

# Querying and Updating Probabilistic Information in XML

Serge Abiteboul   Pierre Senellart



*GEMO* Seminar  
March 10th, 2006

# The Hidden Web

## Definition (Hidden Web)

The set of webpages (which may or may not be dynamically generated) not accessible from the **hyperlinked structure** of the World Wide Web.

# The Hidden Web

## Definition (Hidden Web)

The set of webpages (which may or may not be dynamically generated) not accessible from the **hyperlinked structure** of the World Wide Web.

Size estimate (2001) : 500 times larger than the **surface Web**.

# The Hidden Web

## Definition (Hidden Web)

The set of webpages (which may or may not be dynamically generated) not accessible from the **hyperlinked structure** of the World Wide Web.

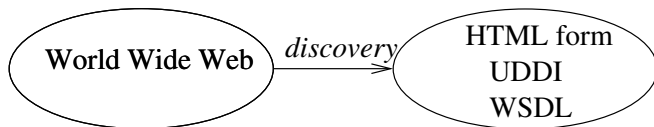
Size estimate (2001) : 500 times larger than the **surface Web**.

How to understand it and benefit from its content?

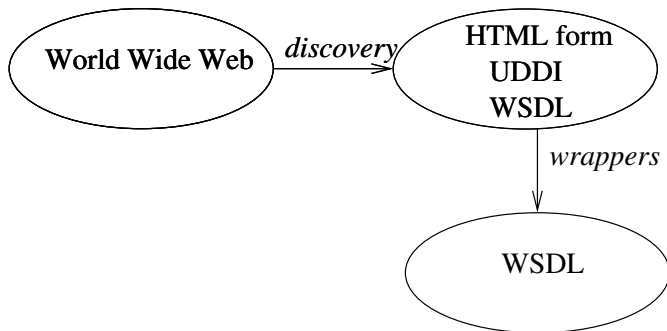
# Semantic Interpretation of the Hidden Web

World Wide Web

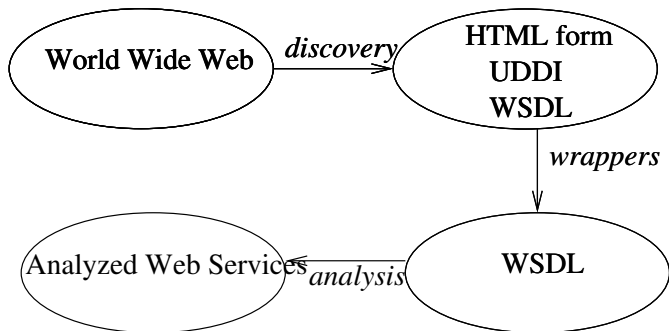
# Semantic Interpretation of the Hidden Web



# Semantic Interpretation of the Hidden Web

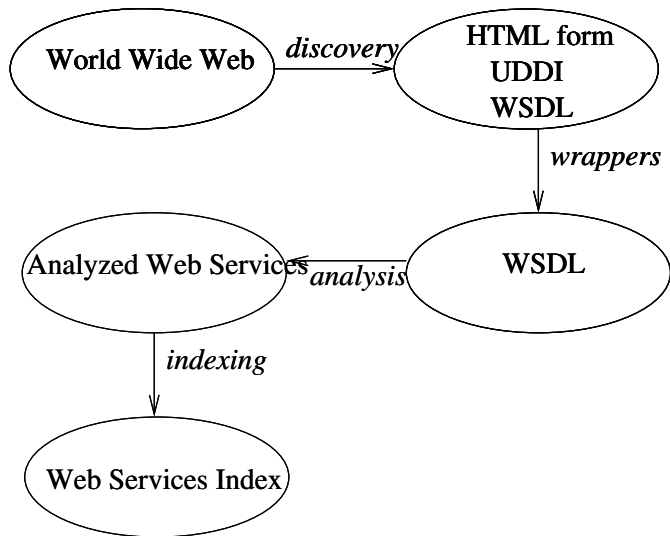


# Semantic Interpretation of the Hidden Web

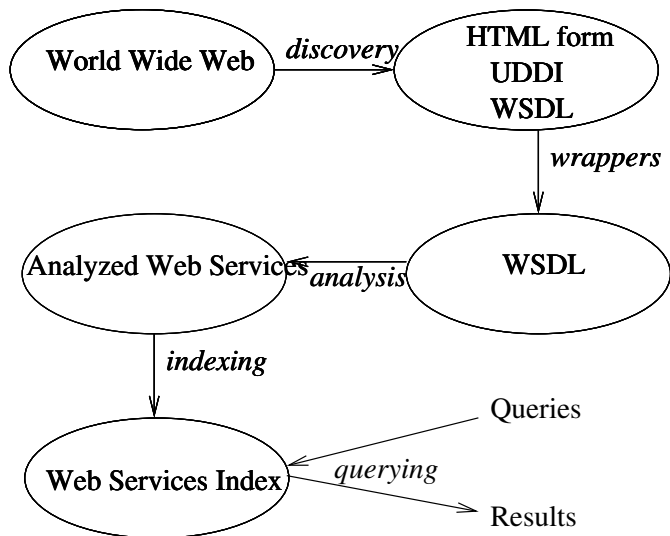




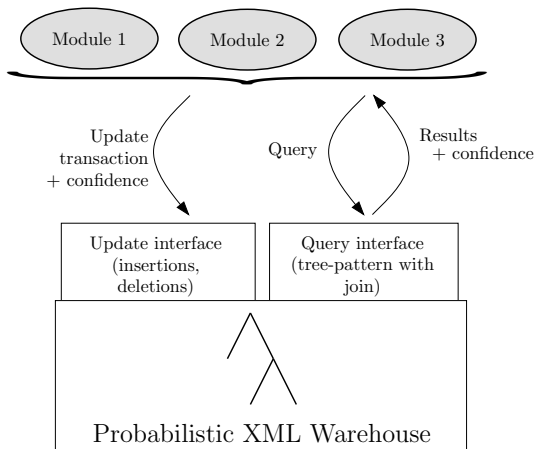
# Semantic Interpretation of the Hidden Web



# Semantic Interpretation of the Hidden Web



# A Probabilistic XML Warehouse



# Outline

1 Introduction

2 **Framework**

- Data Trees
- Queries
- Updates

3 Possible Worlds Model

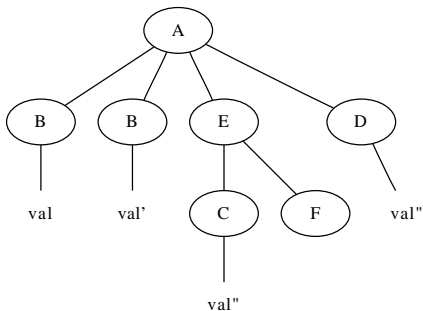
4 Simple Probabilistic Model

5 Fuzzy Tree Model

6 Conclusion

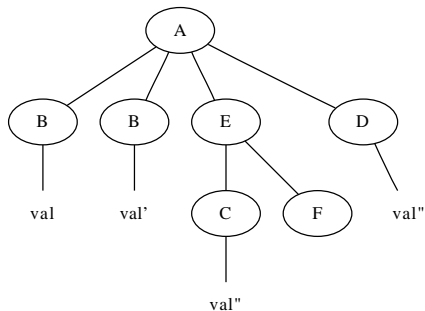
# Data Trees

- Finite, **unordered**, trees.



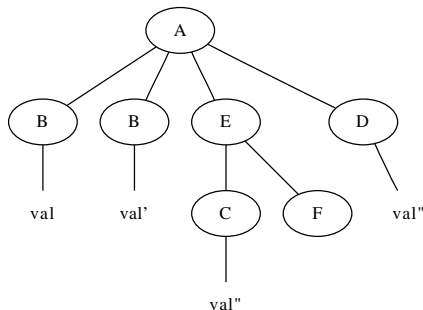
# Data Trees

- Finite, **unordered**, trees.
- No **attribute** nodes, no **mixed** content.



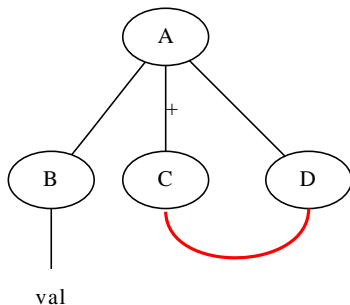
# Data Trees

- Finite, **unordered**, trees.
- No **attribute** nodes, no **mixed** content.
- **"Multiset"** children



# Tree-Pattern With Join Queries

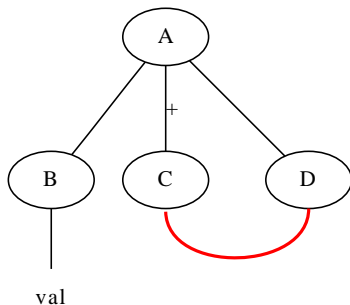
- Queries: **Tree-Pattern With Join** (TPWJ)





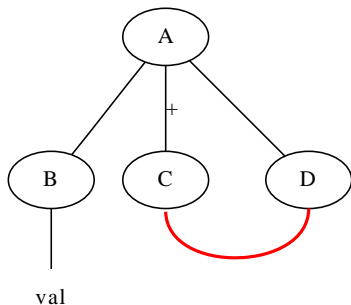
# Tree-Pattern With Join Queries

- Queries: **Tree-Pattern With Join** (TPWJ)
- Join: by value



# Tree-Pattern With Join Queries

- Queries: **Tree-Pattern With Join** (TPWJ)
- Join: by value
- Actually: any **positive** query



# Update Transactions

- **Set** of elementary operations:

# Update Transactions

- **Set** of elementary operations:
  - **Insertions** of subtrees

# Update Transactions

- **Set** of elementary operations:
  - **Insertions** of subtrees
  - **Deletions** of subtrees

# Update Transactions

- **Set** of elementary operations:
  - **Insertions** of subtrees
  - **Deletions** of subtrees
- TPWJ query + mapping, stating **where** to perform operations.

# Update Transactions

- **Set** of elementary operations:
  - **Insertions** of subtrees
  - **Deletions** of subtrees
- TPWJ query + mapping, stating **where** to perform operations.
- **Probabilistic update**: update + **confidence**

# Outline

- 1 Introduction
- 2 Framework
- 3 Possible Worlds Model**
  - Model
  - Queries
  - Updates
- 4 Simple Probabilistic Model
- 5 Fuzzy Tree Model
- 6 Conclusion

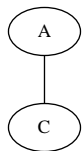


# PW Model

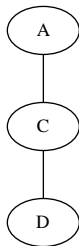
Set of **tree/probability pairs**, one for each **possible world**.

# PW Model

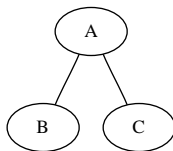
Set of **tree/probability pairs**, one for each **possible world**.



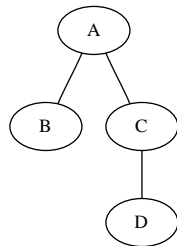
$$P = 0.06$$



$$P = 0.14$$



$$P = 0.24$$



$$P = 0.56$$

# Queries on PW sets

## Definition

If  $T = \{(t_i, p_i)\}$ , the result of query  $Q$  over the PW set  $T$  is the normalization of  $\{(t, p_i) | t \in Q(t_i)\}$

# Queries on PW sets

## Definition

If  $T = \{(t_i, p_i)\}$ , the result of query  $Q$  over the PW set  $T$  is the normalization of  $\{(t, p_i) | t \in Q(t_i)\}$

- **Normalization**: grouping of duplicates (summing probabilities)

# Queries on PW sets

## Definition

If  $T = \{(t_i, p_i)\}$ , the result of query  $Q$  over the PW set  $T$  is the normalization of  $\{(t, p_i) | t \in Q(t_i)\}$

- **Normalization**: grouping of duplicates (summing probabilities)
- $(t, p) \in Q(T)$  means “the probability that  $t$  matches  $T$  is  $p$ ”.

# Updates on PW sets

## Definition

The result of an update  $t$  with confidence  $c$  on a PW set  $T$  is the normalization of:

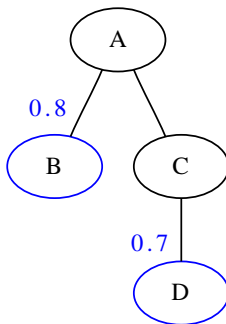
$$\begin{aligned} & \{(t, p) \in T \mid t \text{ is not selected by } Q\} \\ \cup & \{(\tau(t), p \cdot c) \mid t \text{ is selected by } Q\} \\ \cup & \{(t, p \cdot (1 - c)) \mid t \text{ is selected by } Q\} \end{aligned}$$

# Outline

- 1 Introduction
- 2 Framework
- 3 Possible Worlds Model
- 4 Simple Probabilistic Model**
  - Model and Possible Worlds Semantics
  - Queries
  - Incompleteness
- 5 Fuzzy Tree Model
- 6 Conclusion

# SP Trees

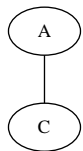
Data tree with **probability assigned to each node.**



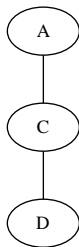


# PW Semantics of SP Trees

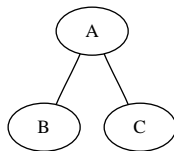
A node is assigned the probability  $p$ : means that the probability the node is **in the tree** if **its parent is in the tree** is  $p$ .



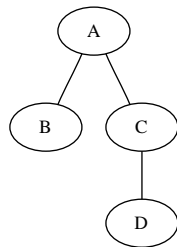
$$P = 0.06$$



$$P = 0.14$$



$$P = 0.24$$



$$P = 0.56$$

# Queries on SP Trees

## Definition

Queries on SP trees:

- Query **on underlying tree**.
- Probabilities: **multiplication of probabilities** of nodes of the mapping.

# Queries on SP Trees

## Definition

Queries on SP trees:

- Query **on underlying tree**.
- Probabilities: **multiplication of probabilities** of nodes of the mapping.

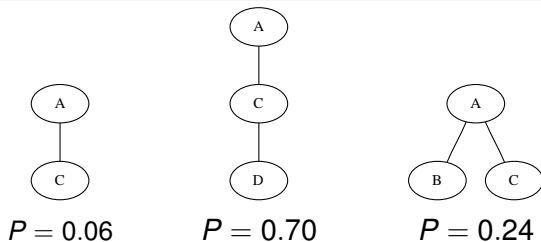
## Theorem

$$Q(\llbracket T \rrbracket) = Q(T)$$

# Incompleteness of SP Trees

## Theorem

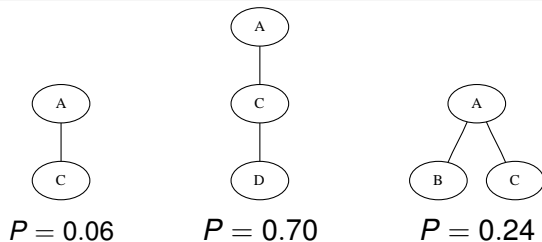
*The SP tree model is incomplete.*



# Incompleteness of SP Trees

## Theorem

*The SP tree model is incomplete.*



## Theorem

*SP trees are not closed under updates.*

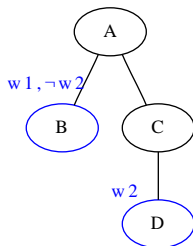
# Outline

- 1 Introduction
- 2 Framework
- 3 Possible Worlds Model
- 4 Simple Probabilistic Model
- 5 Fuzzy Tree Model**
  - Model and Possible Worlds Semantics
  - Queries
  - Updates
  - Implementation

- 6 Conclusion

# Fuzzy Trees

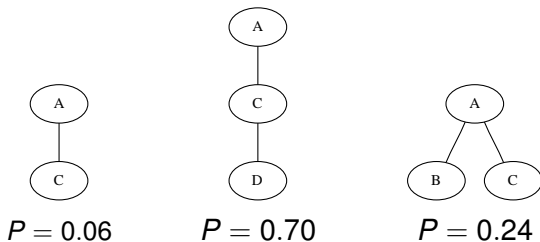
Data tree with **event conditions** (conjunction of probabilistic events or negations of probabilistic events) **assigned to each node**.



| Event | Proba. |
|-------|--------|
| w1    | 0.8    |
| w2    | 0.7    |

# PW Semantics of Fuzzy Trees

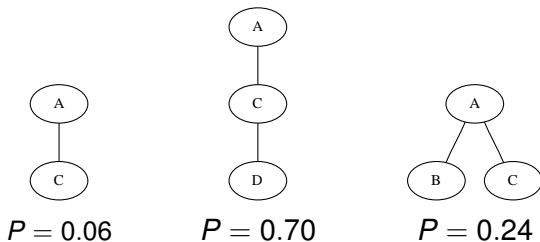
A node is assigned the condition  $C$ : means that this node is in the tree if  $C$  is true (and if its parent is in the tree).





# PW Semantics of Fuzzy Trees

A node is assigned the condition  $C$ : means that this node is in the tree if  $C$  is true (and if its parent is in the tree).



## Theorem

*The fuzzy tree model is as expressive as the PW model.*

# Queries on Fuzzy Trees

## Definition

Queries on fuzzy trees:

- Query **on underlying tree**.
- Probabilities: probability of the conjunction of the conditions of nodes of the mapping.

# Queries on Fuzzy Trees

## Definition

Queries on fuzzy trees:

- Query **on underlying tree**.
- Probabilities: probability of the conjunction of the conditions of nodes of the mapping.

## Theorem

$$Q(\llbracket T \rrbracket) = Q(T)$$

# Updates on Fuzzy Trees

- **Insertions:** no problem. Conditions required for the query to match added to inserted nodes.

# Updates on Fuzzy Trees

- **Insertions:** no problem. Conditions required for the query to match added to inserted nodes.
- **Deletions:** ok, but more problematic. May yield an exponential growth of the fuzzy tree in case of complex dependencies.

# Updates on Fuzzy Trees

- **Insertions:** no problem. Conditions required for the query to match added to inserted nodes.
- **Deletions:** ok, but more problematic. May yield an exponential growth of the fuzzy tree in case of complex dependencies.

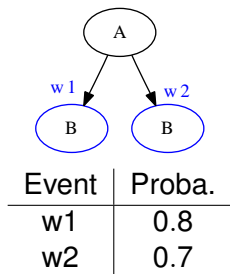
## Theorem

*If  $(\tau, c)$  is a probabilistic update transition and  $T$  a fuzzy tree:*

$$\llbracket (\tau, c)(T) \rrbracket = (\tau, c)(\llbracket T \rrbracket)$$

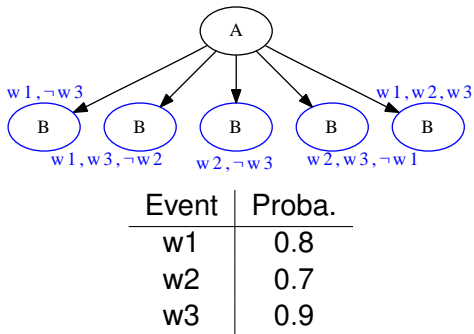
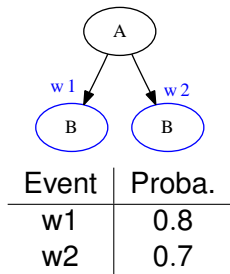
# Example: Deduplication

**Deduplication** of the two  $B$  nodes with confidence 0.9.



# Example: Deduplication

**Deduplication** of the two  $B$  nodes with confidence 0.9.





# Implementation

- Java-based

# Implementation

- Java-based
- File system storage (will look at an XML DB next)

# Implementation

- Java-based
- File system storage (will look at an XML DB next)
- Query evaluation: Qizx/open XQuery engine

# Implementation

- Java-based
- File system storage (will look at an XML DB next)
- Query evaluation: Qizx/open XQuery engine
- Updates expressed in XUpdate

# Implementation

- Java-based
- File system storage (will look at an XML DB next)
- Query evaluation: Qizx/open XQuery engine
- Updates expressed in XUpdate
- cf Demo

# Outline

- 1 Introduction
- 2 Framework
- 3 Possible Worlds Model
- 4 Simple Probabilistic Model
- 5 Fuzzy Tree Model
- 6 Conclusion**
  - Summary
  - Perspectives

# Summary

- A model for representing **probabilistic** information in a **semi-structured** database.

# Summary

- A model for representing **probabilistic** information in a **semi-structured** database.
- Special focus on **updating**, for managing a probabilistic warehouse.



# Summary

- A model for representing **probabilistic** information in a **semi-structured** database.
- Special focus on **updating**, for managing a probabilistic warehouse.
- **More expressive** and as concise as a simple model, **more concise** and as expressive as the naïve model.

# Perspectives



- Implementation finalization.

# Perspectives



- Implementation finalization.
- Full complexity analysis.

# Perspectives



- Implementation finalization.
- Full complexity analysis.
- More expressive queries (negative joints?), while keeping efficiency.

# Perspectives



- Implementation finalization.
- Full complexity analysis.
- More expressive queries (negative joints?), while keeping efficiency.
  
- Optimizations, simplifications of a fuzzy tree.

# Perspectives



- Implementation finalization.
  - Full complexity analysis.
  - More expressive queries (negative joints?), while keeping efficiency.
- 
- Optimizations, simplifications of a fuzzy tree.
  - Validation of fuzzy trees.