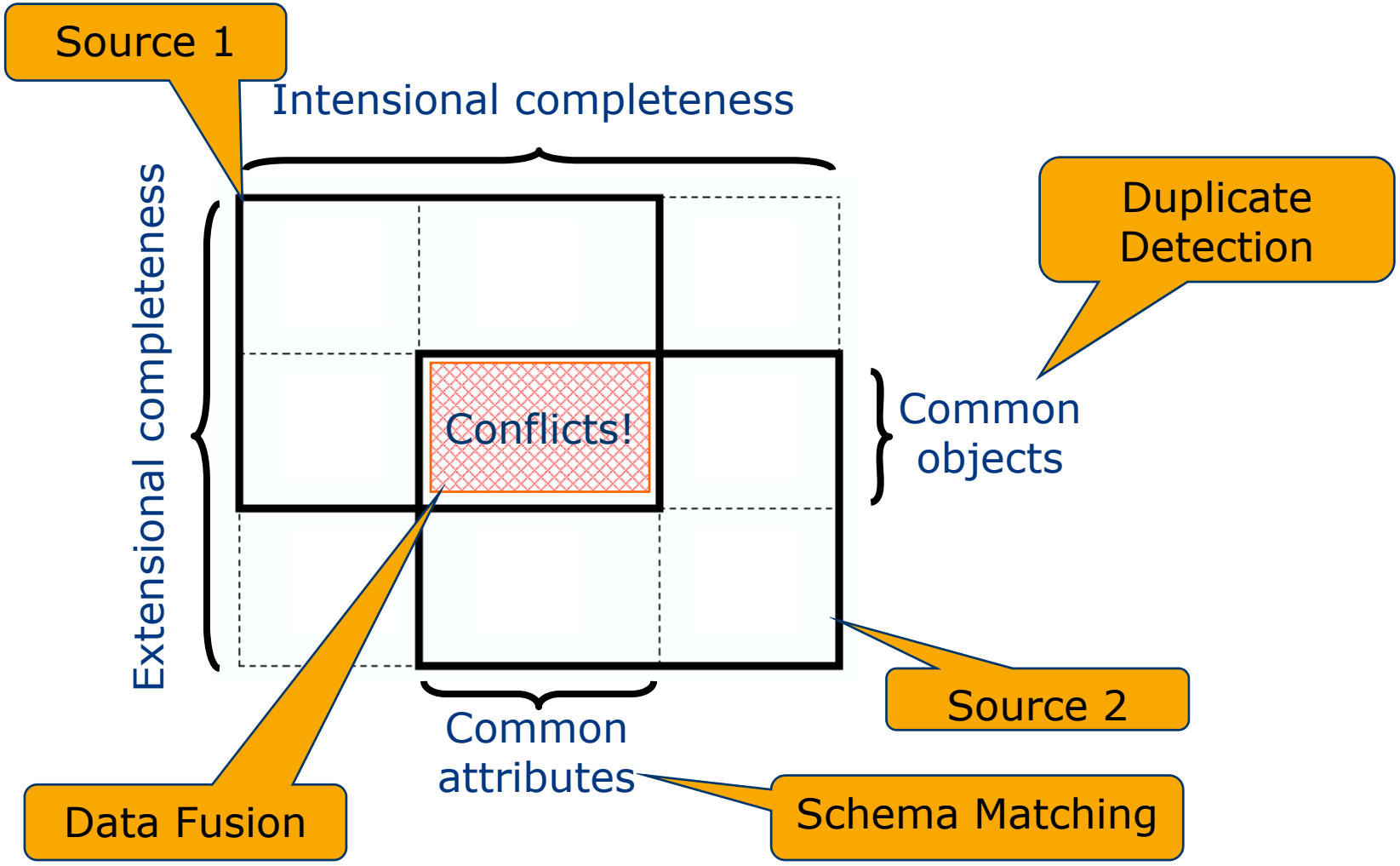
Paris - "Da muss wohl ein Komma verrutscht sein", zitiert "Le Figaro" heute den Vizedirektor der französischen Telefongesellschaft Télé2, Olivier Anstett. Die Kundin aus dem Ort Herserange in der Nähe von Metz hatte sich zunächst über einen ihrer Meinung nach zu hohen Rechnungsbetrag von 67,69 Euro bei der Telefongesellschaft beschwert. Als eine Antwort ausblieb, schickte sie einen zweiten Brief. Daraufhin erhielt sie eine "korrigierte" Rechnung über die Summe 63.280.067,96 Euro.

"Uns bleibt nur, uns bei der Kundin zu entschuldigen und dafür zu sorgen, dass so etwas nie wieder vorkommt", so der lapidare Kommentar des Vizechefs von Télé2.

Data conflict – elimination

- Error correction
  - Reference tables
    - Cities, countries, products ...
  - Similarity measures
    - For typos
    - For language-specific variants (Meier, Mayer,...)
  - Standardization and Transformation
  - Domain-knowledge (meta data)
    - Konventions (country/region-specific spelling)
    - Ontologies
    - Thesauri, dictionaries for homonyms, synonyms, ...
- And data fusion…

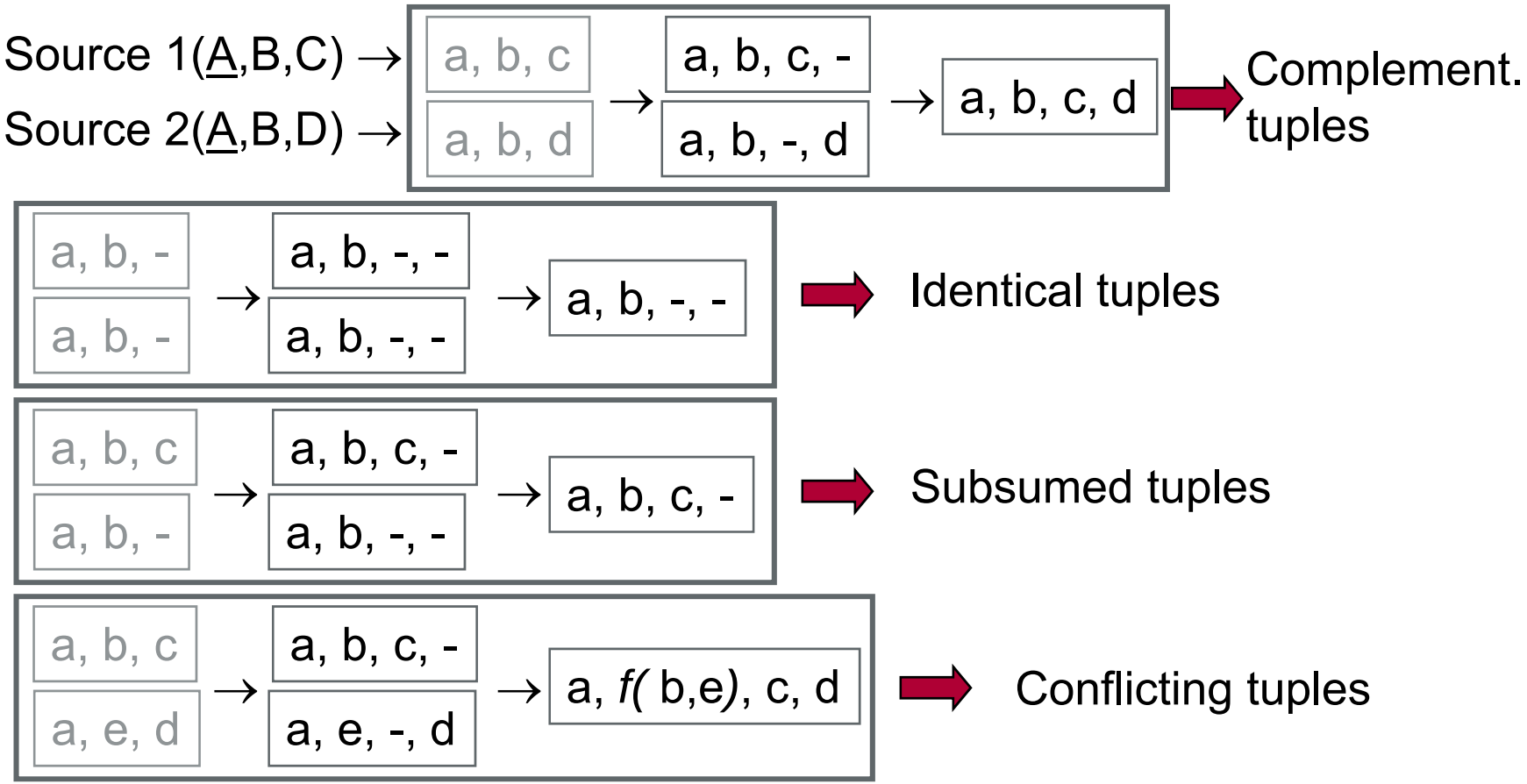
Completeness and Conciseness



Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
  - Data Conflicts
  - Relational Operators
  - Conflict Resolution
  - Tools
- Summary

“Proper” Data Fusion



Relational object integration

Union

✚ Elimination of exact duplicates

Minimum Union, [GL94]

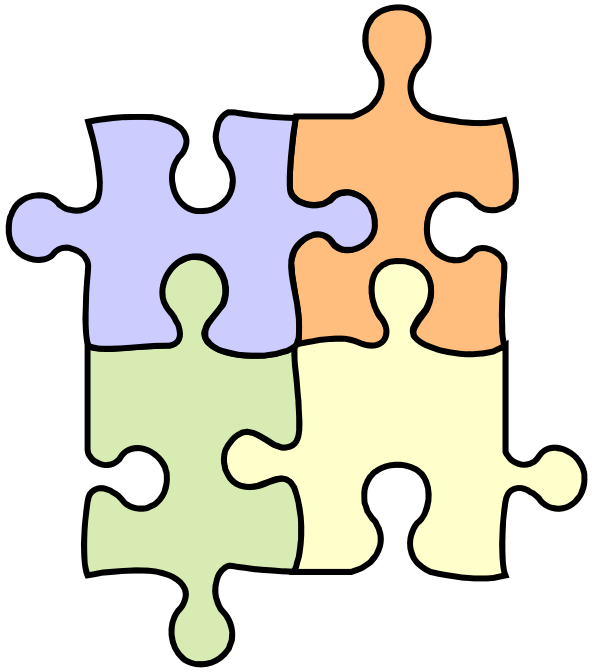
✚ Elimination of subsumed tuples

But

✖ No duplicate integration

✖ Conflict resolution

Later: Join, Merge, Group, …



Minimum Union – Outer Union

- Outer Union pads relations with NULL-values, to unify schemata.
- Then normal UNION.
- Usually not implemented in SQL

R

A	B	C
P	1	2
P	2	1
Q	1	2

S

B	D
2	U
3	V

$R \uplus S$

A	B	C	D
P	1	2	⊥
P	2	1	⊥
Q	1	2	⊥
⊥	2	⊥	U
⊥	3	⊥	V

Minimum Union – Subsumption

- A tuple t_1 subsumes a tuple t_2, if
  - Has same schema,
  - t_2 has more NULL-values than t_1,
  - $t_1 = t_2$ for all non-NULL-values of t_2.

- Notation:
  - R↓ returns those tuples of R, that are not subsumed by any other tuple in R.

R

p_id	fname	lname	age
1	Peter	Müller	32
1	Peter	Müller	⊥
1	Peter	⊥	⊥
1	Peter	⊥	32
1	Peter	⊥	42
2	Wiebke	⊥	2
2	⊥	Meyer	2

How many tuples does R↓ have?

R↓

p_id	fname	lname	age
1	Peter	Müller	32
1	Peter	⊥	42
2	Wiebke	⊥	2
2	⊥	Meyer	2

Minimum Union – NULL-values

Semantics of NULL? [GUW02]

- "*unknown*"
  - There is a value, but I do not know it.
  - E.g.: Unknown birthday
- "*inapplicable*"
  - There is no meaningful value
  - E.g.: partner for singles
- "*withheld*"
  - There is a value, but we are not authorized to see it.
  - E.g.: Private phone

Minimum Union – NULL-values

"Value not supplied"

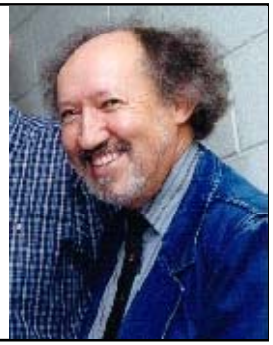
"Value does not exist"

"Value undefined"

„Distinguished" NULL

"Total ignorance" NULL

C.J. Date:
- "Into the Unknown"
- "Much Ado About Nothing"
- "NOT Is Not Not!"
- "Oh No Not Nulls Again"
-…



From now on: "**Unknown**"

Merge and Prioritized Merge

Merge (⊠), [GPZ01]

- Mixes Join and Union to a new operator
  - Fuses complementary tuples (only from different sources)
- COALESCE removes NULL values
- Priorization possible (▷)
- Can be expressed with standard SQL

```
( SELECT K.p_id, K.fname, Coalesce(K.lname, C.lname), Coalesce(K.age, C.age)
FROM K LEFT OUTER JOIN C ON K.p_id = C.p_id )
        UNION
( SELECT C.p_id, K.fname, Coalesce(C.lname, K.lname), Coalesce(C.age, K.age)
FROM K RIGHT OUTER JOIN C ON K.p_id = C.p_id )
```

Merge – example

Kunde K

p_id	fname	lname	age
1	Peter	Müller	32
2	Franz	Schmidt	55
3	Wiebke	Meyer	⊥
4	Klaus	Lehmann	28

Customer C

p_id	lname	age
1	⊥	32
2	Schmidt	⊥
3	Meier	56
5	Weger	47

C ⊠ K

p_id	fname	lname	age
1	Peter	Müller	32
2	Franz	Schmidt	55
3	Wiebke	Meier	56
3	Wiebke	Meyer	56
4	Klaus	Lehmann	28
5	⊥	Weger	47

What else is there?

Match Join [YaÖz99]

- Complex operator
- *HighConfidence*, *RandomEvidence*, and *PossibleAtAll*

ConQuer [FuFM05]

- „Consistent Query Answering“
- Rewriting of SQL queries

Burdick et. al. [BDJR05]

- Uncertainty in Data Warehouses
- „Possible Worlds“

Probabilistic Models [Mich89]

- Extending schema by probabilities

Grouping for Integration

$K \uplus C$

p_id	fname	lname	age
1	Peter	Müller	32
2	Franz	Schmidt	55
3	Wiebke	Meyer	55
1	⊥	Müller	32
2	⊥	Schmidt	⊥
3	⊥	Meier	56

p_id	fname	lname	age
1	Peter	Müller	32
2	Franz	Schmidt	55
3	Wiebke	Meier	56

```
SELECT      p_id, MAXLEN(fname), CHOOSE(lname,C), MAX(age)
FROM        K ⊎ C
GROUP BY    p_id
```

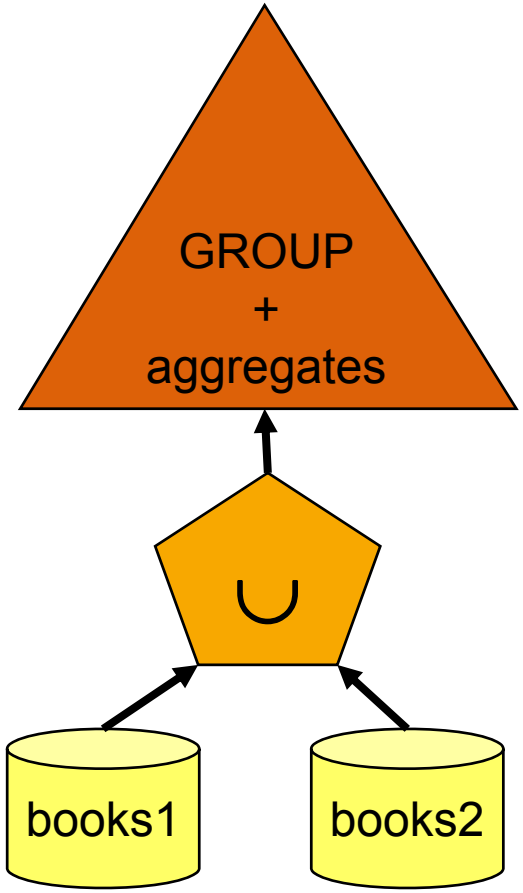
Longest String

C is favored source

Highest value

Grouping – example

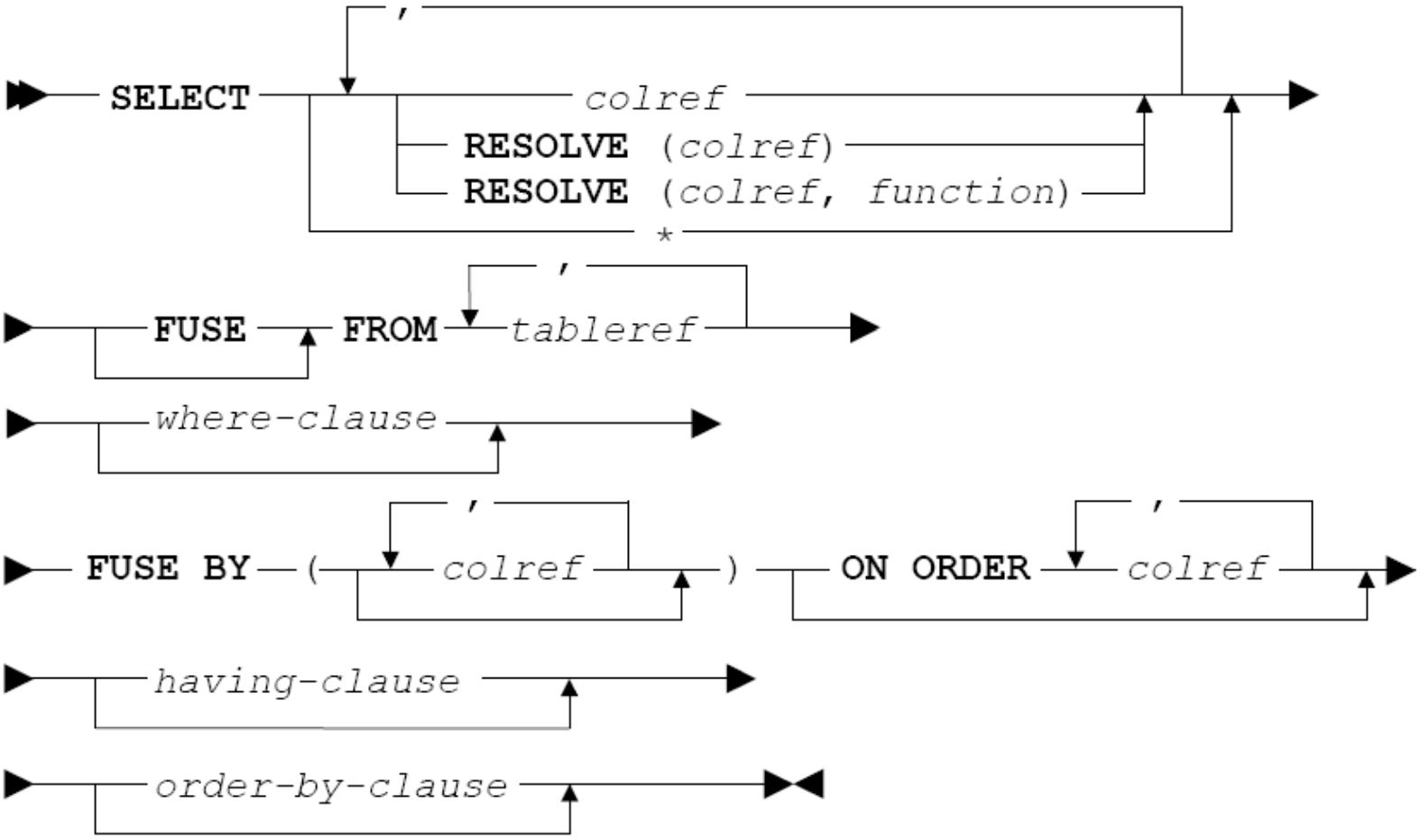

```
SELECT books.isbn,
       MAXLEN(books.title),
       MIN(books.price)
FROM   (
            SELECT * FROM books1
                  UNION
            SELECT * FROM books2
       )
       AS books
GROUP BY
       books.isbn
```



Grouping – Pros and Cons

- Pros
  - Efficient
    - Implemented by sorting
  - Catches duplicates within source and across sources
  - Simple / short
- Cons
    - Restricted to built-in standard aggregate-functions:
      - MAX, MIN, AVG, VAR, STDDEV, SUM, COUNT
    - Grouping only by equality
      - ID attribute is necessary
    - Outer Union usually not implemented

FUSE BY Syntax Diagram



Fuse By – Queries


```
SELECT *
FUSE FROM Q1
FUSE BY (Name)
```

Grouping with coalesce aggregation

```
SELECT *
FUSE FROM Q1
FUSE BY ()
```

Subsumption

```
SELECT *
FUSE FROM Q1, Q2
FUSE BY ()
```

Minimum Union

```
SELECT Name, RESOLVE(Age, max), RESOLVE(Student,
       vote), RESOLVE(Place), RESOLVE(Phone)
FUSE FROM Q1, Q2
FUSE BY (Name) ON ORDER Q2.Age DESC
```

FUSE BY – Example

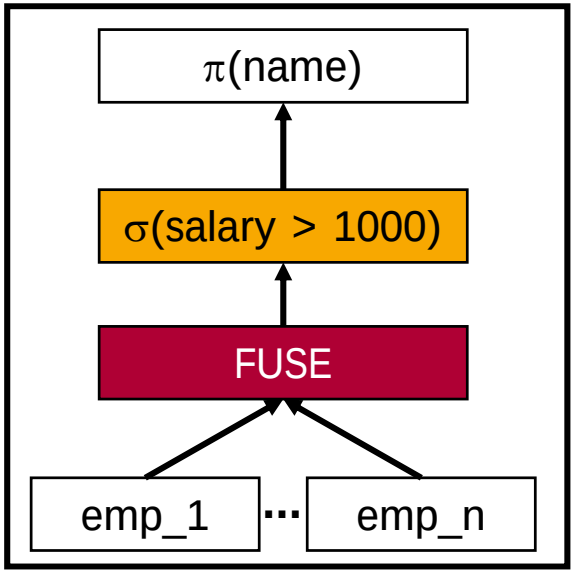
Name	Age	Student	Place
Felix	⊥	No	Hamburg
Melanie	22	Yes	⊥
Jens	⊥	Yes	Karlsruhe
Christoph	25	Yes	Berlin
Sven	26	Yes	Berlin
Sven	⊥	Yes	Berlin

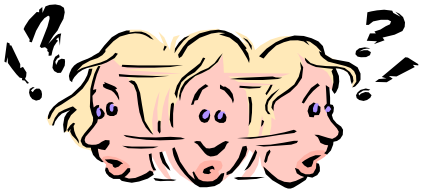
Name	Age	Student	Phone
Melanie	⊥	Yes	030/12345
Jens	27	⊥	030/54321
Christoph	24	Yes	⊥
Melanie	21	No	030/98765
Karsten	24	Yes	⊥
Karsten	24	Yes	⊥

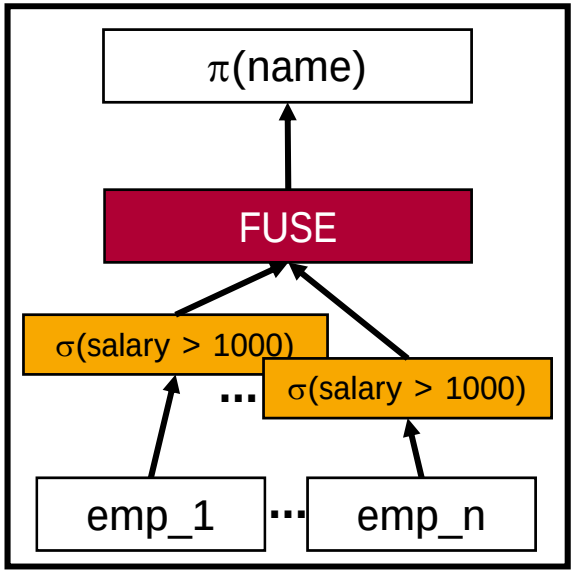
```
SELECT Name,
 RESOLVE(Age,max),
 RESOLVE(Student,vote),
 RESOLVE(Place),
 RESOLVE(Phone)
FUSE FROM Q1, Q2
FUSE BY (Name)
ON ORDER Q2.Alter DESC
```

Name	Age	Student	Place	Phone
Felix	⊥	No	Hamburg	⊥
Melanie	22	Yes	⊥	030/98765
Jens	27	Yes	Karlsruhe	030/54321
Christoph	25	Yes	Berlin	⊥
Sven	26	Yes	Berlin	⊥
Karsten	24	Yes	⊥	⊥

Query Optimization with Fusion





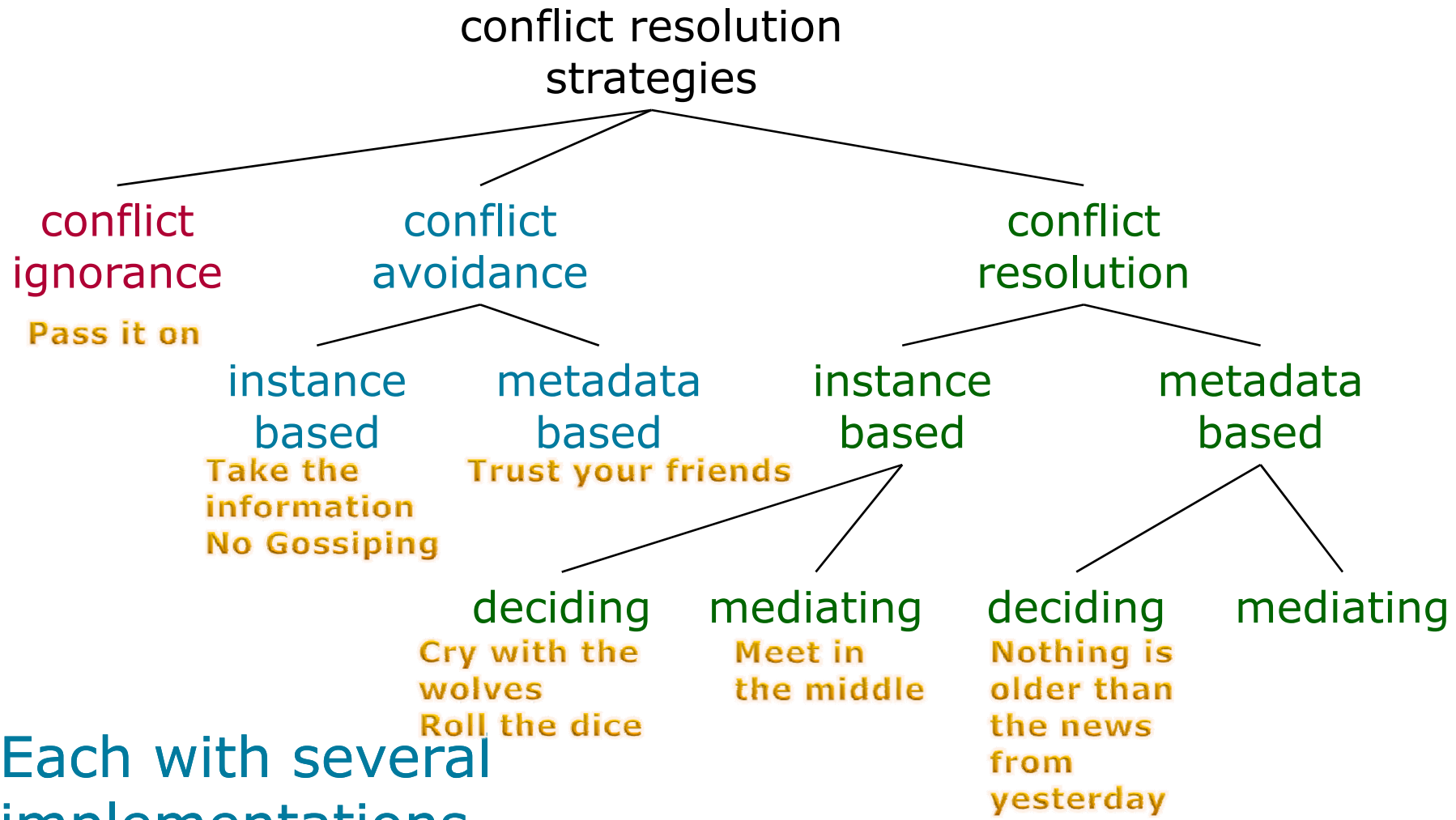


Correctness?
Completeness?
Efficiency?

Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
  - Data Conflicts
  - Relational Operators
  - Conflict Resolution
  - Tools
- Summary

Classification of strategies



Each with several implementations

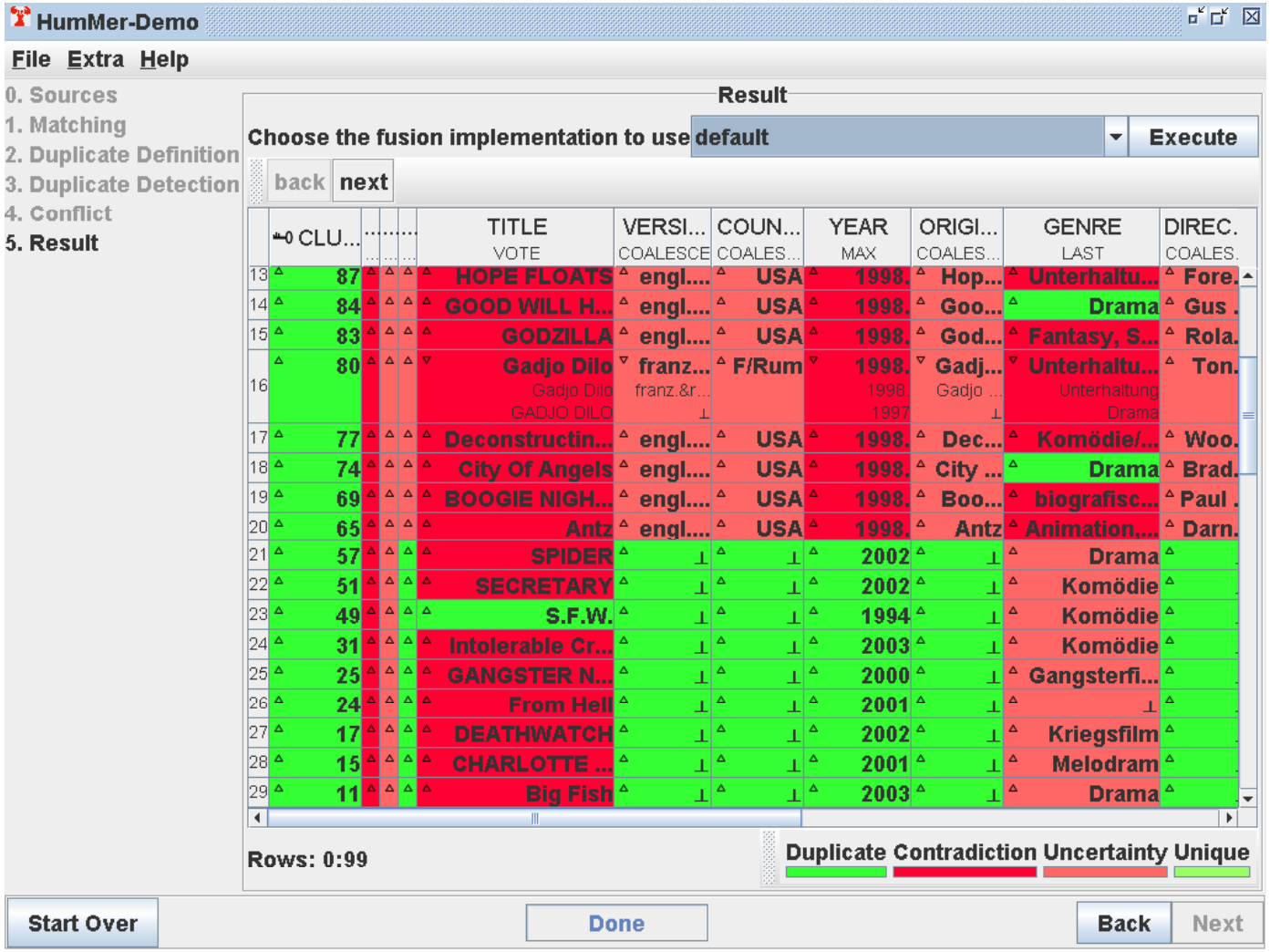
Conflict Resolution Functions

Min, Max, Sum, Count, Avg, StdDev	Standard aggregation
Random	Random choice
First, Last	Choose first/last value; depends on order
Longest, Shortest	Choose longest/shortest value
Choose(*source*)	Choose value froma particular source
ChooseDepending(*col*, *val*)	Choose depending on *val* in other column *col*
Vote	Majority decision
Coalesce	Choose first non-null value
Group, Concat	Group or concatenate all values
MostRecent	Choose most recent (up-to-date) value
MostAbstract, MostSpecific	Use a taxonomy / ontology
….	….

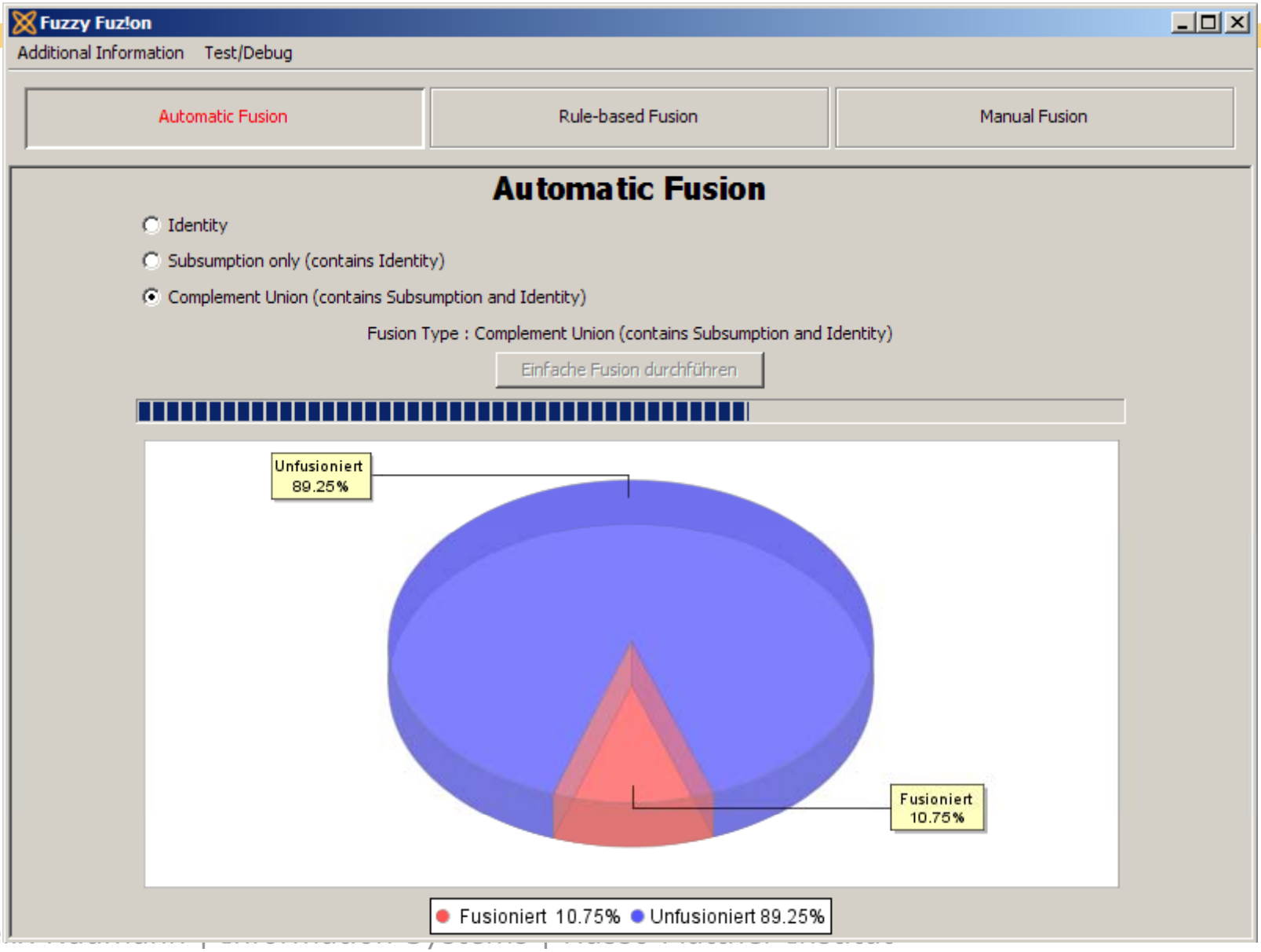
Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
  - Data Conflicts
  - Relational Operators
  - Conflict Resolution
  - Tools
- Summary

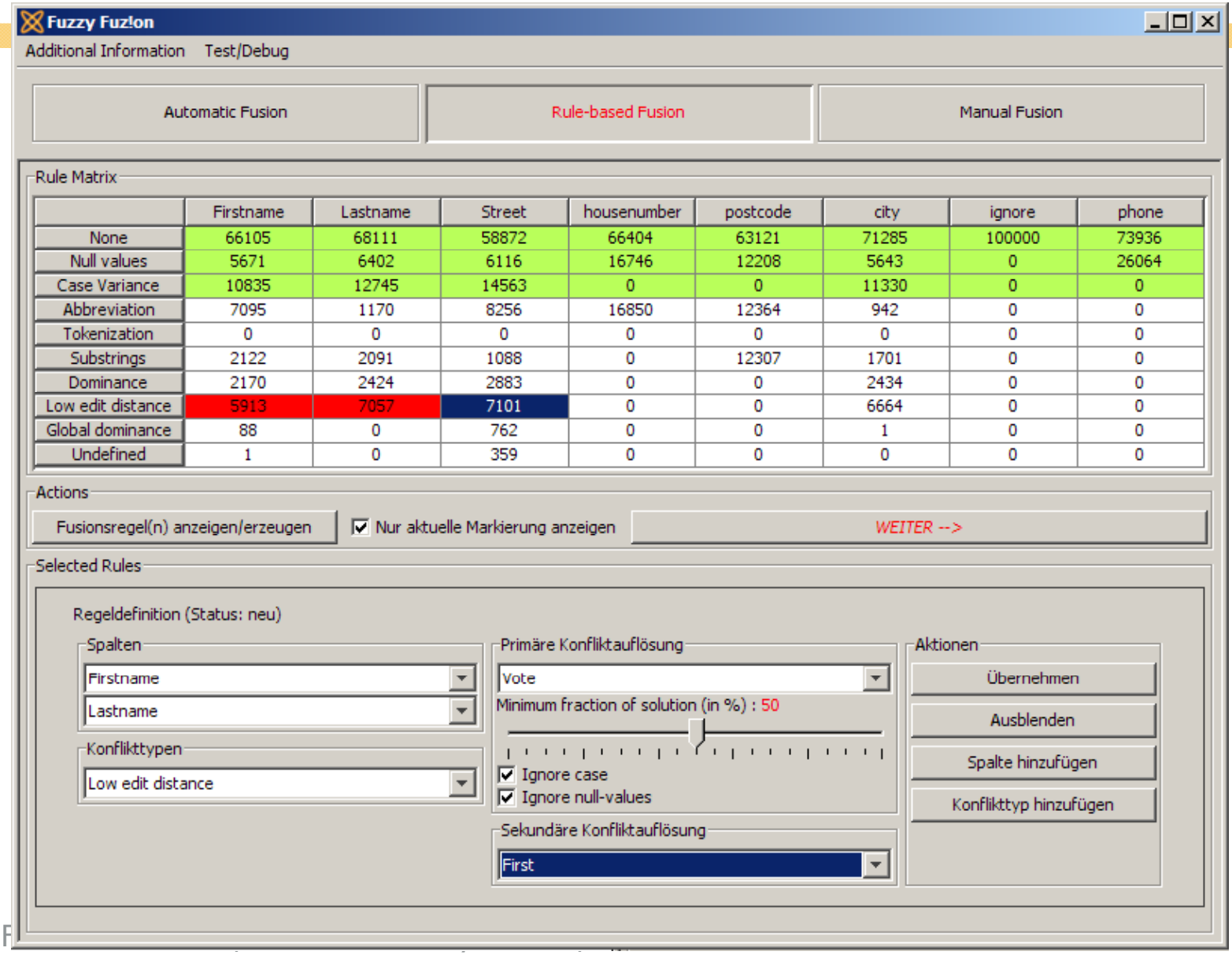
Visualization of Integrated Data



Bachelorprojekt „Fuz!on“

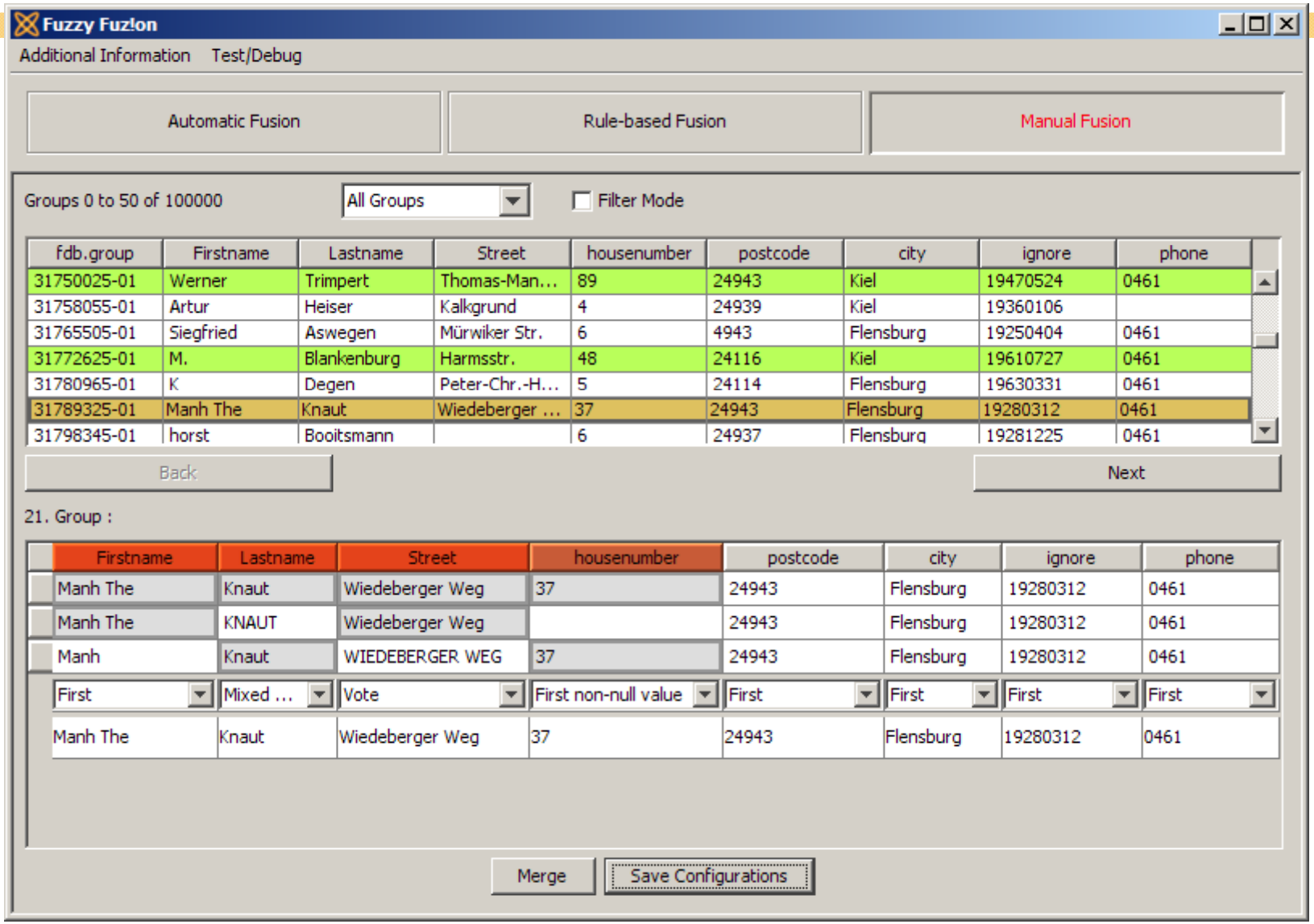


Bachelorprojekt „Fuz!on“

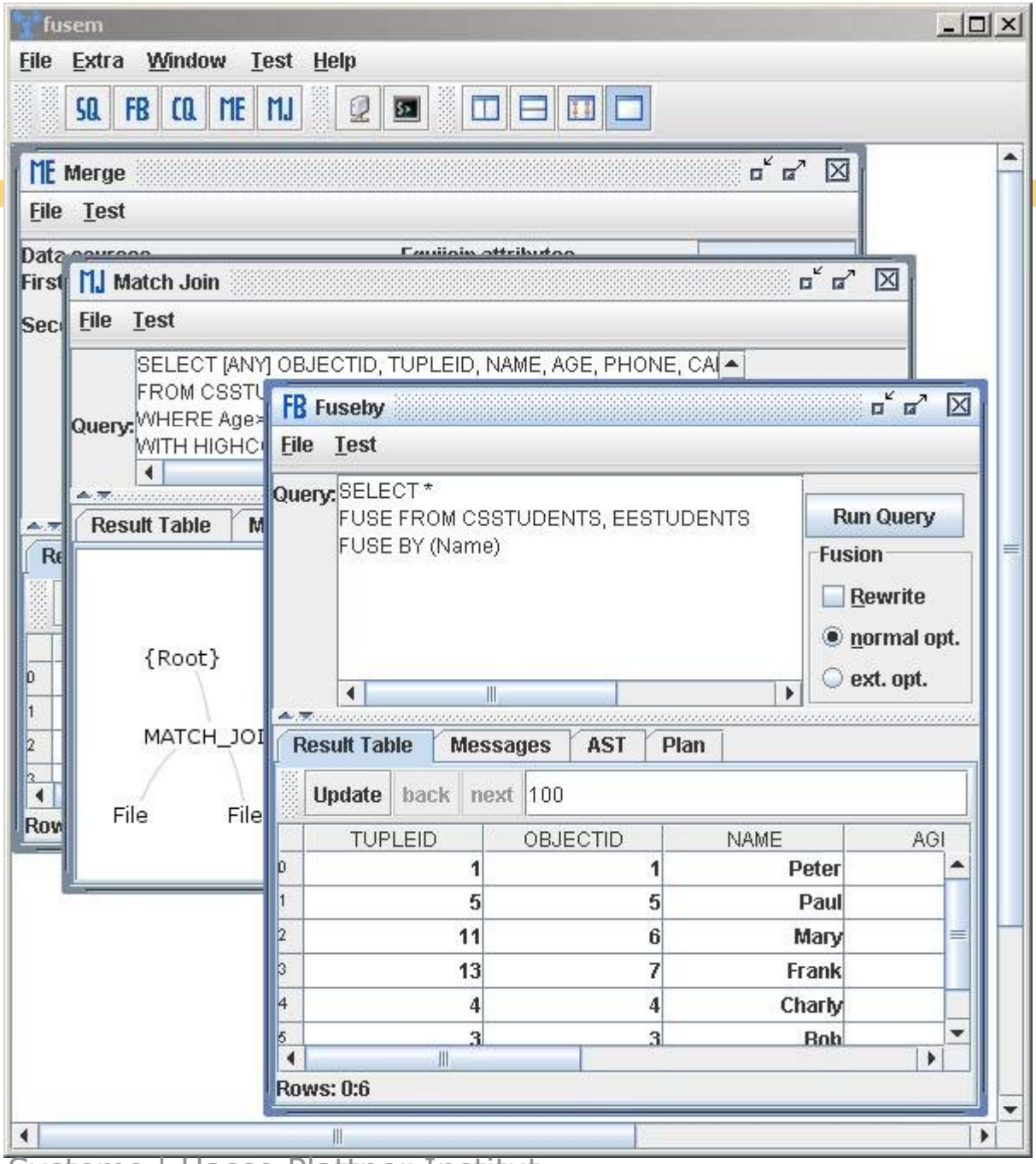


	Firstname	Lastname	Street	housenumber	postcode	city	ignore	phone
None	66105	68111	58872	66404	63121	71285	100000	73936
Null values	5671	6402	6116	16746	12208	5643	0	26064
Case Variance	10835	12745	14563	0	0	11330	0	0
Abbreviation	7095	1170	8256	16850	12364	942	0	0
Tokenization	0	0	0	0	0	0	0	0
Substrings	2122	2091	1088	0	12307	1701	0	0
Dominance	2170	2424	2883	0	0	2434	0	0
Low edit distance	5913	7057	7101	0	0	6664	0	0
Global dominance	88	0	762	0	0	1	0	0
Undefined	1	0	359	0	0	0	0	0

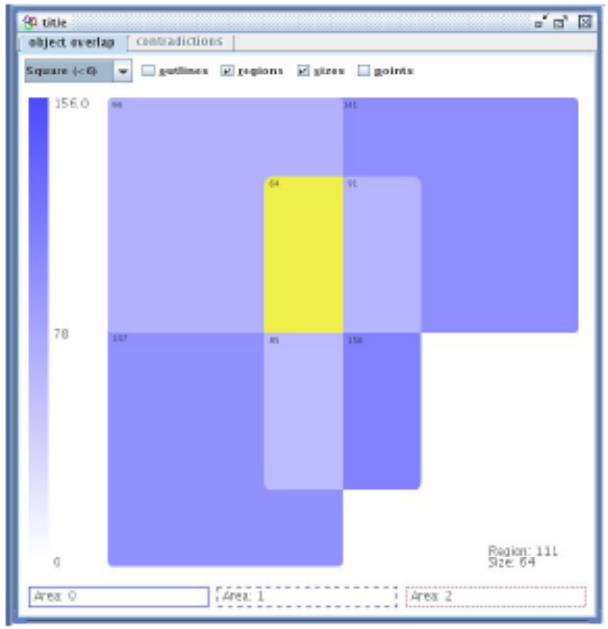
Bachelorprojekt „Fuz!on“



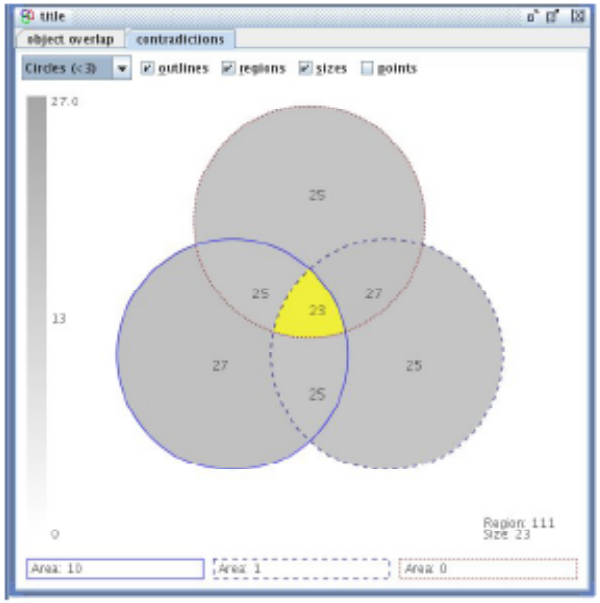
Tool: fusem


fusem
File Extra Window Test Help
SQ FB CQ ME MJ
ME Merge
File Test
MJ Match Join
File Test
Query: SELECT [ANY] OBJECTID, TUPLEID, NAME, AGE, PHONE, CA
FROM CSSTU
WHERE Age=
WITH HIGHC
Result Table
{Root}
MATCH_JOI
File
File
FB Fuseby
File Test
Query: SELECT *
FUSE FROM CSSTUDENTS, EESTUDENTS
FUSE BY (Name)
Run Query
Fusion
Rewrite
normal opt.
ext. opt.
Result Table
Messages
AST
Plan
Update
back
next
100
TUPLEID
OBJECTID
NAME
AGI
0 1 1 Peter
1 5 5 Paul
2 11 6 Mary
3 13 7 Frank
4 4 4 Charly
5 3 3 Bob
Rows: 0:6

Visualizing semantics and conflicts



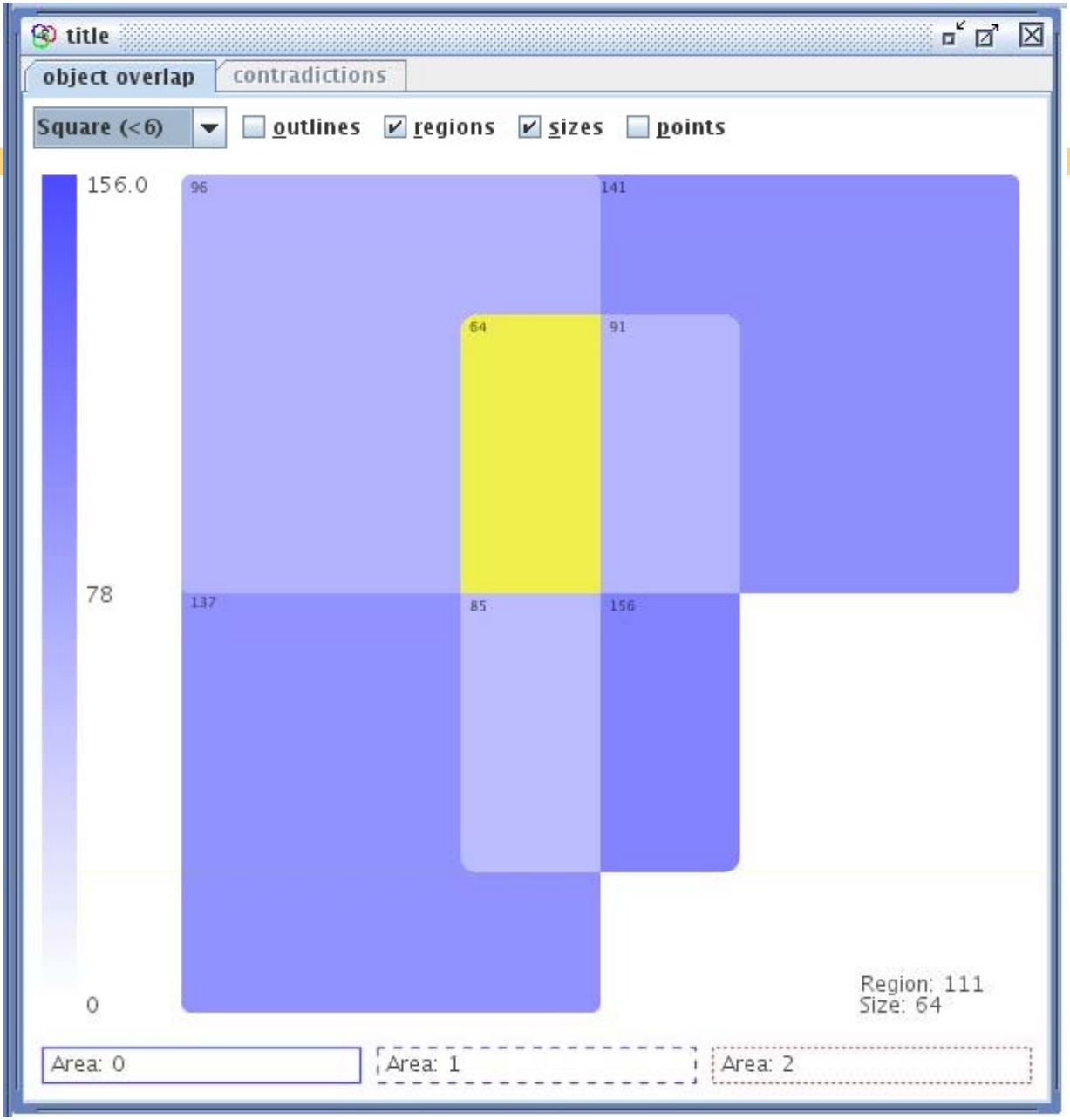
(a) Object overlap

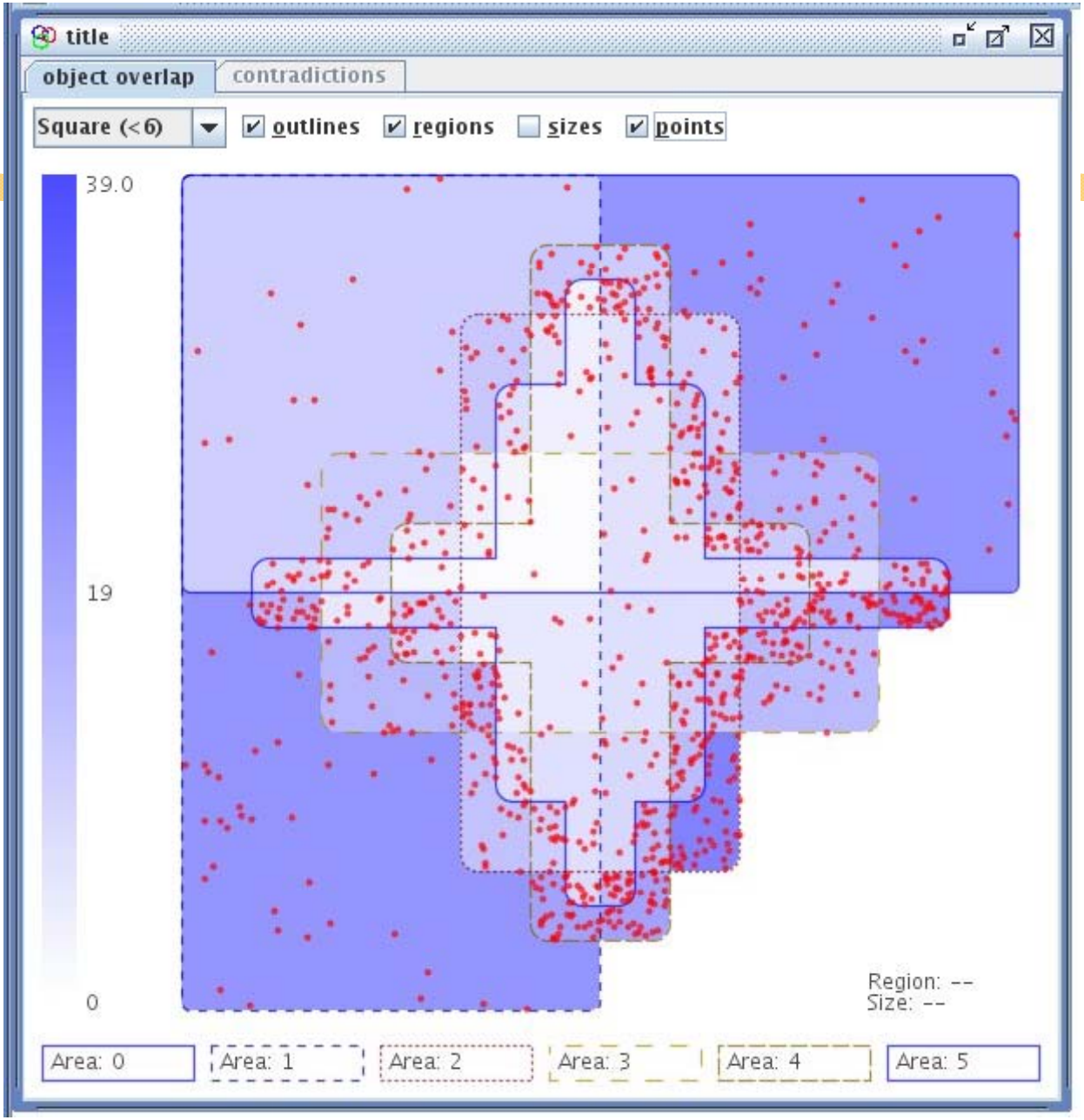


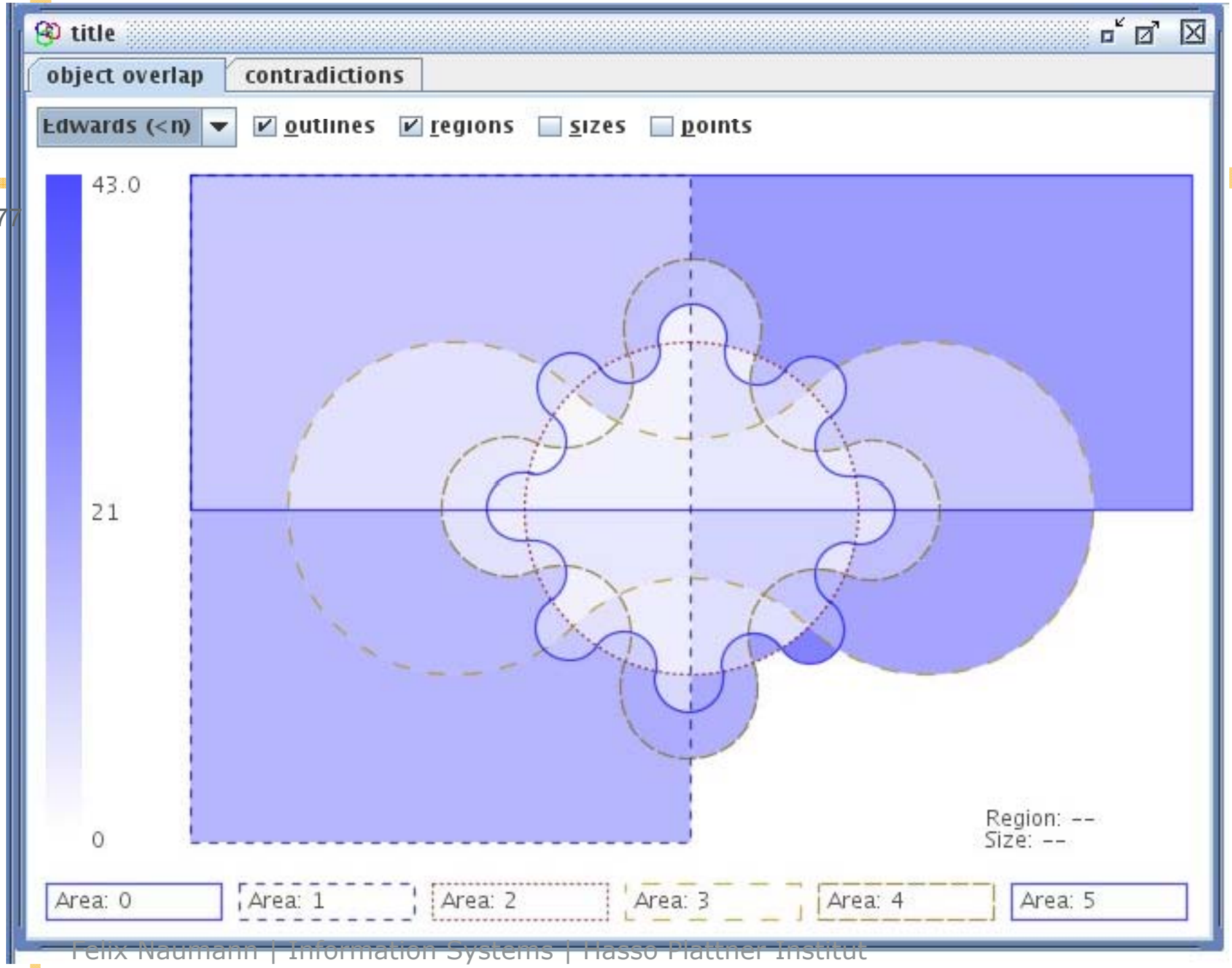
(b) Contradiction Count

Object	Sematics	X	Y
o_1	Merge	1	2
	FuseBy	1	2
	MatchJoin	1	3
o_2	Merge	1	1
	FuseBy	1	1
	MatchJoin	1	1
o_3	Merge	1	2
	FuseBy	3	4
	MatchJoin	5	6
o_4	...	...	...
...	...	...	...

(c) Contradicting tuples


title
object overlap
contradictions
Square (<6)
outlines
regions
sizes
points
156.0
78
0
96
141
64
91
137
85
156
Region: 111
Size: 64
Area: 0
Area: 1
Area: 2


title
object overlap
contradictions
Square (<6)
outlines
regions
sizes
points
39.0
19
0
Region: --
Size: --
Area: 0
Area: 1
Area: 2
Area: 3
Area: 4
Area: 5


title
object overlap
contradictions
Edwards (<n)
outlines
regions
sizes
points
43.0
21
0
Region: --
Size: --
Area: 0
Area: 1
Area: 2
Area: 3
Area: 4
Area: 5

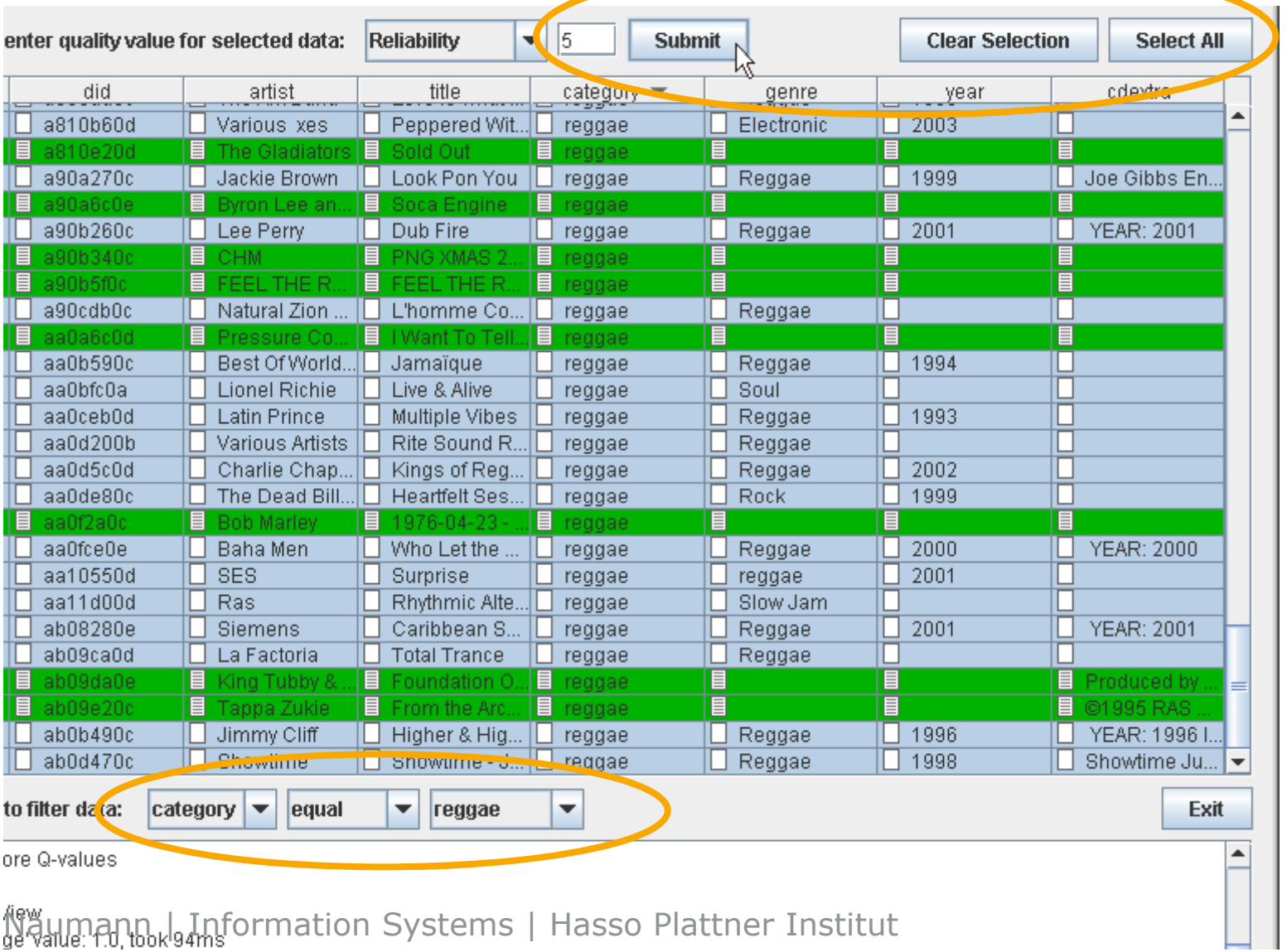
Overview

- Introductory example
- Step 1: Schema Matching
- Step 2: Duplicate detection
- Step 3: Data fusion
  - Data Conflicts
  - Relational Operators
  - Conflict Resolution
  - Tools
- Summary

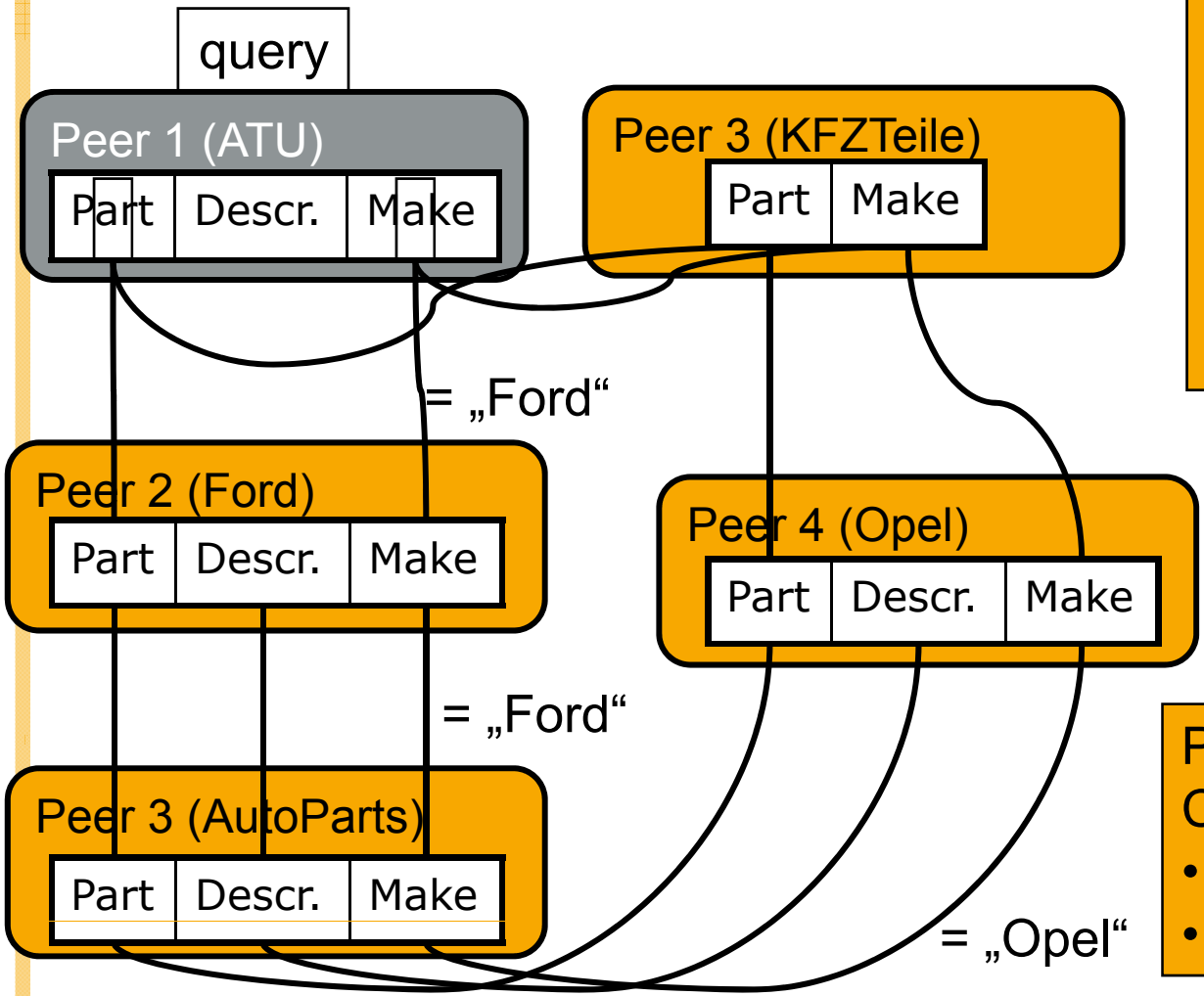
Summary

- Step 1: Schema Matching
  - Similarity Measure
  - Combination of methods
- Step 2: Duplicate Detection
  - Similarity Measure
  - Algorithm
  - Data Model
- Step 3: Data Fusion
  - Relational Operators
  - Conflict Resolution
  - Visualization of Semantics and Overlap

VIQTOR: Quality Annotations



PDMS: Incomplete and Selective Mappings



Problem:
Cumulated selections
- implicit in schemata
- explicit in mappings
- Point selections and range selections

Problem:
Cumulated projections
- in schemata
- in mappings