Australian National University

COMP3430 / COMP8430
Data wrangling

Lecture 24: Data fusion
(Lecturer: Peter Christen)

Based on slides by Xin Luna Dong (Amazon) and Felix Naumann (HPI Potsdam), VLDB tutorial (2009)

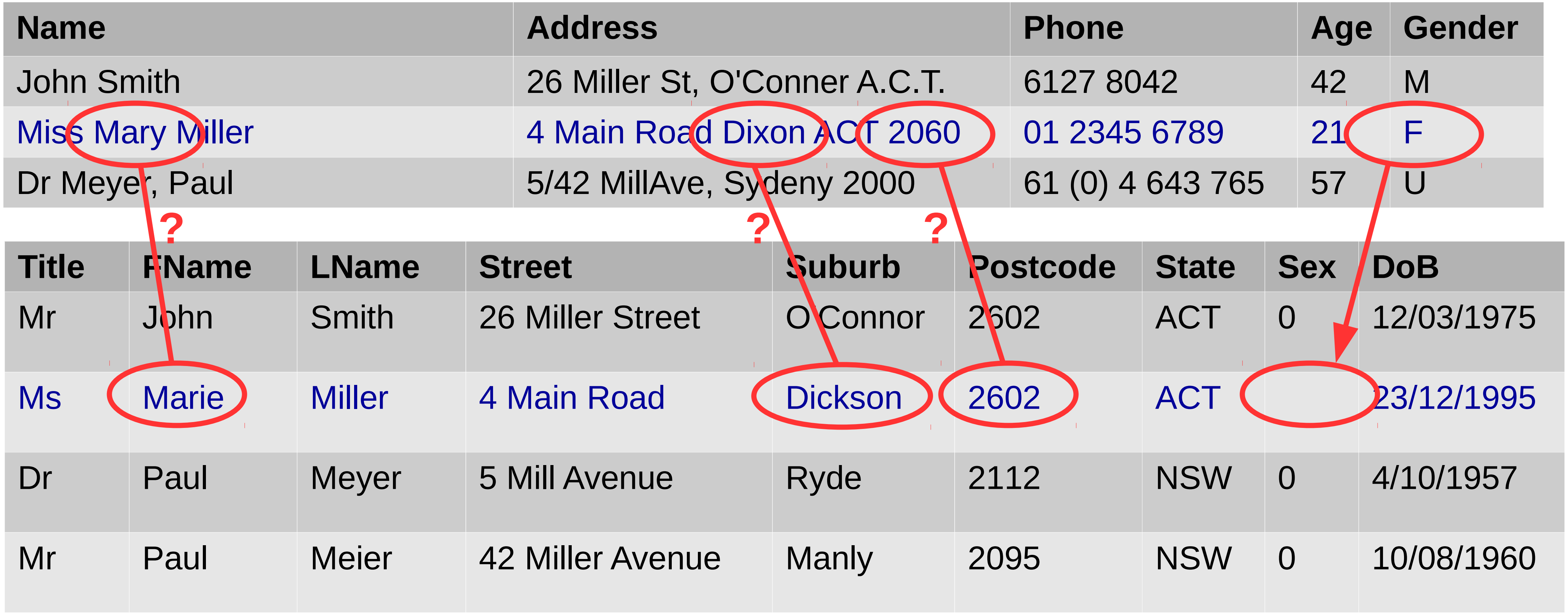
Lecture outline

- What is data fusion
- Resolving conflicts
- Resolution strategies, functions and operators

What is data fusion?

- Given a set of two or more records that have been classified to refer to the same entity, create a single record (representation) by resolving conflicting data values
- Various difficulties, including
  - Missing values in some source attributes
  - Contradicting attribute values
  - Uncertainty in the actual source values
  - Use of metadata (such as confidence in data sources, recency of data, and accuracy of data)
  - Implementation of fusion into database and data warehouse systems

Data fusion example



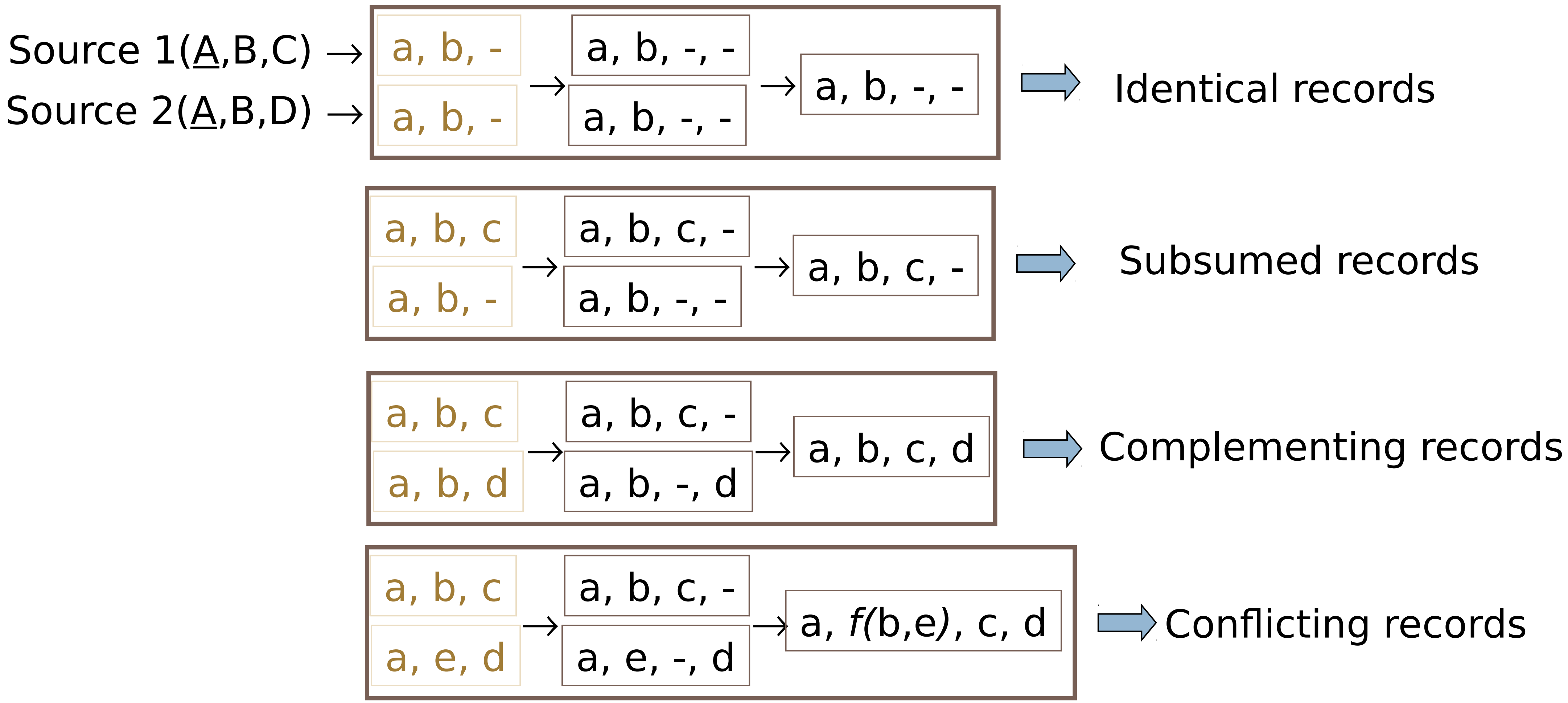
Name	Address	Phone	Age	Gender
John Smith	26 Miller St, O'Conner A.C.T.	6127 8042	42	M
Miss Mary Miller	4 Main Road Dixon ACT 2060	01 2345 6789	21	F
Dr Meyer, Paul	5/42 MillAve, Sydeny 2000	61 (0) 4 643 765	57	U

Title	FName	LName	Street	Suburb	Postcode	State	Sex	DoB
Mr	John	Smith	26 Miller Street	O'Connor	2602	ACT	0	12/03/1975
Ms	Marie	Miller	4 Main Road	Dickson	2602	ACT		23/12/1995
Dr	Paul	Meyer	5 Mill Avenue	Ryde	2112	NSW	0	4/10/1957
Mr	Paul	Meier	42 Miller Avenue	Manly	2095	NSW	0	10/08/1960

Three main tasks of data integration

- Schema mapping and matching
  - Identify which attributes or attribute sets across database tables contain the same type of information
- Record linkage / data matching / entity resolution
  - Identify which records in one or more databases correspond to the same real-world entity (person, business, product, etc.)
  - A special case is deduplication (or duplicate detection) in a single database
- **Data fusion**
  - Merge pairs or groups of records that correspond to the same entity into one clean, up-to-date, and consistent record that represents the entity

Data fusion goals



Source 1(A,B,C) →

Source 2(A,B,D) →

a, b, -
a, b, -
→
a, b, -, -
a, b, -, -
→
a, b, -, -

Identical records

a, b, c
a, b, -
→
a, b, c, -
a, b, -, -
→
a, b, c, -

Subsumed records

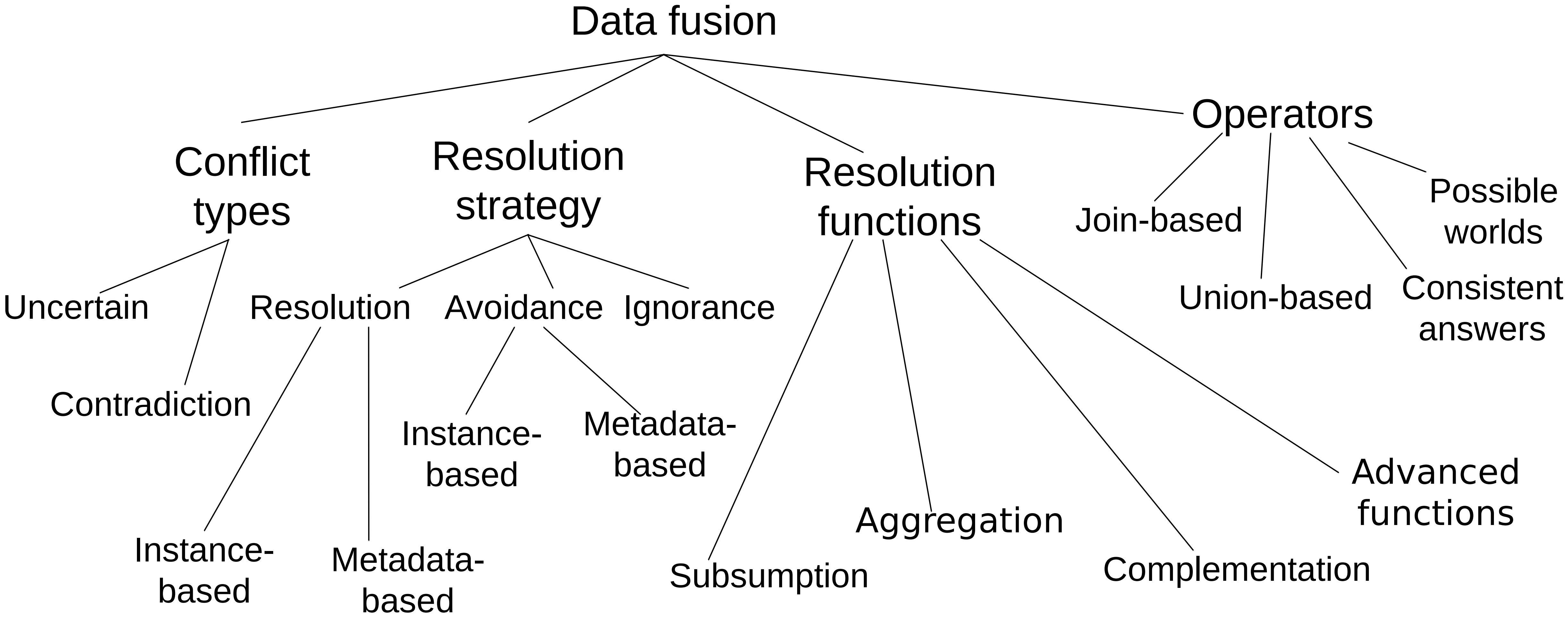
a, b, c
a, b, d
→
a, b, c, -
a, b, -, d
→
a, b, c, d

Complementing records

a, b, c
a, e, d
→
a, b, c, -
a, e, -, d
→
a, *f*(b,e), c, d

Conflicting records

The field of data fusion



- Data fusion
  - Conflict types
    - Uncertain
    - Contradiction
  - Resolution strategy
    - Resolution
      - Instance-based
      - Metadata-based
    - Avoidance
      - Instance-based
      - Metadata-based
    - Ignorance
  - Resolution functions
    - Subsumption
    - Aggregation
    - Complementation
    - Advanced functions
  - Operators
    - Join-based
    - Union-based
    - Consistent answers
    - Possible worlds

Conflict types: Uncertainty versus contradiction

- Uncertainty: Missing values versus non-missing value
- Contradiction: Two different non-missing values

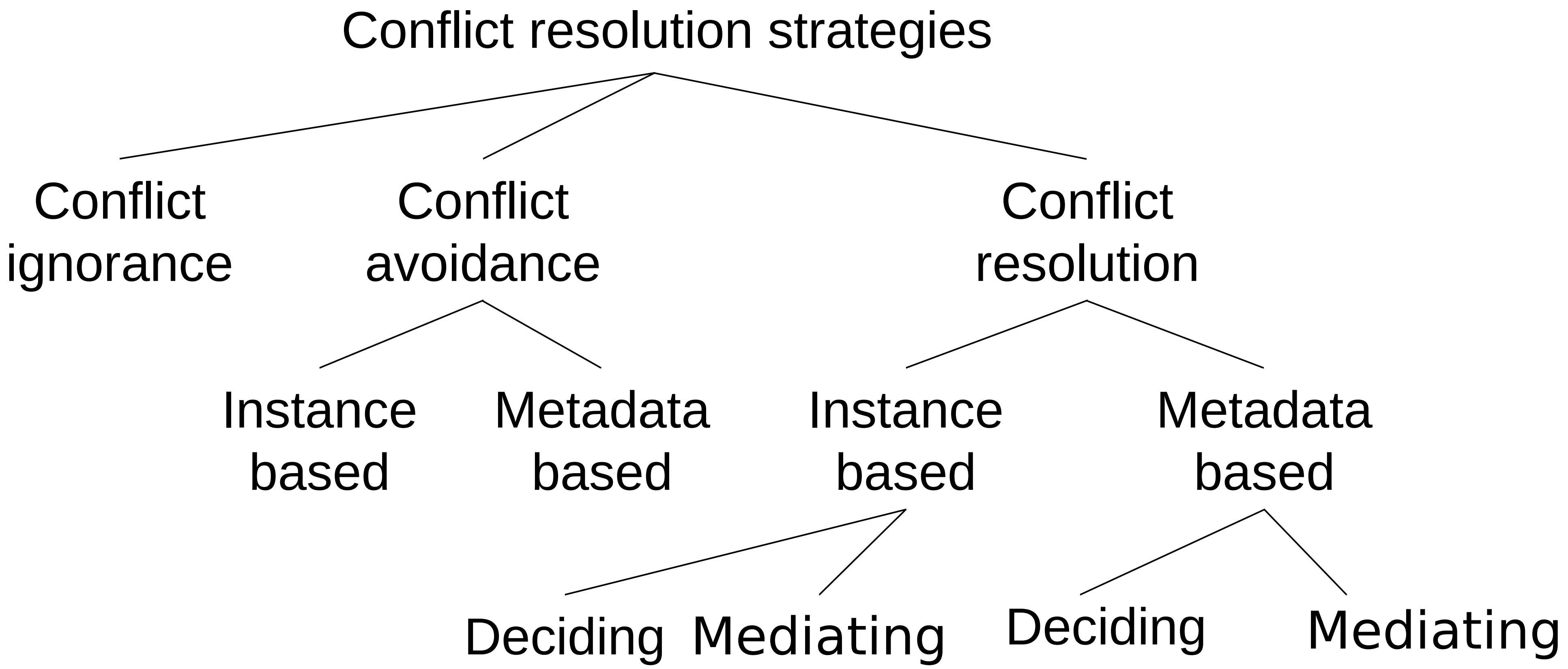
- Semantics of 'missing'

1) *unknown* – There is a value, but we don't know it (for example, an unknown date of birth)

2) *not applicable* – There is no meaningful value (for example, spouse for singles)

3) *withheld* – There is a value, but we are not authorised to see it (for example a private telephone number or bank account number)

Classification of conflict resolution strategies (1)



Conflict resolution strategies

Conflict ignorance

Conflict avoidance

Instance based

Metadata based

Conflict resolution

Instance based

Deciding

Mediating

Metadata based

Deciding

Mediating

Classification of conflict resolution strategies (2)

Strategy	Classification	Short Description
PASS IT ON	ignoring	Escalate conflict to user or application
CONSIDER ALL POSSIBILITIES	ignoring	Create all possible value combinations
TAKE THE INFORMATION	avoiding / instance based	Prefer values over null values
NO GOSSIPING	avoiding / instance based	Return only consistent tuples
TRUST YOUR FRIENDS	avoiding / metadata based	Take the value of a preferred source
CRY WITH THE WOLVES	resolution / instance based / deciding	Take the most often occurring value
ROLL THE DICE	resolution / instance based / deciding	Take a random value
MEET IN THE MIDDLE	resolution / instance based / mediating	Take an average value
KEEP UP TO DATE	resolution / metadata based / deciding	Take the most recent value

(based on Bleiholder and Naumann, ACM Computing Surveys, 2009)

Conflict resolution functions

Function	Description	Examples
Min, Max, Sum, Count, Avg	Standard aggregation	Number of children, salary, height
Random	Random choice	Shoe size
Longest, Shortest	Longest or shortest value	First name
Choose(source)	Value from a particular source	DoB (source 1), salary (source 2)
ChooseDepending(val, attr)	Value depends on value chosen in other attribute	City and postcode, e-mail and employer
Vote	Majority decision	Movie or wine rating
Coalesce	First non-null value	First name
Group, Concat	Group or concatenate all values	Book reviews
MostRecent	Most recent (up-to-date) value	Address
MostAbstract, MostSpecific, CommonAncestor	Use a taxonomy / ontology	Location
Escalate	Export conflicting values	Gender
…	…	…

Operators

- Identical records: UNION and OUTER UNION
- Subsumed records (uncertainty): MINIMUM UNION
- Complementing records (uncertainty): COMPLEMENT UNION and MERGE

- Conflicting records (contradiction)
  - Relational approaches: Match, Group, Fuse, …
- Other approaches
  - Possible worlds, probabilistic answers, consistent answers

Minimum union

- Union: Elimination of exact duplicates
- Minimum Union: Elimination of subsumed records

- A record r1 subsumes a record r2 if it has the same schema, has less missing values, and coincides in all non-missing values

A	B	C
a	b	c
e	f	g
m	n	o

$\uplus$

A	B	D
a	b	⊥
e	f	h
m	p	⊥

=

A	B	C	D
a	b	c	⊥
a	b	⊥	⊥
e	f	g	⊥
e	f	⊥	h
m	n	o	⊥
m	p	⊥	⊥

R↓

A	B	C	D
a	b	c	⊥
e	f	g	⊥
e	f	⊥	h
m	n	o	⊥
m	p	⊥	⊥

Complement union

- Elimination of complementing records
  - Outer union
  - Complementation
  - No known SQL rewriting
- Includes duplicate removal and subsumption

- A record r1 complements a record r2 if it has the same schema and coincides in all non-missing values

A	B	C
a	b	c
e	f	g
m	n	o

⊎

A	B	D
a	b	⊥
e	f	h
m	p	⊥

=

A	B	C	D
a	b	c	⊥
a	b	⊥	⊥
e	f	g	⊥
e	f	⊥	h
m	n	o	⊥
m	p	⊥	⊥

R⇅

A	B	C	D
a	b	c	⊥
e	f	g	h
m	n	o	⊥
m	p	⊥	⊥

Full disjunction

- Represents all possible combinations of source records
  - Full outer join on all common attributes
  - All combinations for more than two sources
  - Minimum union over results
  - Combines complementing records

A	B	C
a	b	c
e	f	g
k	⊥	o
k	m	⊥

|⋈|

A	B	D
a	b	⊥
e	f	h
m	p	⊥
k	q	r

=

A	B	C	D
a	b	c	⊥
e	f	g	h
m	p	⊥	⊥
k	⊥	o	⊥
k	m	⊥	⊥
k	q	⊥	r

R↓

A	B	C	D
a	b	c	⊥
e	f	g	h
m	p	⊥	⊥
k	m	o	⊥
k	q	⊥	r

Other approaches for operators

- Consistent Query Answering
  - Avoid conflicts and report only certain records (those that do not have conflicts)
- “Possible worlds” models
  - Build all possible solutions, annotated with likelihood (Yes/No/Maybe, or a probability value)
- *Probabilistic databases*
  - Extend relational algebra to produce probabilities
  - Extend query language to query and export probabilities

Some practical aspects (1)

- Different data sources are likely of different data quality, and so we should trust records from accurate sources more

- Real world data are dynamic, and true values can change over time
  - Therefore more recent data might be more accurate and useful

- Values might be copied from one data source to another
  - Including errors!

Some practical aspects (2)

- Therefore, in practice, we need to consider:

  (1) **Accuracy** of data sources

  (2) **Freshness** (timeliness) of data sources

  (3) **Dependencies** between data sources

Open problems in data fusion

- The accuracy of fusion
  - At the attribute and record level (requires truth data for evaluation)
- The efficiency of fusion
  - For example incremental fusion as new data arrives (real-time fusion)
- The usability of fusion
  - Adaptive to the needs of a user and/or application
  - Legal requirement with regard to data provenance and data lineage
- Interaction between data fusion and other data integration tasks
  - Such as the *Swoosh* entity resolution system as developed by Stanford University