Uncertainty in Webdam

Pierre Senellart

DBWeb

Webdam

Webdone, 30 September 2013

Outline

Uncertain data

Numerous sources of uncertain data:

- Measurement errors
- Data integration from contradicting sources
- Imprecise mappings between heterogeneous schemata
- Imprecise automatic process (information extraction, natural language processing, graph mining, etc.)
- Imperfect human judgment
- Lies, opinions, rumors

Managing uncertainty

Objective

Not to pretend this imprecision does not exist, and manage it as rigorously as possible throughout a long, automatic and human, potentially complex, process

Especially:

- Represent different forms of uncertainty
- Probabilities are used to measure uncertainty in the data
- Query data and retrieve uncertain results
- Allow adding, deleting, modifying data in an uncertain way

Managing uncertainty

Objective

Not to pretend this imprecision does not exist, and manage it as rigorously as possible throughout a long, automatic and human, potentially complex, process

Especially:

- Represent different forms of uncertainty
- Probabilities are used to measure uncertainty in the data
- Query data and retrieve uncertain results
- Allow adding, deleting, modifying data in an uncertain way

Why probabilities?

- Not the only option: fuzzy set theory [Galindo et al., 2005], Dempster-Shafer theory [Zadeh, 1986]
- Mathematically rich theory, nice semantics with respect to traditional database operations (e.g., joins)
- Some applications already generate probabilities (e.g., statistical information extraction or natural language probabilities)
- Naturally arising in case of conflicting information, based on the trust in the sources
- In other cases, we "cheat" and pretend that (normalized) confidence scores are probabilities: see this as a first-order approximation

Webdam Achievements on Uncertainty

- Resolving contradictions in facts derived by Webdamlog through sampling possible worlds (see Daniel's talk) [Abiteboul et al., 2012c]
- (Pseudo-)probabilistic model for matching two large ontologies (see Fabian's talk and Antoine's demo) [Suchanek et al., 2011, Oita et al., 2012]
- Deriving probabilistic generators from XML corpora, and an auto-completion editor application (see Yael's talk and demo) [Abiteboul et al., 2012a,b]
- Probabilistic models for crowd mining (with MoDaS) [Amsterdamer et al., 2013a,b]
- Models, algorithms, tools, for probabilistic XML querying and managing (see further and Asma's demo)
- ...

Outline

Why probabilistic XML?

- Different typical querying languages: SQL and conjunctive queries vs XPath and tree-pattern queries (possibly with joins)
- Cases where a tree-like model might be appropriate:
  - No schema or few constraints on the schema
  - Independent modules annotating freely a content warehouse
  - Inherently tree-like data (e.g., mailing lists, parse trees) with naturally occurring queries involving the descendant axis

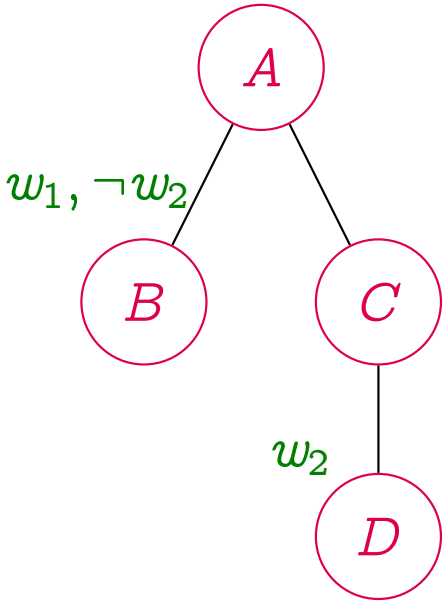
Remark

Some results can be transferred from one model to the other. In other cases, connection much trickier! [Amarilli and Senellart, 2013]

joint work with 

A general discrete Probabilistic XML Model

[Abiteboul et al., 2009]



semantics →

$p_1 = 0.06$

$p_2 = 0.70$

$p_3 = 0.24$

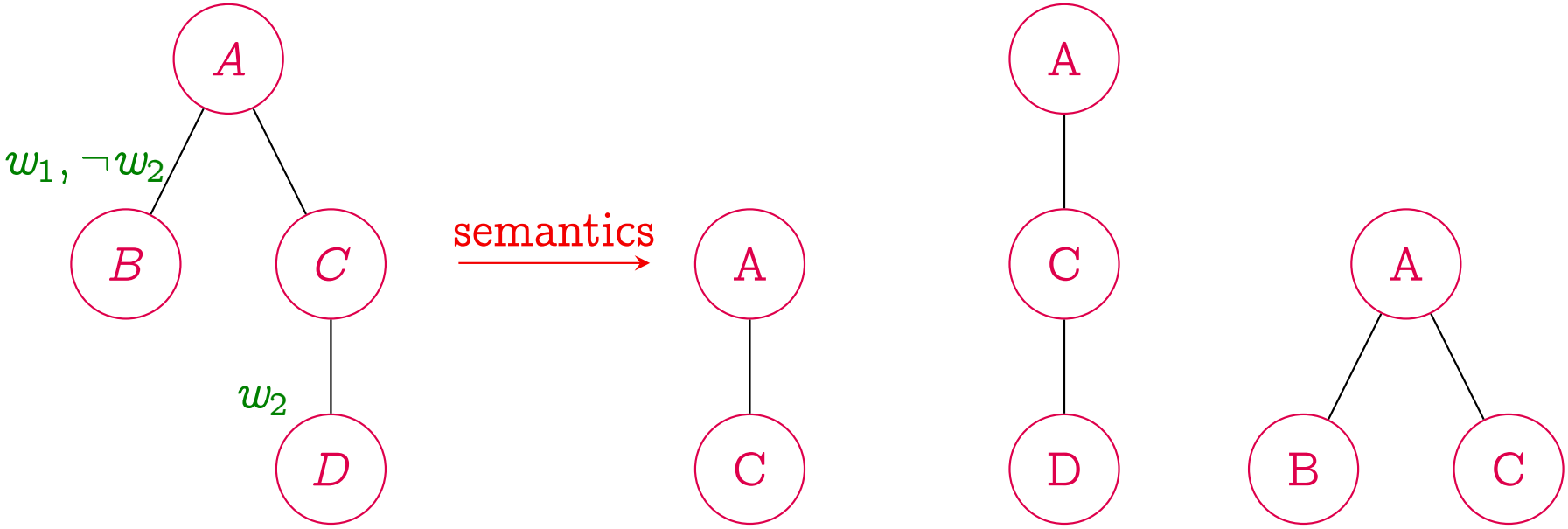
Event	Prob.
w_1	0.8
w_2	0.7

- Expresses arbitrarily complex dependencies
- Analogous to probabilistic c-tables

joint work with

A general discrete Probabilistic XML Model

[Abiteboul et al., 2009]



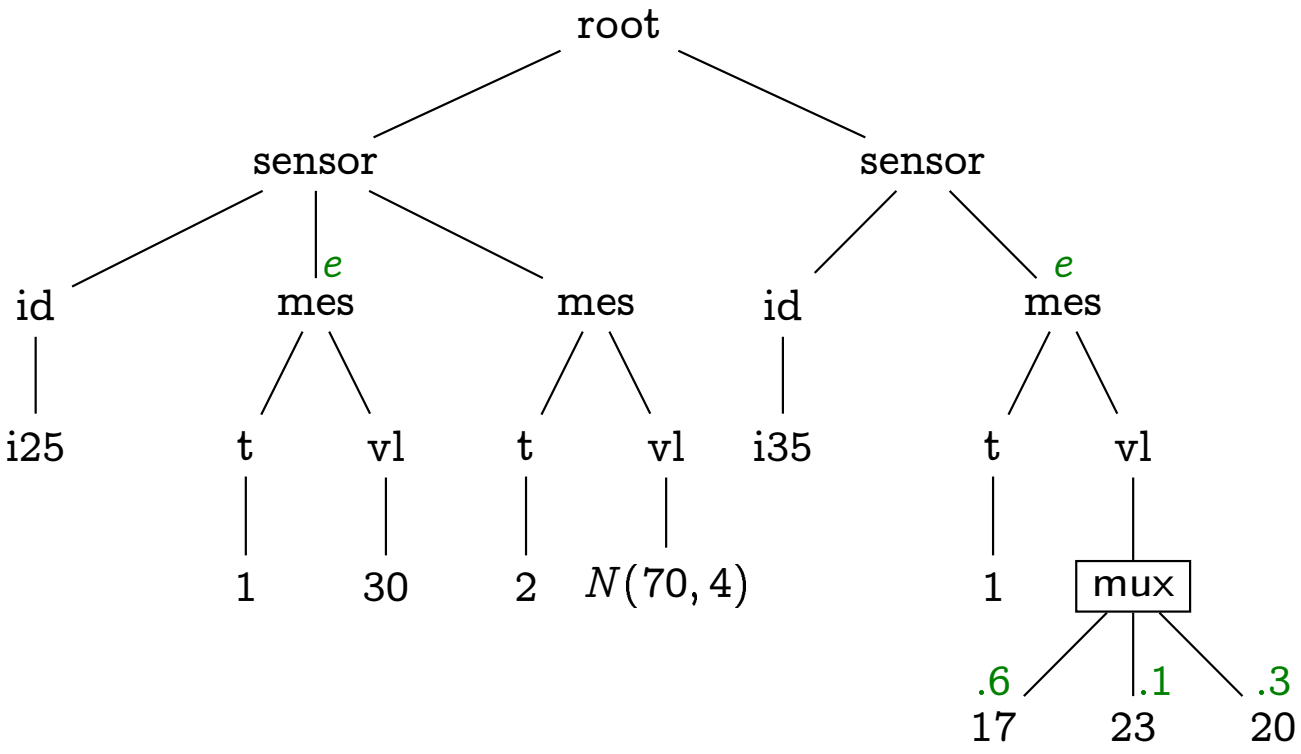
Event	Prob.
w_1	0.8
w_2	0.7

- Expresses arbitrarily complex dependencies
- Analogous to probabilistic c-tables

joint work with

TELECOM ParisTech

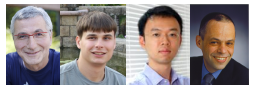
Continuous distributions [Abiteboul et al., 2011]



- e: event "it did not rain" at time 1
- mux: mutually exclusive options
- $N(70, 4)$: normal distribution

- Two kinds of dependencies in discrete probabilistic XML: global (e) and local (mux)
- Add continuous leaves; non-obvious semantics issues

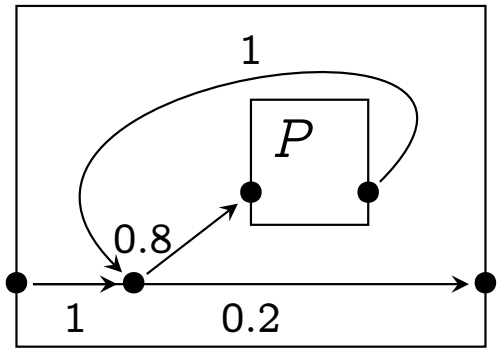
joint work with



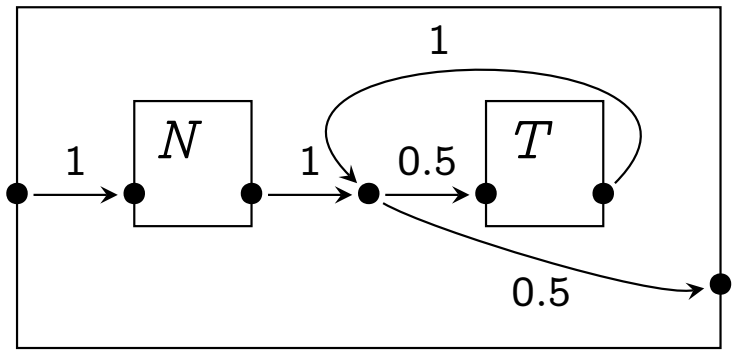
Recursive Markov chains [Benedikt et al., 2010]

```
<!ELEMENT directory (person*)>
<!ELEMENT person (name,phone*)>
```

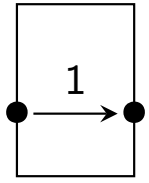
D: directory



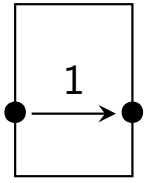
P: person



N: name



T: phone



- Probabilistic model that subsumes local discrete models
- Allows generating documents of unbounded width or depth

joint work with

Querying probabilistic XML

Semantics of a (Boolean) query = probability the query is true:

1. Generate all possible worlds of a given probabilistic document

2. In each world, evaluate the query
3. Add up the probabilities of the worlds that make the query true

EXPTIME algorithm! Can we do better, i.e., can we apply directly the algorithm on the probabilistic document?

We shall talk about data complexity of query answering.

Querying probabilistic XML

Semantics of a (Boolean) query = probability the query is true:

1. Generate all possible worlds of a given probabilistic document (possibly exponentially many)
2. In each world, evaluate the query
3. Add up the probabilities of the worlds that make the query true

EXPTIME algorithm! Can we do better, i.e., can we apply directly the algorithm on the probabilistic document?

We shall talk about data complexity of query answering.

Previously known results
[Kimelfeld et al., 2009, Cohen et al., 2009]

- Tree-pattern queries are evaluable in linear time over local dependencies with a bottom-up, dynamic programming algorithm
- Indeed, any monadic second-order query is tractable over local dependencies
- Even trivial queries are #P-hard over global dependencies
- Monte-Carlo sampling works
- Multiplicative approximation is also tractable (existence of a FPRAS)

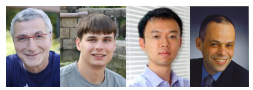
Aggregation queries [Abiteboul et al., 2010]

Aggregate Queries: sum, count, avg, countd, min, max, etc.
Distributions? Possible values? Expected value?

- Computing expected values of sum and count tractable with global dependencies; everything else intractable
- Computing expected values of every of these aggregate functions tractable with local dependencies
- Computing distributions and possible values tractable for count, min, max, intractable for the others

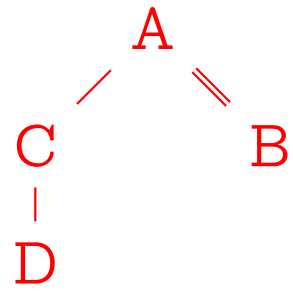
Continuous distributions do not add any more complexity!

joint work with



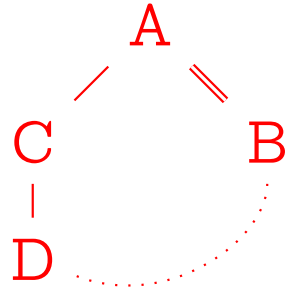
Queries with joins are hard [Kharlamov et al., 2011]

Tree-pattern queries (TP) `/A[C/D]//B`



Tree-pattern queries with joins (TPJ)

```
for $x in $doc/A/C/D
return $doc/A//B[.=$x]
```



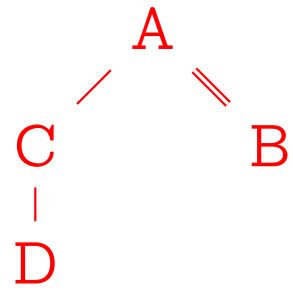
Over local dependencies, dichotomy for queries with a single join:

- If equivalent to a join-free query, linear-time
- Otherwise, #P-hard

joint work with

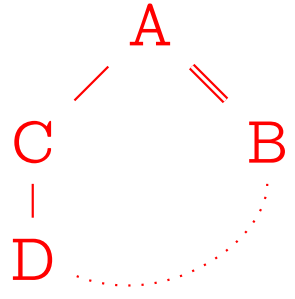
Queries with joins are hard [Kharlamov et al., 2011]

Tree-pattern queries (TP) `/A[C/D]//B`



Tree-pattern queries with joins (TPJ)

```
for $x in $doc/A/C/D
return $doc/A//B[.=$x]
```



Over local dependencies, dichotomy for queries with a single join:

- If equivalent to a join-free query, linear-time
- Otherwise, #P-hard

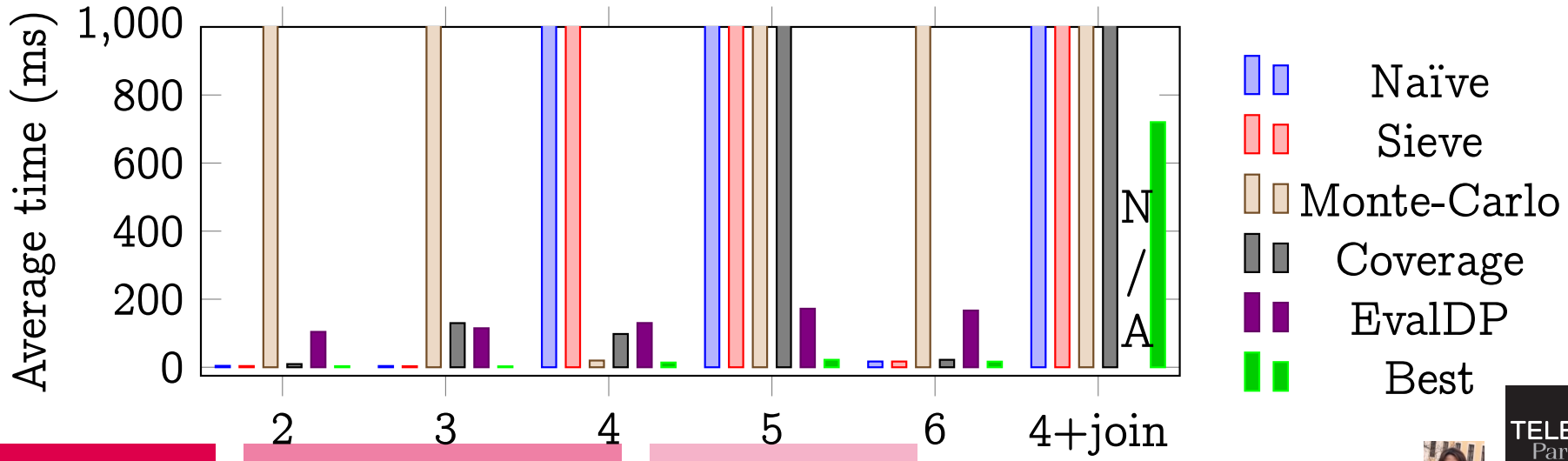
joint work with

Rely on approximations... when needed

[Senellart and Souihli, 2011, Souihli, 2011, Souihli and Senellart, 2012, 2013]

- Monte-Carlo is very good at approximating high probabilities
- Sometimes the structure of a query makes the probability of a query easy to evaluate
- For small formulas, naïve evaluation techniques good enough
- Refined approximation methods best when everything else fails

⇒ Use a cost model to decide which evaluation algorithm to use!



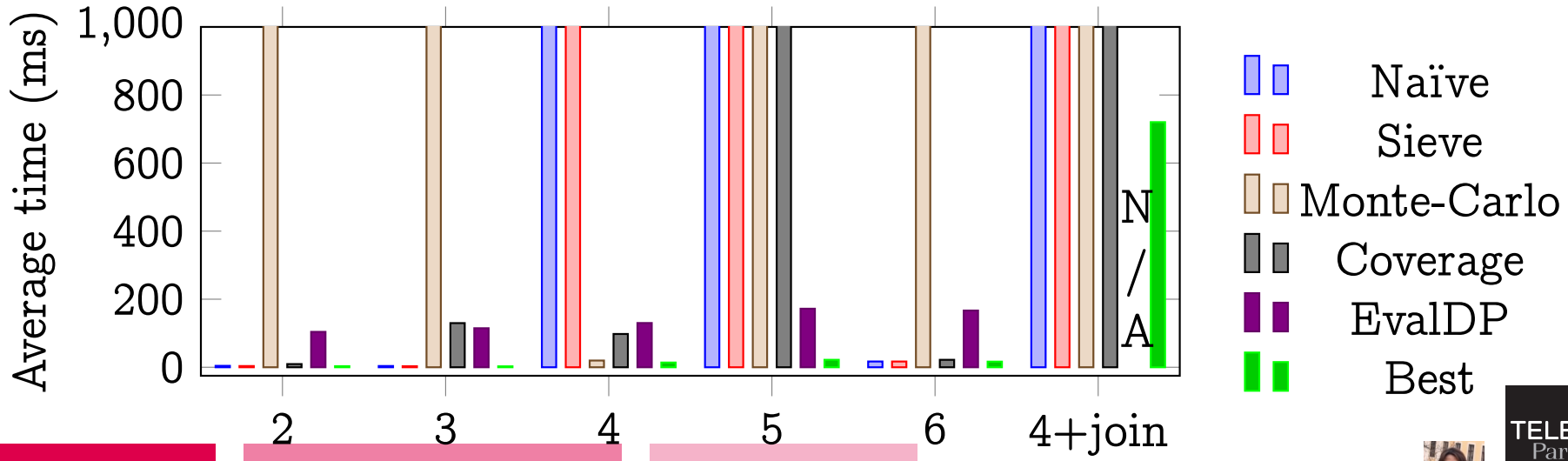
joint work with

Rely on approximations... when needed

[Senellart and Souihli, 2011, Souihli, 2011, Souihli and Senellart, 2012, 2013]

- Monte-Carlo is very good at approximating high probabilities
- Sometimes the structure of a query makes the probability of a query easy to evaluate
- For small formulas, naïve evaluation techniques good enough
- Refined approximation methods best when everything else fails

$\Rightarrow$ Use a cost model to decide which evaluation algorithm to use!



joint work with

TELECOM ParisTech

The complexity of updates

[Abiteboul et al., 2009, Kharlamov et al., 2010]

Updates defined by a query (cf. XUpdate, XQuery Update).
Semantics: for all matches of a query, insert or delete a node in the tree at a place located by the query.

Results

- Most updates are intractable with local dependencies: the result of an update can require an exponentially larger representation size
- Insertions with a for-each-match semantics are tractable with arbitrary dependencies; deletions are intractable
- Some insert-if-there-is-a-match operations tractable for local dependencies but not for arbitrary dependencies

joint work with

Other Probabilistic XML Works in Webdam

- A survey of probabilistic XML [Kimelfeld and Senellart, 2013]
- Query answering using views in probabilistic XML [Cautis and Kharlamov, 2011, 2012]
- Mining probabilistic XML data [Kharlamov and Senellart, 2011]

joint work with

Outline

Intensional data is everywhere

Lots of data sources can be seen as intensional: accessing all the data in the source (in extension) is impossible or very costly, but it is possible to access the data through views, with some access constraints, associated with some access cost.

- Indexes over regular data sources
- Deep Web sources: Web forms, Web services
- The Web or social networks as partial graphs that can be expanded by crawling
- Outcome of complex automated processes: information extraction, natural language analysis, machine learning, ontology matching
- Crowd data: (very) partial views of the world
- etc.

Introducing UnSAID

- Uncertainty and Structure in the Access to Intensional Data
- Jointly deal with Uncertainty, Structure, and the fact that access to data is limited and has a cost, to solve a user's knowledge need
- Lazy evaluation whenever possible
- Evolving probabilistic, structured view of the current knowledge of the world
- Knowledge acquisition plan (recursive, dynamic, adaptive) that minimizes access cost, and provides probabilistic guarantees

Merci.

2009–2013

Serge Abiteboul, Benny Kimelfeld, Yehoshua Sagiv, and Pierre Senellart. On the expressiveness of probabilistic XML models. *VLDB Journal*, 18(5):1041–1064, October 2009.

Serge Abiteboul, T-H. Hubert Chan, Evgeny Kharlamov, Werner Nutt, and Pierre Senellart. Aggregate queries for discrete and continuous probabilistic XML. In *Proc. ICDT*, pages 50–61, Lausanne, Switzerland, March 2010.

Serge Abiteboul, T-H. Hubert Chan, Evgeny Kharlamov, Werner Nutt, and Pierre Senellart. Capturing continuous data and answering aggregate queries in probabilistic XML. *ACM Transactions on Database Systems*, 36(4), 2011.

Serge Abiteboul, Yael Amsterdamer, Daniel Deutch, Tova Milo, and Pierre Senellart. Finding optimal probabilistic generators for XML collections. In *Proc. ICDT*, pages 127–139, Berlin, Germany, March 2012a.

Serge Abiteboul, Yael Amsterdamer, Tova Milo, and Pierre Senellart. Auto-completion learning for XML. In *Proc. SIGMOD*, pages 669–672, Scottsdale, USA, May 2012b. Demonstration.

Serge Abiteboul, Meghyn Bienvenu, and Daniel Deutch. Deduction in the presence of distribution and contradictions. In *WebDB*, 2012c.

Antoine Amarilli and Pierre Senellart. On the connections between relational and XML probabilistic data models. In *Proc. BNCOD*, pages 121–134, Oxford, United Kingdom, July 2013.

Yael Amsterdamer, Yael Grossman, Tova Milo, and Pierre Senellart. Crowd mining. In *Proc. SIGMOD*, pages 241–252, New York, USA, June 2013a.

Yael Amsterdamer, Yael Grossman, Tova Milo, and Pierre Senellart. Crowd miner: Mining association rules from the crowd. In *Proc. VLDB*, Riva del Garda, Italy, August 2013b. Demonstration.

Michael Benedikt, Evgeny Kharlamov, Dan Olteanu, and Pierre Senellart. Probabilistic XML via Markov chains. *Proceedings of the VLDB Endowment*, 3(1):770–781, September 2010. Presented at the VLDB 2010 conference, Singapore.

Bogdan Cautis and Evgeny Kharlamov. Challenges for view-based query answering over probabilistic XML. In *AMW*, 2011.

Bogdan Cautis and Evgeny Kharlamov. Answering queries using views over probabilistic xml: Complexity and tractability. *Proceedings of the VLDB Endowement*, 5(11), 2012.

Sara Cohen, Benny Kimelfeld, and Yehoshua Sagiv. Running tree automata on probabilistic XML. In *PODS*, 2009.

José Galindo, Angelica Urrutia, and Mario Piattini. *Fuzzy Databases: Modeling, Design And Implementation*. IGI Global, 2005.

Evgeny Kharlamov and Pierre Senellart. Modeling, querying, and mining uncertain XML data. In Andrea Tagarelli, editor, *XML Data Mining: Models, Methods, and Applications*, pages 29–52. IGI Global, November 2011.

Evgeny Kharlamov, Werner Nutt, and Pierre Senellart. Updating probabilistic XML. In *Proc. Updates in XML*, Lausanne, Switzerland, March 2010.

Evgeny Kharlamov, Werner Nutt, and Pierre Senellart. Value joins are expensive over (probabilistic) XML. In *Proc. LID*, pages 41–48, Uppsala, Sweden, March 2011.

Benny Kimelfeld and Pierre Senellart. Probabilistic XML: Models and complexity. In Zongmin Ma and Li Yan, editors, *Advances in Probabilistic Databases for Uncertain Information Management*, pages 39–66. Springer-Verlag, May 2013.

Benny Kimelfeld, Yuri Kosharovsky, and Yehoshua Sagiv. Query evaluation over probabilistic XML. *VLDB Journal*, 18(5), 2009.

Marilena Oita, Antoine Amarilli, and Pierre Senellart. Cross-fertilizing deep Web analysis and ontology enrichment. In *Proc. VLDS*, pages 5–8, Istanbul, Turkey, August 2012. Vision article.

Pierre Senellart and Asma Souihli. ProApproX: A lightweight approximation query processor over probabilistic trees. In *Proc. SIGMOD*, pages 1295–1298, Athens, Greece, June 2011. Demonstration.

Asma Souihli. Efficient query evaluation over probabilistic XML with long-distance dependencies. In *EDBT/ICDT PhD Workshop*, 2011.

Asma Souihli and Pierre Senellart. Demonstrating ProApproX 2.0: A predictive query engine for probabilistic XML. In *Proc. CIKM*, Maui, Hawaii, USA, October 2012. Demonstration.

Asma Souihli and Pierre Senellart. Optimizing approximations of DNF query lineage in probabilistic XML. In *Proc. ICDE*, pages 721–732, Brisbane, Australia, April 2013.

Fabian M. Suchanek, Serge Abiteboul, and Pierre Senellart. PARIS: Probabilistic alignment of relations, instances, and schema. *Proceedings of the VLDB Endowment*, 5(3):157–168, December 2011. Presented at the VLDB 2012 conference, Istanbul, Turkey.

Lotfi A. Zadeh. A simple view of the Dempster-Shafer theory of evidence and its implication for the rule of combination. *AI Magazine*, 7(2), 1986.