ProApproX is a query processor over probabilistic trees that represents a first step towards building a fully-fledged probabilistic semi-structured data management system.

It relies on:
- A generalization of the different uncertain data models in XML.
- Allows for efficient data querying with a subset of the XPath query language.
- Through techniques of exact calculations or efficient approximations of the result.

**Probabilistic Data**

This tree is a result of a Probabilistic Data Integration Process.

**Querying P-Documents**

Tree Pattern Queries (TPQ)

Tree Pattern Queries with joins (TPQJ)

**Probability of a Query**

Q1: /person[name="Chris"]//phone/text()

- How to compute the probability of the YES answer to this query?
- Lineage to the query (probabilistic path of each answer): a DNF

\[ F = (e_2 \land e_3 \land e_4) \lor (e_2 \land e_3 \land e_5) \]

**Approximations**

When a Linear Computation is not possible, we run an appropriate approximation:

- Naïve Monte Carlo sampling for additive app.:
  - Linear but could take exponentially many samples to converge to a good approximation for low probabilities.

- Biased Monte Carlo sampling for multiplicative app.:
  - Running time grows in \( O(n^3 \ln n) \) in the number of clauses.

- Self-Adjusting Coverage Algorithm for the DNF probability problem:
  - For a fixed error \( \varepsilon \) and a fixed reliability factor \( \delta \), the algorithm is linear in the length of \( F \) times \( \ln(1/\varepsilon/\delta^2) \).

**Complexity**

Find and sum the probabilities of the satisfying assignments for the DNF (lineage formula) : \#P-Hard problem

- No polynomial time algorithm for the exact solution if \( P \neq NP \)
- \#P problems ask "how many" rather than "are there any".

How many graph coloring using \( k \) colors are there for a particular graph \( G \)?