Aggregate Queries for Discrete and Continuous Probabilistic XML

S. Abiteboul,[1] T-H. H. Chan,[2] E. Kharlamov,[1,3] W. Nutt,[3] P. Senellart[4]

[1]INRIA Saclay – Île-de-France
[2]The University of Hong Kong
[3]Free University of Bozen-Bolzano
[4]Télécom ParisTech

ICDT, March 2010

Outline

1. Probabilistic data
2. Problem definition
3. Aggregating discrete Probabilistic XML
4. Aggregating continuous Probabilistic XML

Applications of Probabilistic Data

- Approximate query processing: ranking, linkage
- Information extraction: approximate search for entities (e.g. names) in text
- Sensor data: imprecise or missing readings
- ...

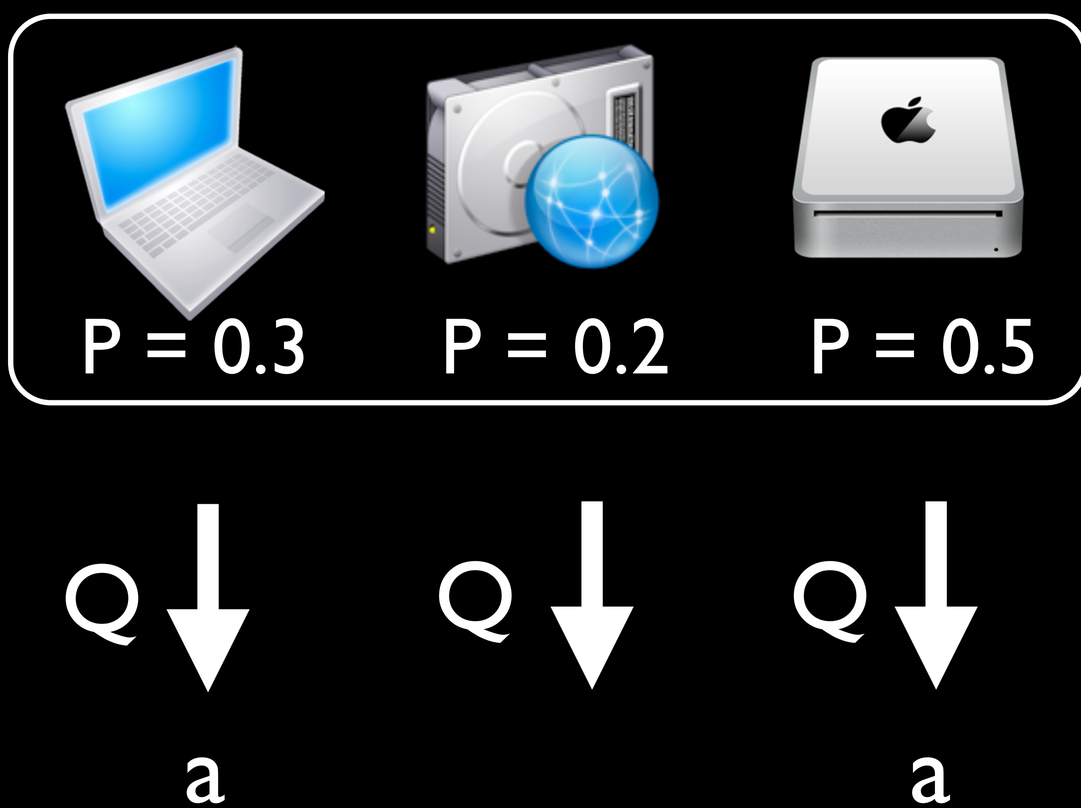
Probabilistic Database

Probabilistic DB:

P = 0.3 P = 0.2 P = 0.5

Probabilistic Database

Probabilistic DB:



Probabilistic Database

Probabilistic DB:

P = 0.3

P = 0.2

P = 0.5

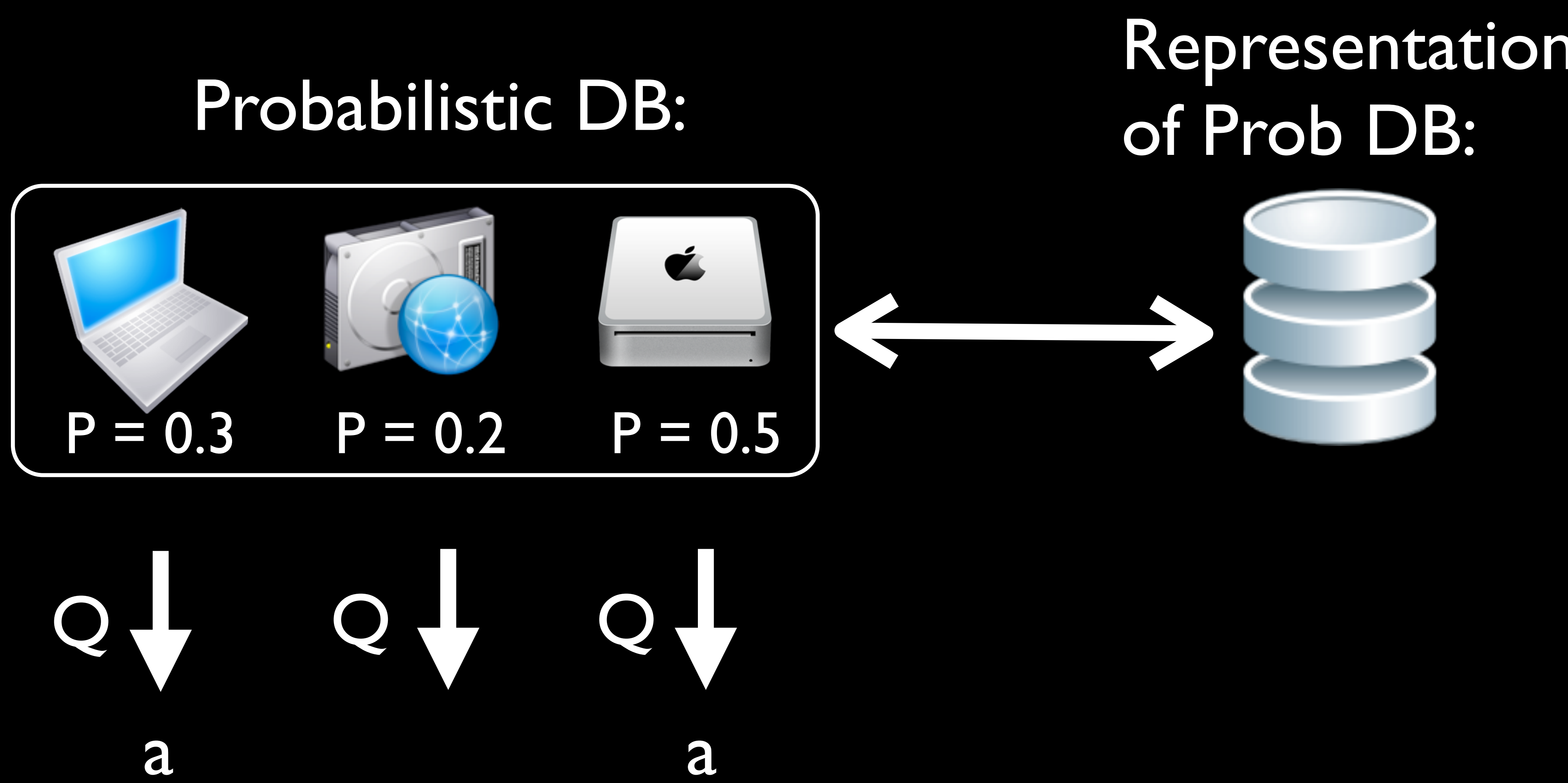
Q ↓ a

Q ↓

Q ↓ a

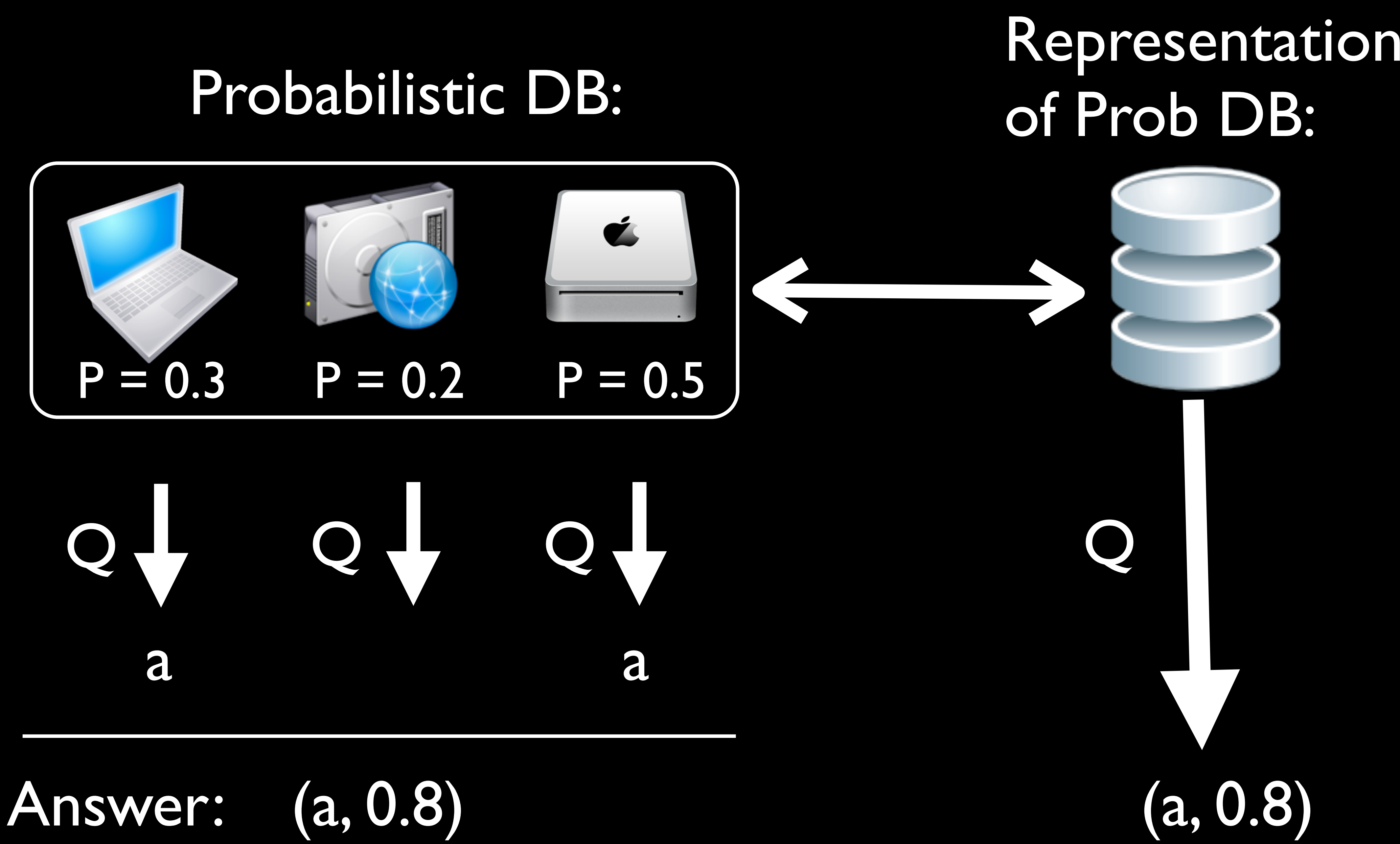
Answer: (a, 0.8)

Probabilistic Database

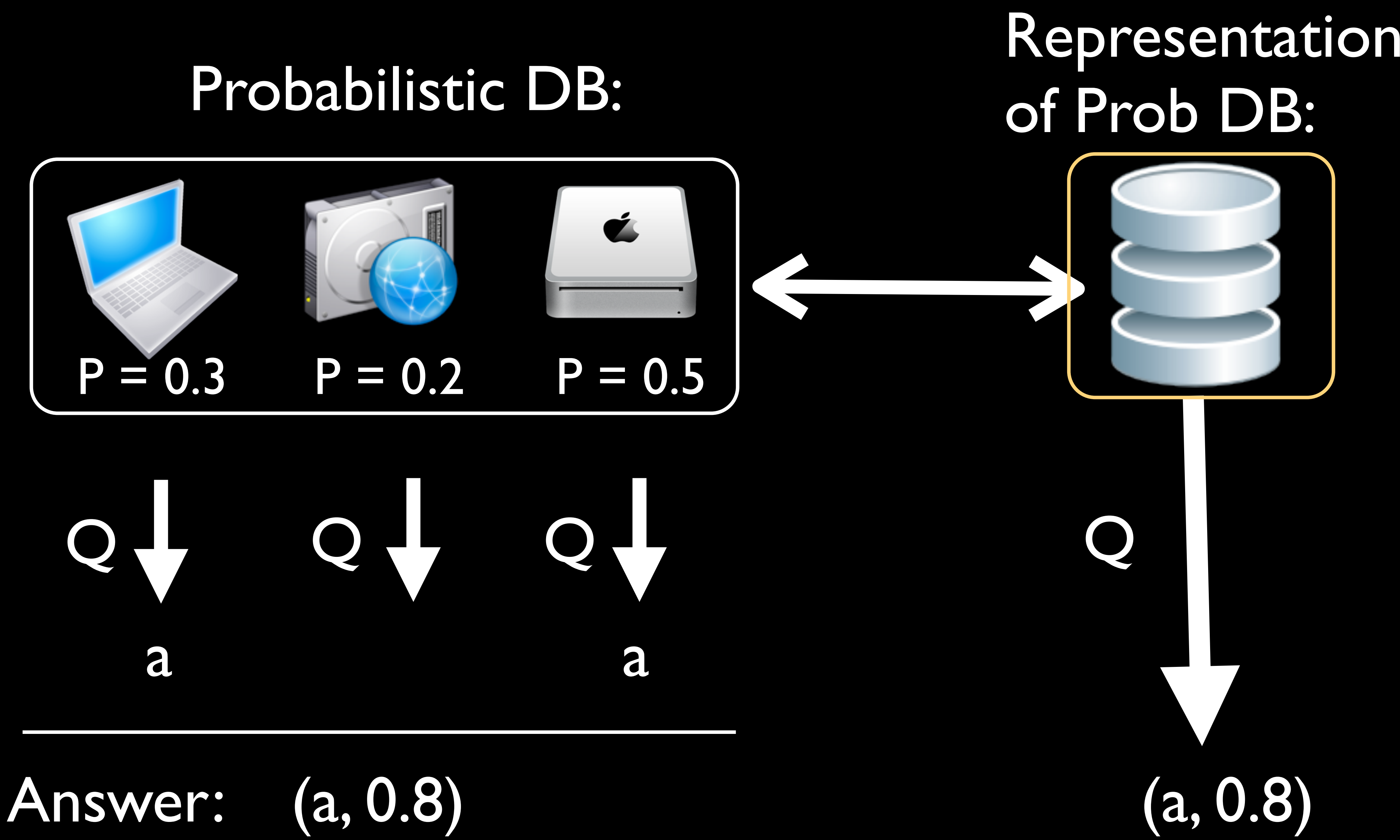


Answer: (a, 0.8)

Probabilistic Database

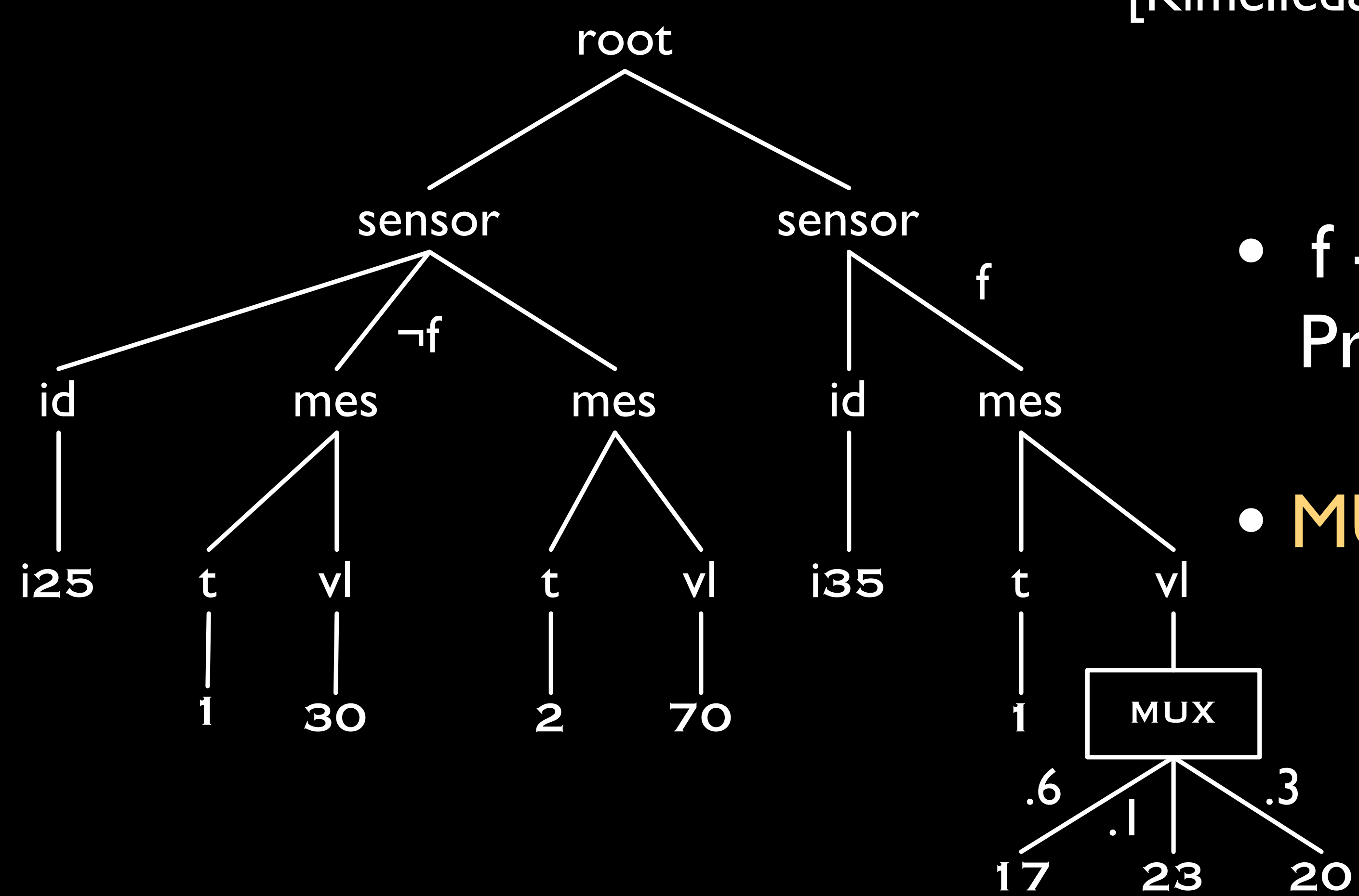


Probabilistic Database



PXML with Events and Distributional Nodes

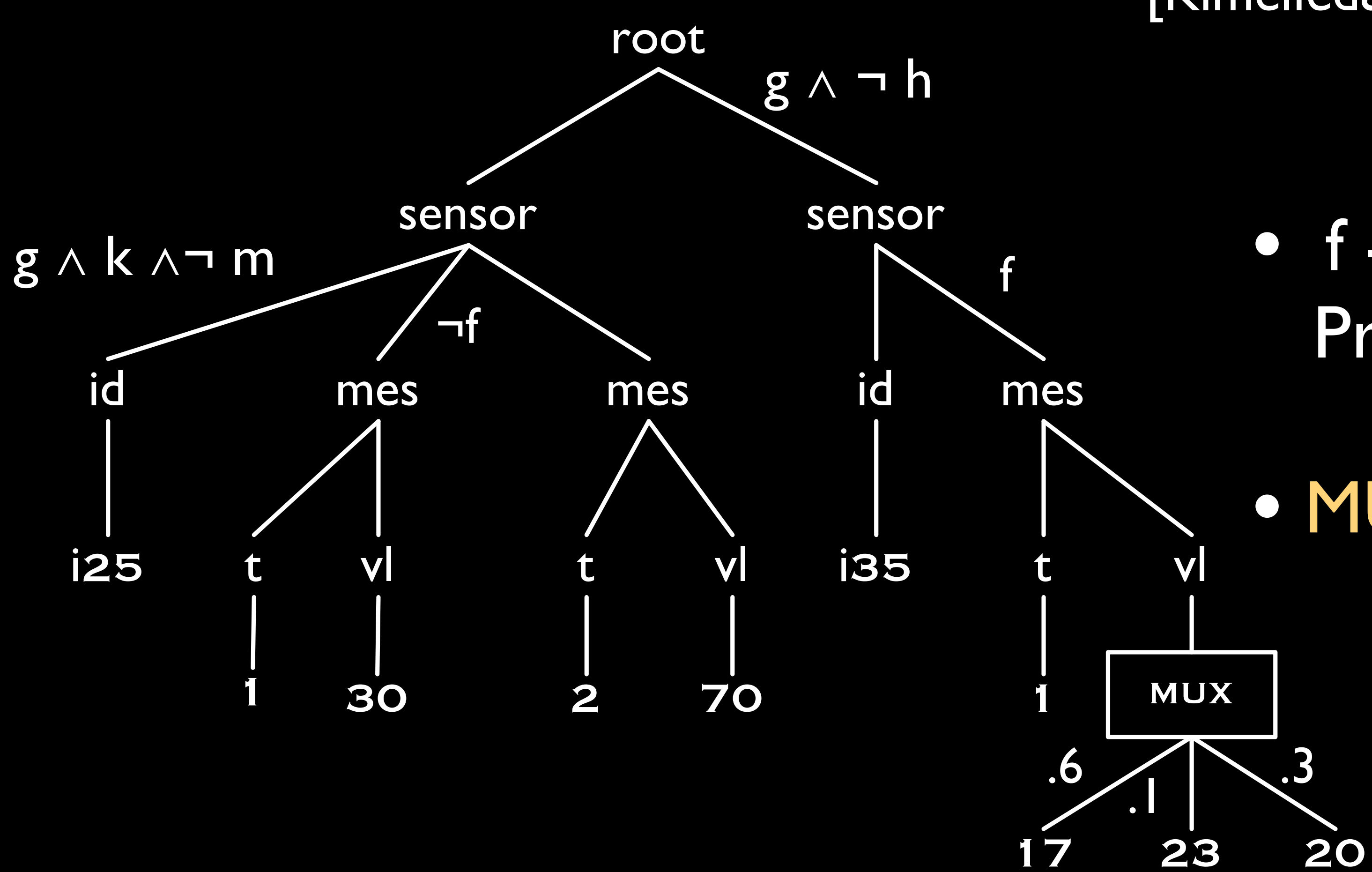
[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: “weather is fine”
  Pr(f) = .4
- MUX - mutually exclusive options

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]

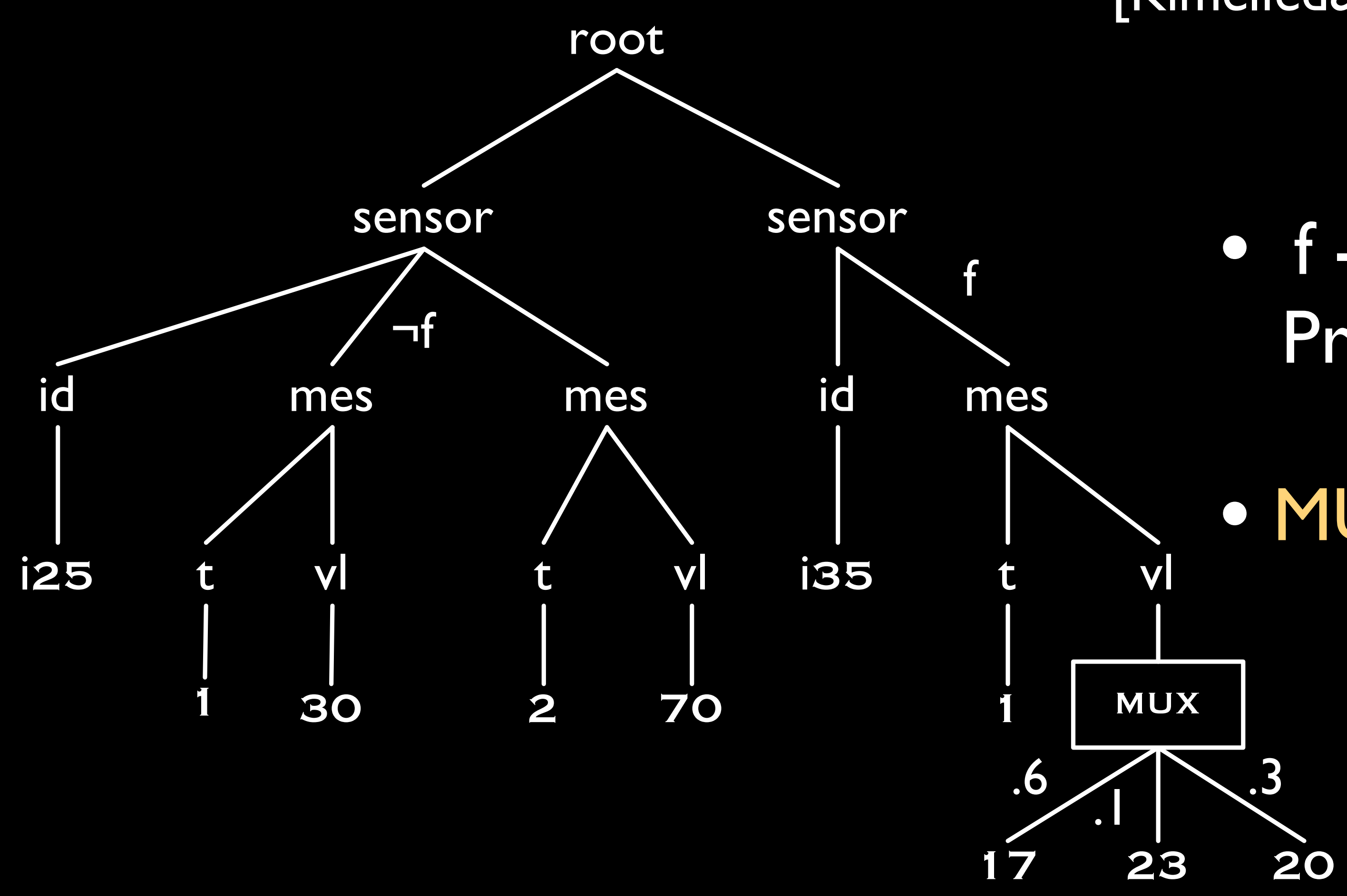


- f - event: “weather is fine”
  Pr(f) = .4

- MUX - mutually exclusive options

PXML with Events and Distributional Nodes

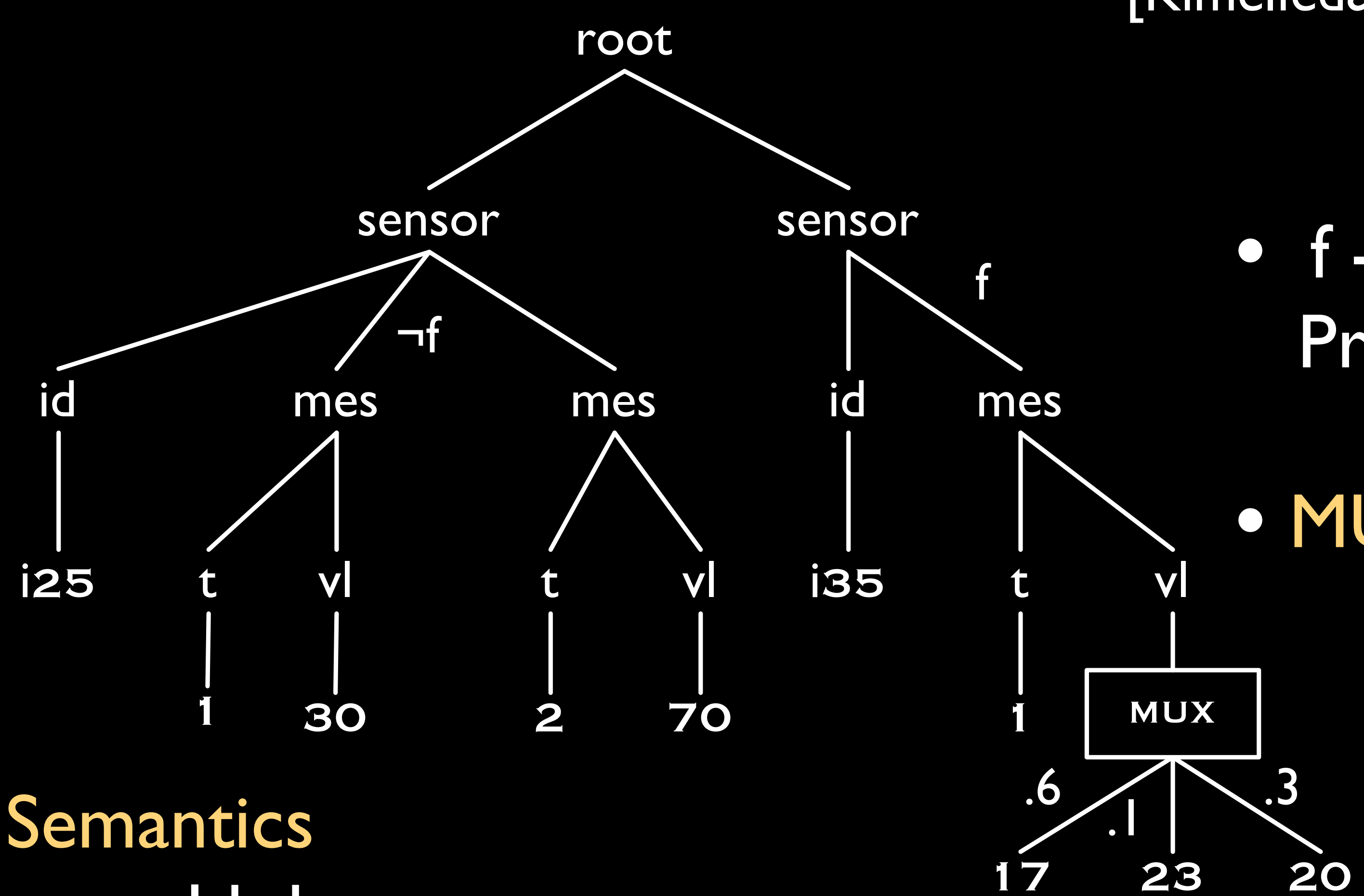
[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: “weather is fine”
  Pr(f) = .4
- MUX - mutually exclusive options

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: "weather is fine"
  Pr(f) = .4

- MUX - mutually exclusive options

Semantics

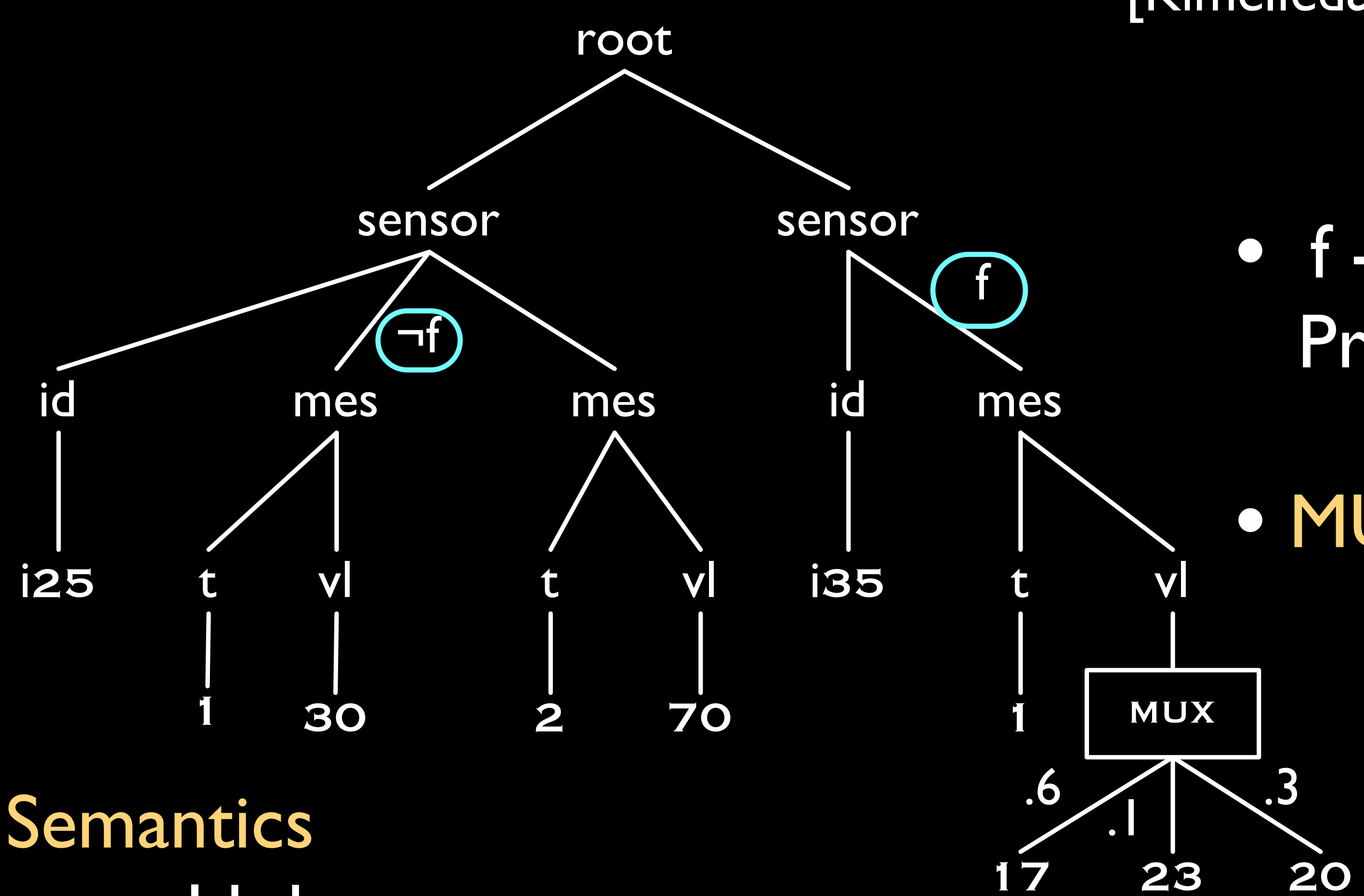
a world d:

- f = true, Pr(f) = 0.4
- MUX: 23, Pr(23) = 0.1

Pr(d) = 0.4 x 0.1

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: “weather is fine”
  $Pr(f) = .4$
- MUX - mutually exclusive options

Semantics

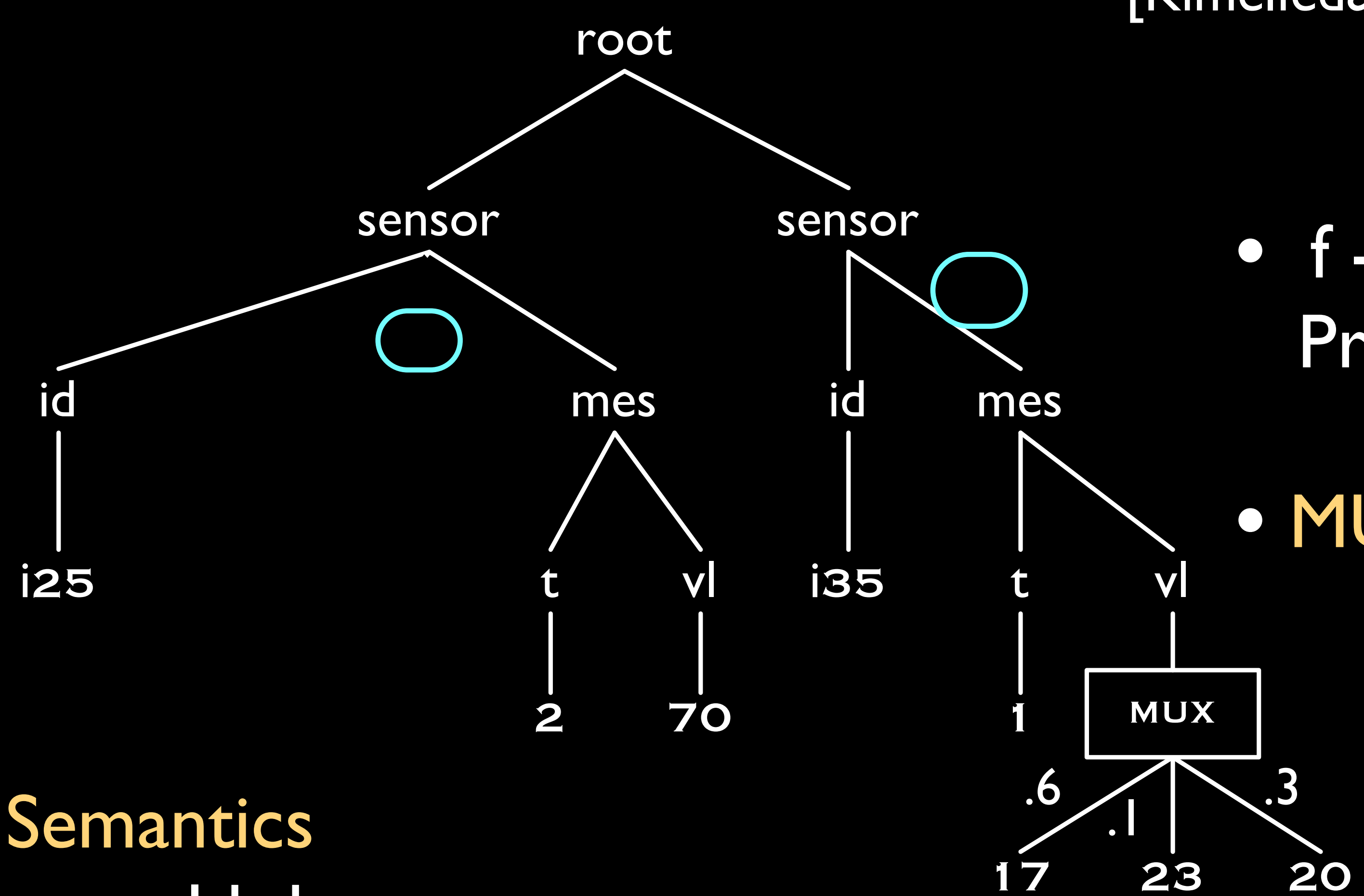
a world d:

- f = true, $Pr(f) = 0.4$
- MUX: 23, $Pr(23) = 0.1$

$Pr(d) = 0.4 \times 0.1$

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: “weather is fine”
  Pr(f) = .4
- MUX - mutually exclusive options

Semantics

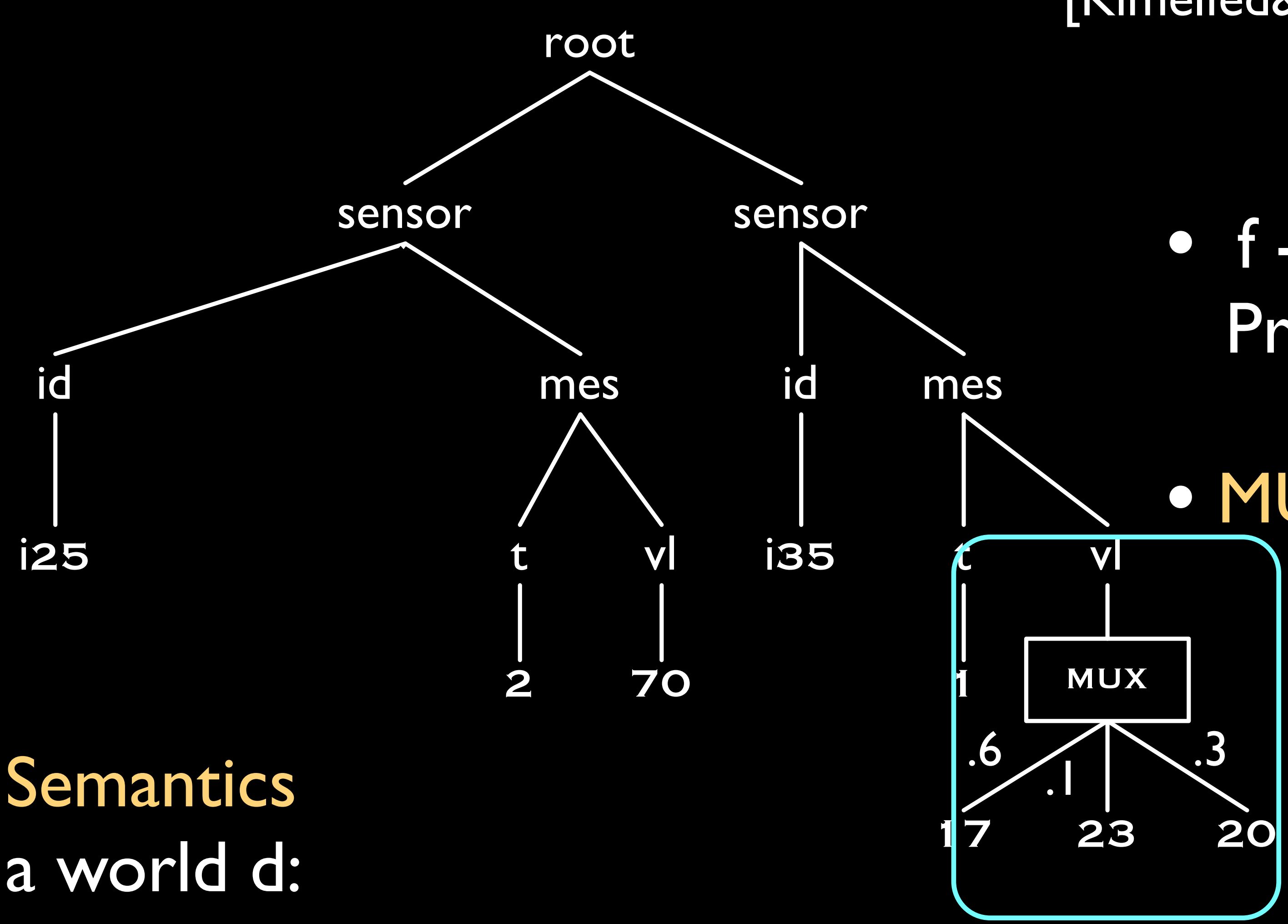
a world d:

- f = true, Pr(f) = 0.4
- MUX: 23, Pr(23) = 0.1

Pr(d) = 0.4 x 0.1

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: “weather is fine”
  Pr(f) = .4

- MUX - mutually exclusive options

Semantics

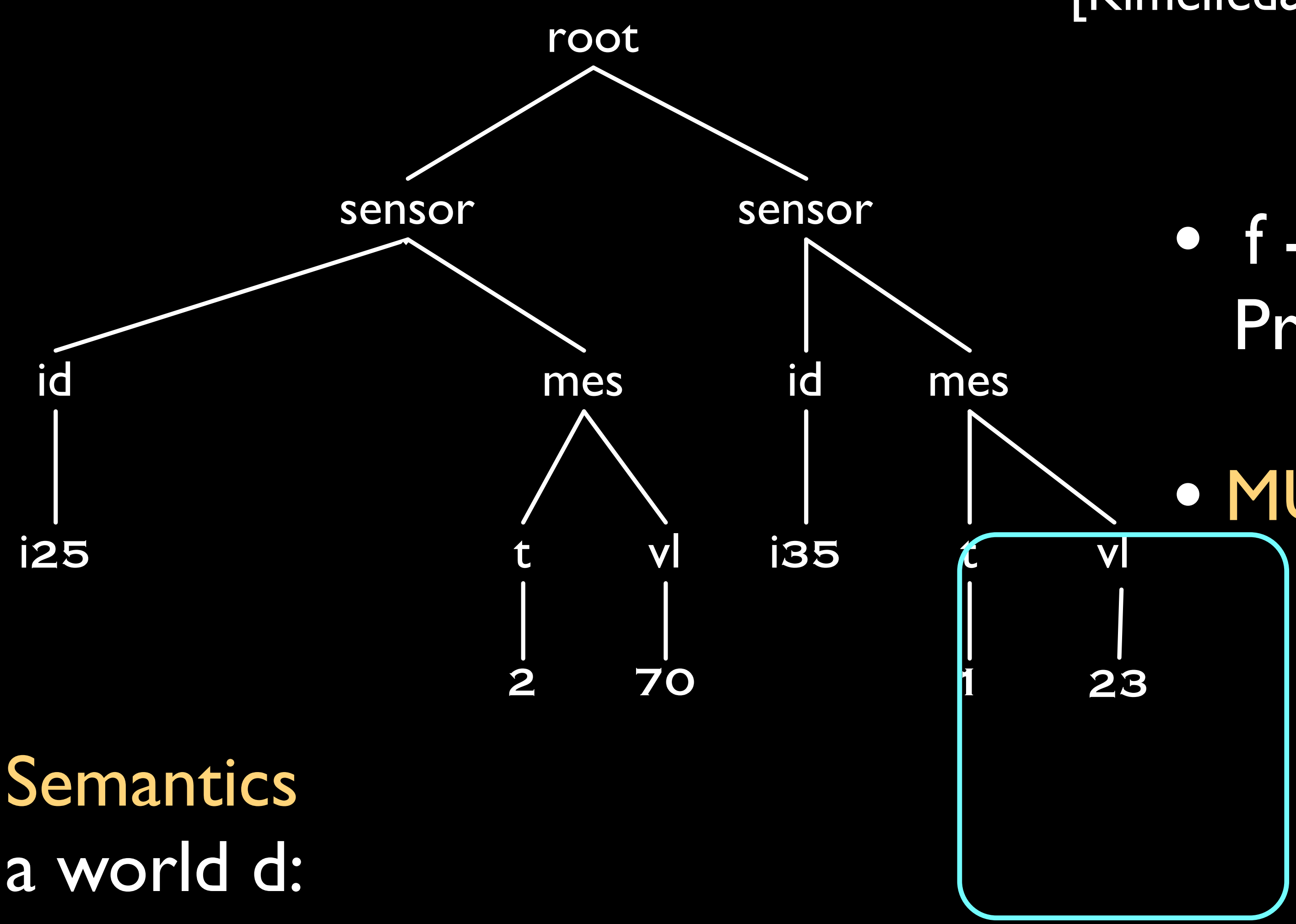
a world d:

- f = true, Pr(f) = 0.4
- MUX: 23, Pr(23) = 0.1

Pr(d) = 0.4 x 0.1

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: "weather is fine"
  Pr(f) = .4

- MUX - mutually exclusive options

Semantics

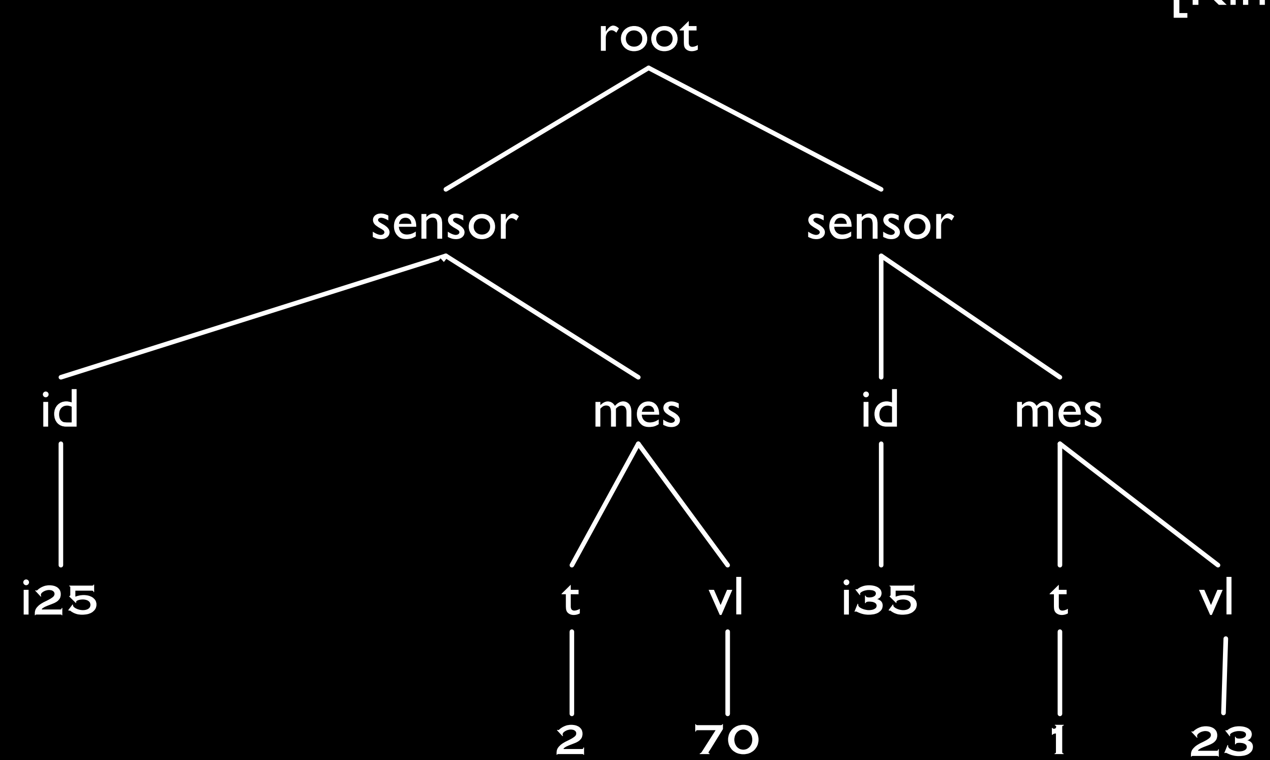
a world d:

- f = true, Pr(f) = 0.4
- MUX: 23, Pr(23) = 0.1

Pr(d) = 0.4 x 0.1

PXML with Events and Distributional Nodes

[Kimelfed&al:2007] [Senellart&al:2007]



- f - event: “weather is fine”
  Pr(f) = .4
- MUX - mutually exclusive options

Semantics

a world d:

- f = true, Pr(f) = 0.4
- MUX: 23, Pr(23) = 0.1

Pr(d) = 0.4 x 0.1

Discrete Probabilistic XML Documents

- Probabilistic XML document D
  - represents (exponentially) many documents d
  - each with a probability Pr(d)
- It is achieved by
  - Conjunctions of event literals on edges. Capture long-distance dependencies
  - Distributional nodes: Mux, Ind, Det, Exp. Capture local (hierarchical) dependencies

Discrete Probabilistic XML Documents

- Probabilistic XML document D
  - represents (exponentially) many documents d
  - each with a probability Pr(d)
- It is achieved by
  - Conjunctions of event literals on edges.
    Capture long-d

    Special case of event formulas
  - Distributional nodes: Mux, Ind, Det, Exp.
    Capture local (hierarchical) dependencies

What is Known?

[Kimelfed&al:2007]

[Senellart&al:2007]

- Answering simple XPath queries
  - Distributional nodes: PTIME
  - Events: $FP^{\#P}$-complete

[Cohen&al:2008]

[Re&al:2007]

- Simple XPath over Mux-Det PXML with HAVING constraints:
  - PTIME for COUNT and MIN
  - NP-hard for SUM and AVG

What is Known?

[Kimelfed&al:2007]
[Senellart&al:2007]

- Answering simple XPath queries
  - Distributional nodes: PTIME
  - Events: $FP^{\#P}$-complete

[Cohen&al:2008]
[Re&al:2007]

- Simple XPath over Mux-Det PXML with HAVING constraints:
  - PTIME for COUNT and MIN
  - NP-hard for SUM and AVG

NO events

Outline

1. Probabilistic data
2. Problem definition
3. Aggregating discrete Probabilistic XML
4. Aggregating continuous Probabilistic XML

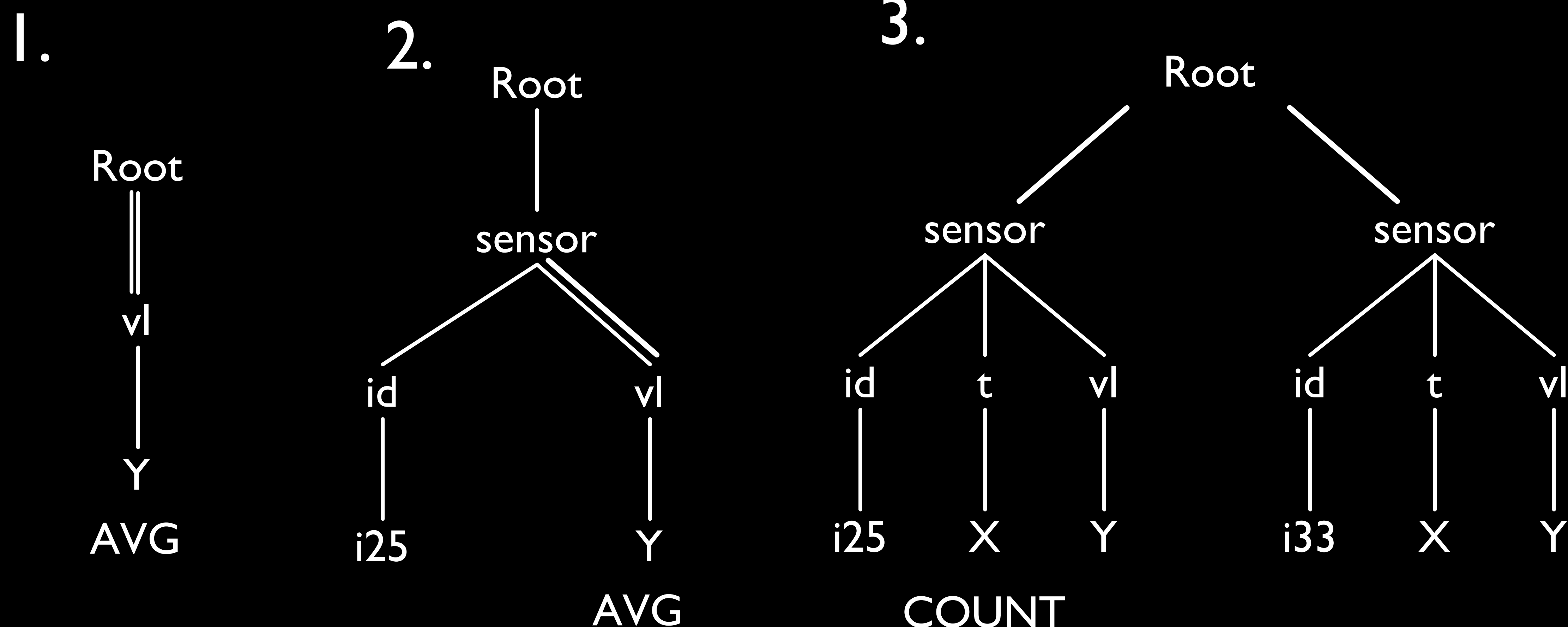
Aggregate Queries

1. What is the average temperature across sensors?
2. What is the average temperature for sensor i25?
3. How often did sensors i25 and i33
   give the same measurement simultaneously?

$\Rightarrow$ we want to answer queries with aggregate functions:
MIN/MAX, TopK, COUNT, SUM, COUNTD, AVG

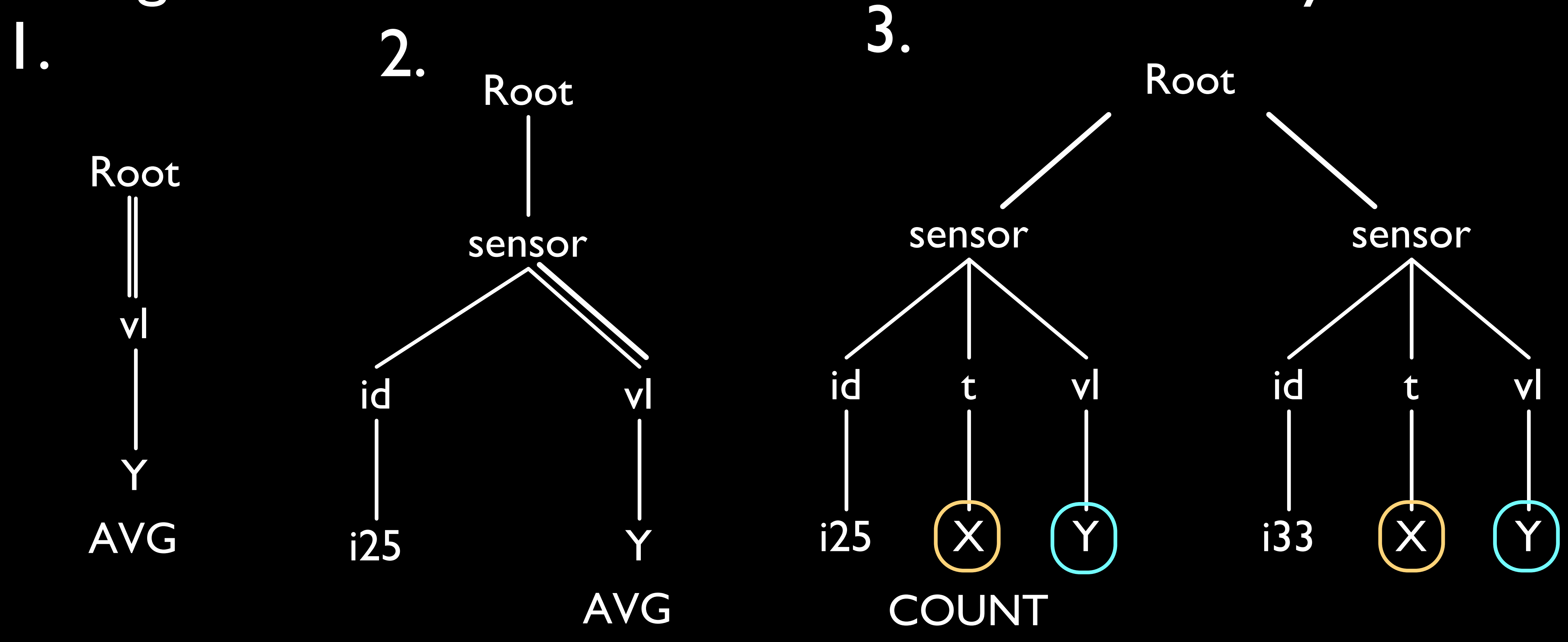
Query Models

1. What is the average temperature across sensors?
2. What is the average temperature for sensor i25?
3. How often did sensors i25 and i33 give the same measurement simultaneously?



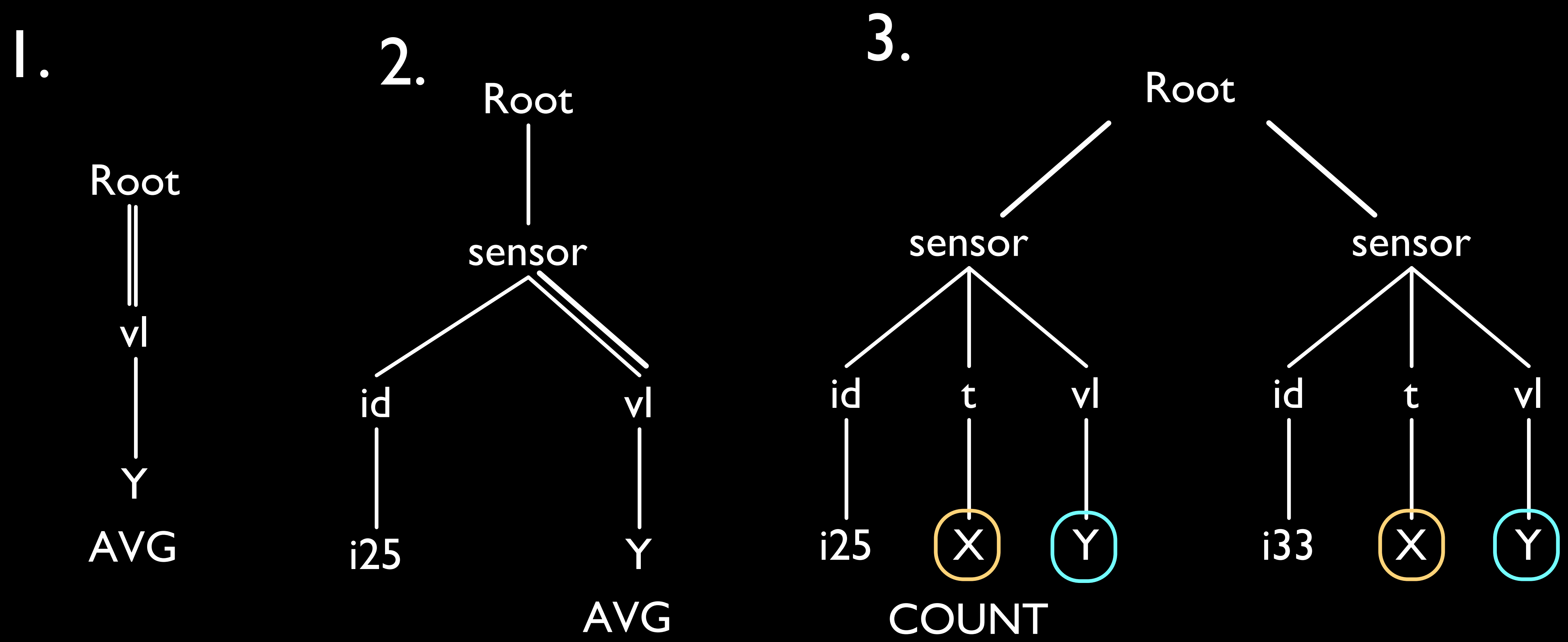
Query Models

1. What is the average temperature across sensors?
2. What is the average temperature for sensor i25?
3. How often did sensors i25 and i33 give the same measurement simultaneously?

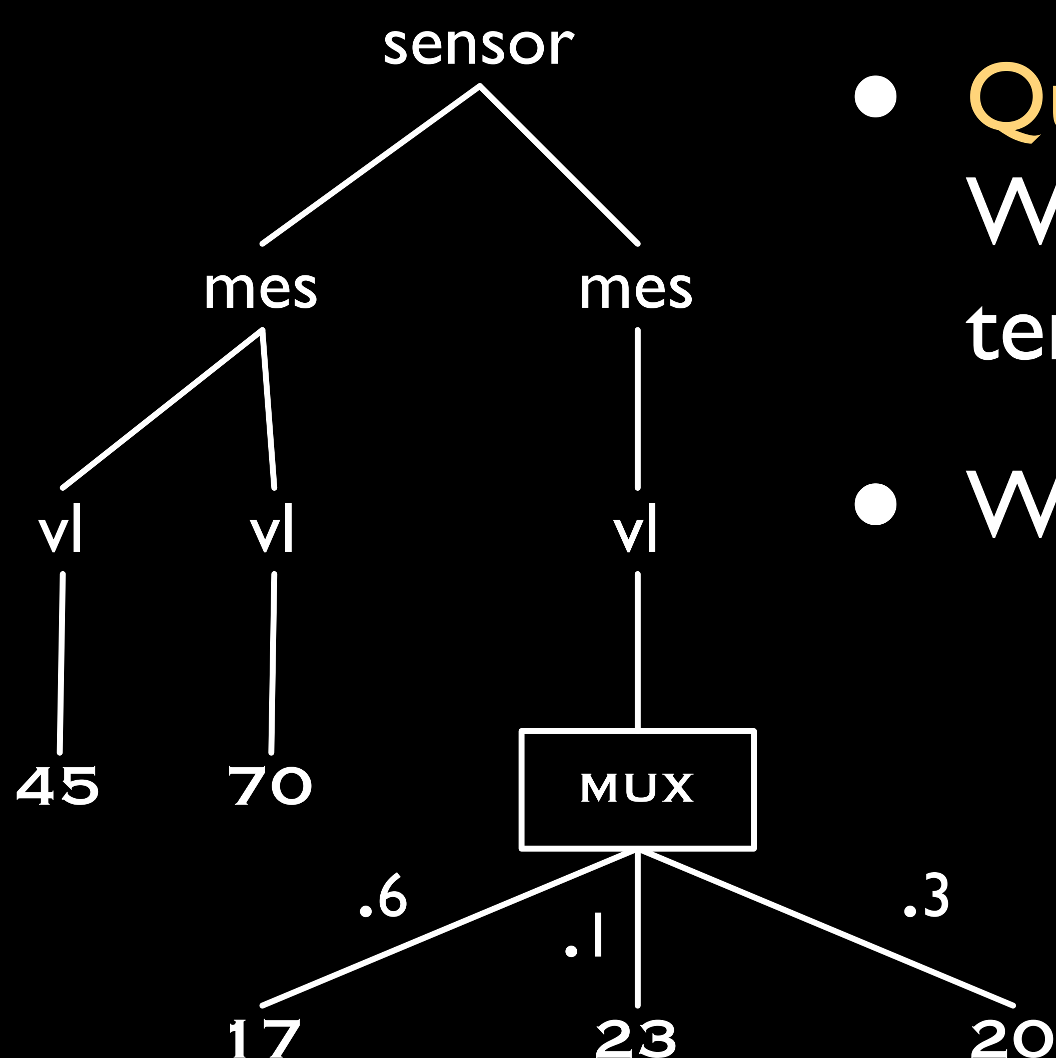


Query Models

1. Single-Path queries - SP
2. Tree-Pattern queries - TP
3. Tree-Pattern queries with Joins - TPJ



Semantics of AQs



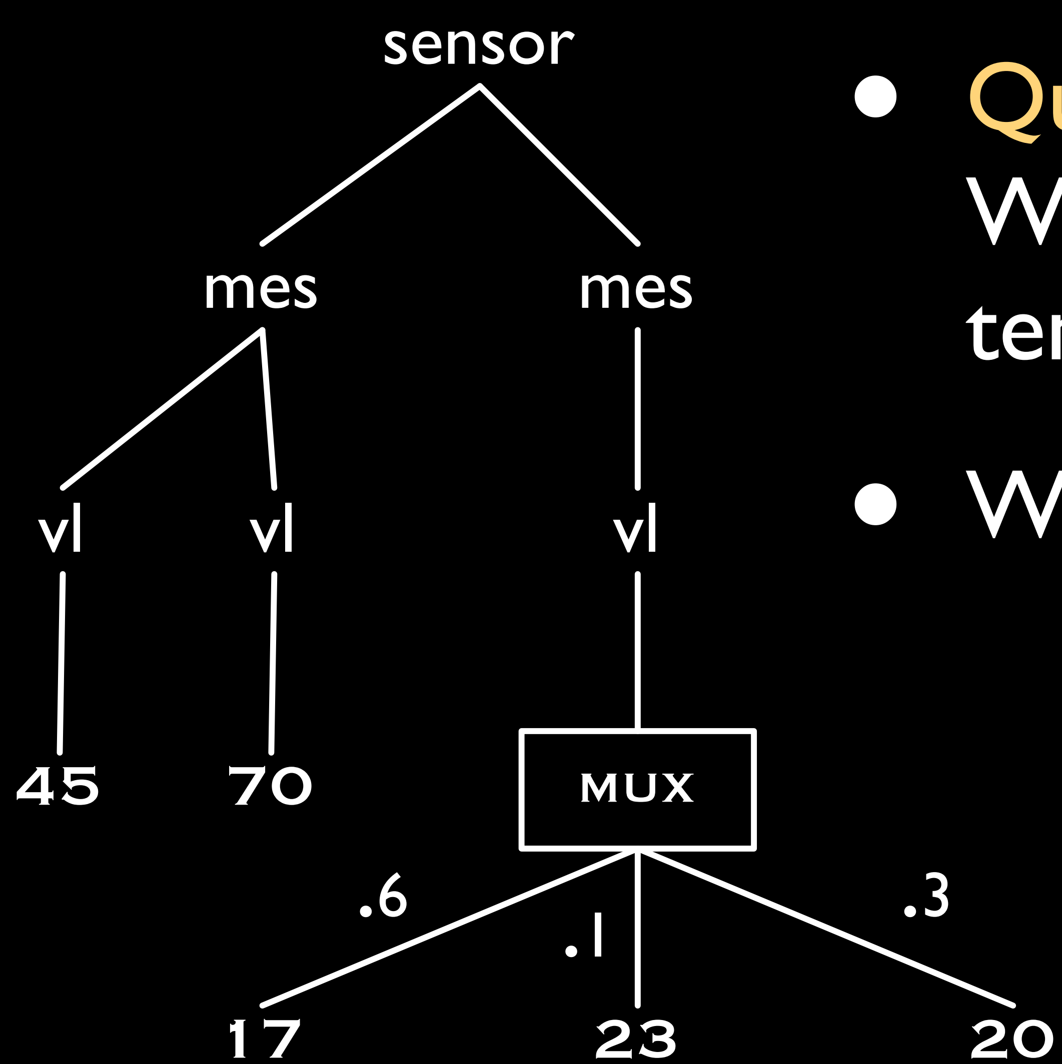
- Query:
  What is the average temperature?
- What should be an answer?

  AVG(d17) = 44, Pr(d17)=.6

  AVG(d23) = 46, Pr(d23)=.1

  AVG(d20) = 45, Pr(d20)=.3

Semantics of AQs



- Query:
  What is the average temperature?
- What should be an answer?

  AVG(d17) = 44, Pr(d17)=.6

  AVG(d23) = 46, Pr(d23)=.1

  AVG(d20) = 45, Pr(d20)=.3

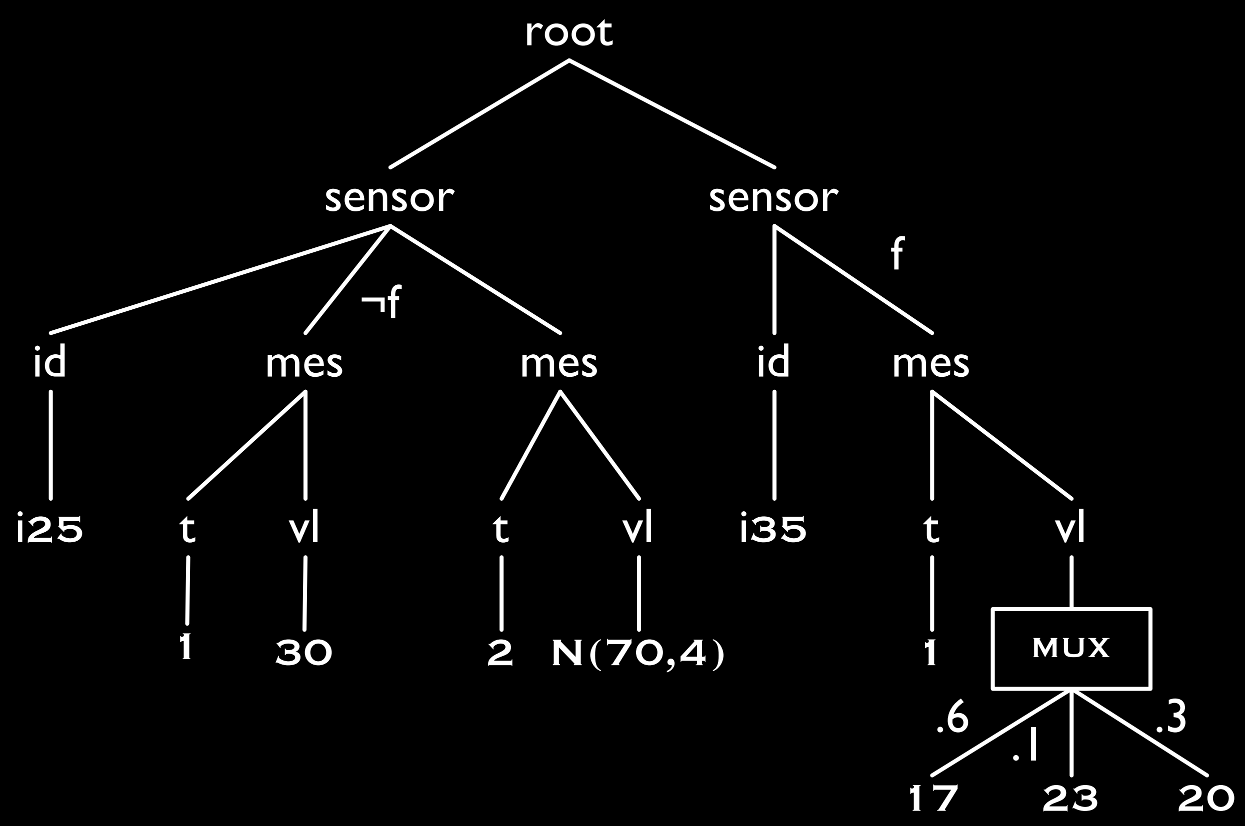
Distribution of aggregate values over all documents represented by the PXML document

Problems to Investigate for Discrete PXML

For PXML document D, constant C

- Possible answers:
  decide $Pr(Q(D)=C) > 0$
- Probability computation:
  compute $Pr(Q(D)=C)$
- Moment computation:
  compute $E(Q(D)^k)$ E is "expected value"

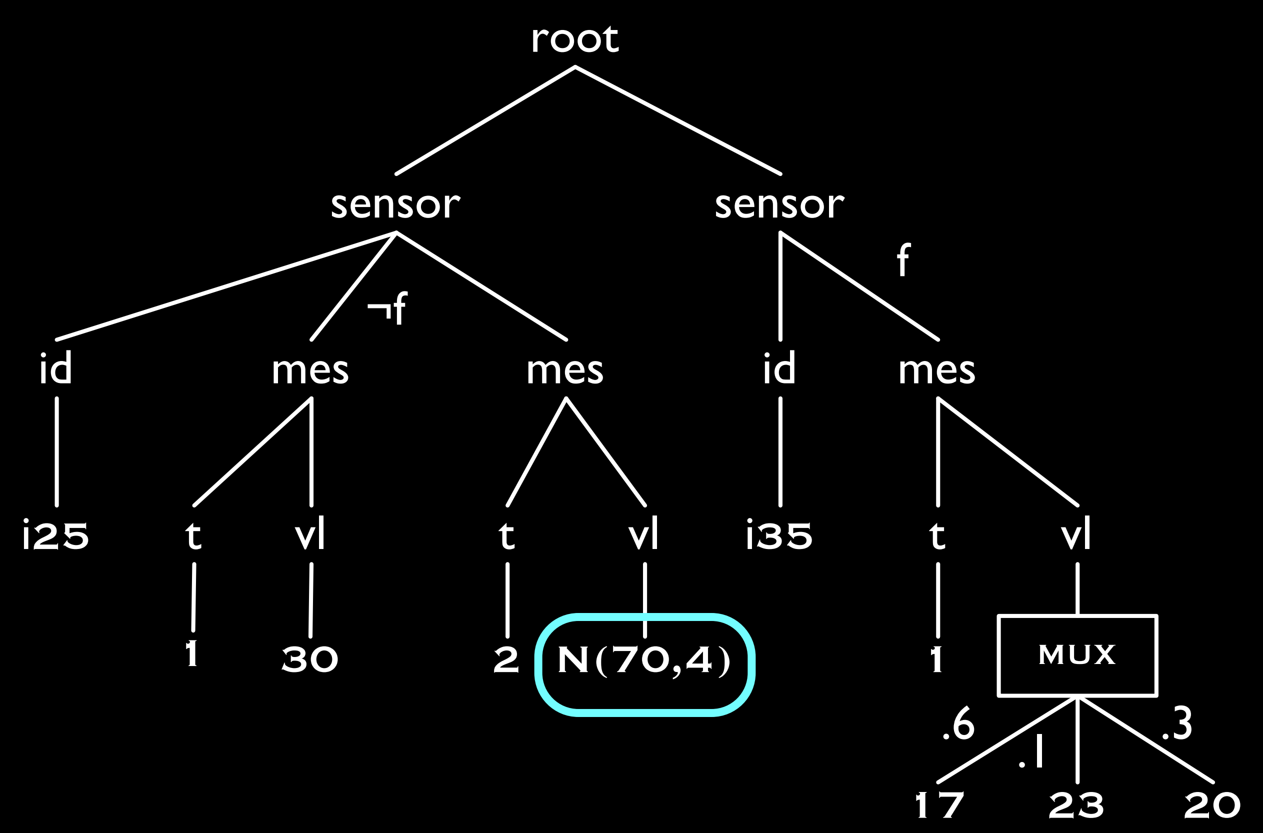
Continuous PXML



- Incorporate continuous distributions in PXML leaves
- Aggregate continuous PXML

At the moment there is no formal semantics for continuous probabilistic XML models

Continuous PXML



- Incorporate continuous distributions in PXML leaves
- Aggregate continuous PXML

At the moment there is no formal semantics for continuous probabilistic XML models

Outline

1. Probabilistic data
2. Problem definition
3. Aggregating discrete Probabilistic XML
4. Aggregating continuous Probabilistic XML

Data Complexity of Query Answering

	Query Language		
PXML Model	Single Path	Tree Pattern	Tree Pat. Joins
Event Conjunctions	$FP^{\#P}$-complete		
Distributional Nodes	P		$FP^{\#P}$-complete

What is difficult?

- joins in queries
- events in data

Data Complexity of Query Answering

	Query Language		
PXML Model	Single Path	Tree Pattern	Tree Pat. Joins
Event Conjunctions	$FP^{\#P}$-complete		
Distributional Nodes	P		$FP^{\#P}$-complete

What is difficult?

- joins in queries
- events in data

Is it getting more difficult with aggregation?

Aggregating PXML-Events

	Aggregate Query Language		
Problems	Single Path	Tree Pattern	Tree Pat. Joins
Possible Answers	NP-complete		
Probability Computation	$FP^{\#P}$-complete		
Moment Computation	COUNT, SUM: PTIME MIN, AVG COUNTD: $FP^{\#P}$-comp	$FP^{\#P}$-complete	

Data-complexity

Aggregates: COUNT, SUM, MIN, COUNTD, AVG

Aggregating PXML-Events

	Aggregate Query Language		
Problems	Single Path	Tree Pattern	Tree Pat. Joins
Possible Answers	NP-complete		
Probability Computation	$FP^{\#P}$-complete		
Moment Computation	COUNT, SUM: PTIME MIN, AVG COUNTD: $FP^{\#P}$-comp	$FP^{\#P}$-complete	

Data-complexity

Aggregates: COUNT, SUM, MIN, COUNTD, AVG

Aggregating PXML with Distributional Nodes

<table>
<tr><th></th><th colspan="3">Aggregate Query Language</th></tr>
<tr><th>Problems</th><th>Single Path</th><th>Tree Pattern</th><th>Tree Pat. Joins</th></tr>
<tr><td rowspan="2">Possible Answers</td><td colspan="3">SUM, AVG, COUNTD: NP-complete</td></tr>
<tr><td colspan="2">COUNT, MIN: PTIME</td><td>COUNT, MIN : NP</td></tr>
<tr><td>Probability Computation</td><td colspan="2">SUM, AVG, COUNTD: $FP^{\#P}$-complete
COUNT, MIN: PTIME</td><td rowspan="3">$FP^{\#P}$-complete</td></tr>
<tr><td>Probability SUM in |input| +|output|</td><td rowspan="2">PTIME</td><td>$FP^{\#P}$</td></tr>
<tr><td>Moment Computation</td><td>AVG: $FP^{\#P}$
others: PTIME</td></tr>
</table>

Data-complexity

Aggregates: COUNT, SUM, MIN, COUNTD, AVG

Aggregating PXML with Distributional Nodes

	Aggregate Query Language		
Problems	Single Path	Tree Pattern	Tree Pat. Joins
Possible Answers	SUM, AVG, COUNTD: NP-complete		
	COUNT, MIN: PTIME		COUNT, MIN : NP
Probability Computation	SUM, AVG, COUNTD: $FP^{\#P}$-complete COUNT, MIN: PTIME		$FP^{\#P}$-complete
Probability SUM in |input| +|output|	PTIME	$FP^{\#P}$	
Moment Computation		AVG: $FP^{\#P}$ others: PTIME	

Data-complexity

Aggregates: COUNT, SUM, MIN, COUNTD, AVG

Tractable Cases

Key components of tractability:

- Hierarchical structure of PXML documents imposed by distributional nodes
- Some aggregate functions can exploit the hierarchy - monoid functions

Monoid: COUNT, SUM, MIN, TopK, PARITY, ...

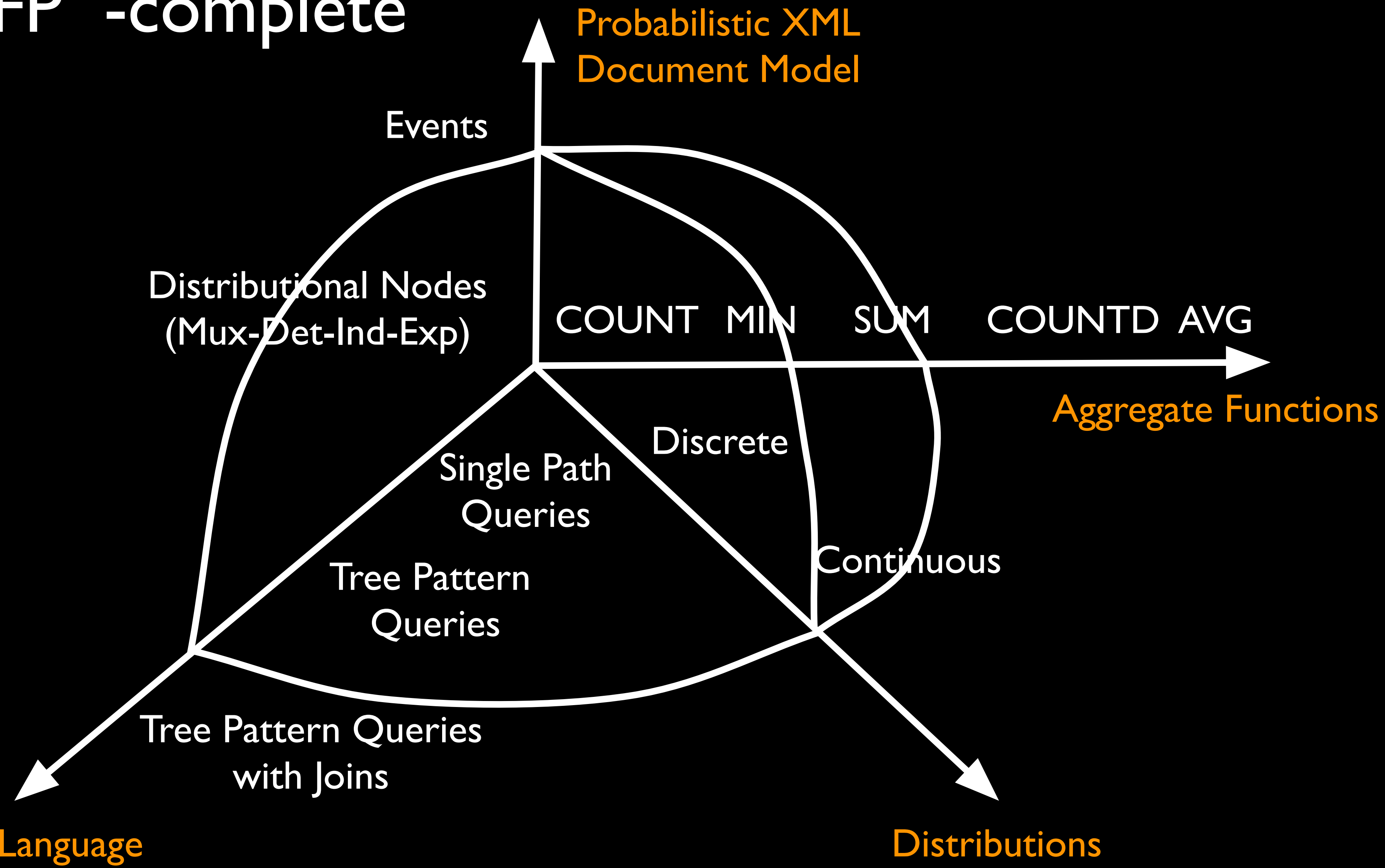
Non Monoid: COUNTD, AVG

PTIME algorithm to compute distributions:
Bottom-up evaluation using convex sums and convolutions

The Problem Space

Outside: intractable, i.e., NP-, $FP^{\#P}$-complete

Inside: PTIME



Approximating Query Answers

- Many problems are NP- or $FP^{\#P}$-complete
  How good are Monte-Carlo methods?
- By Hoeffding bound, to achieve

  $| E(\alpha(D)^k) - \text{Estimate} | < \varepsilon$ with $Pr = 1-\delta$

  at most $O(R^{2k}\, 1/\varepsilon^2 \log(1/\delta))$ samples is needed

$\Rightarrow$ for α=COUNTD

quadratically many samples are needed

Approximating Query Answers

- Many problems are NP- or $FP^{\#P}$-complete
  How good are Monte-Carlo methods?
- By Hoeffding bound, to achieve

  $| E(\alpha(D)^k) - \text{Estimate} | < \varepsilon$ with $Pr = 1-\delta$

  at most $\boxed{O(R^{2k} 1/\varepsilon^2 \log(1/\delta))}$ samples is needed

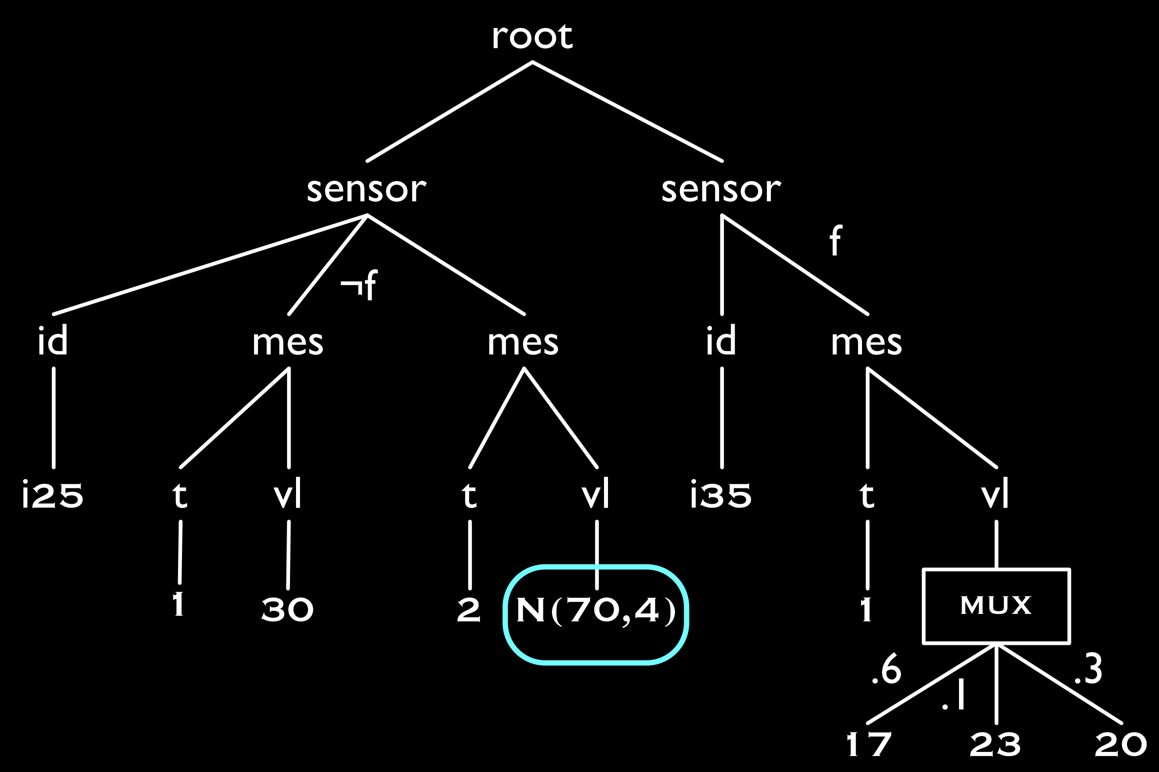
$\Rightarrow$ for α=COUNTD
quadratically many samples are needed

Outline

1. Probabilistic data
2. Problem definition
3. Aggregating discrete Probabilistic XML
4. Aggregating continuous Probabilistic XML

Discrete vs Continuous Models

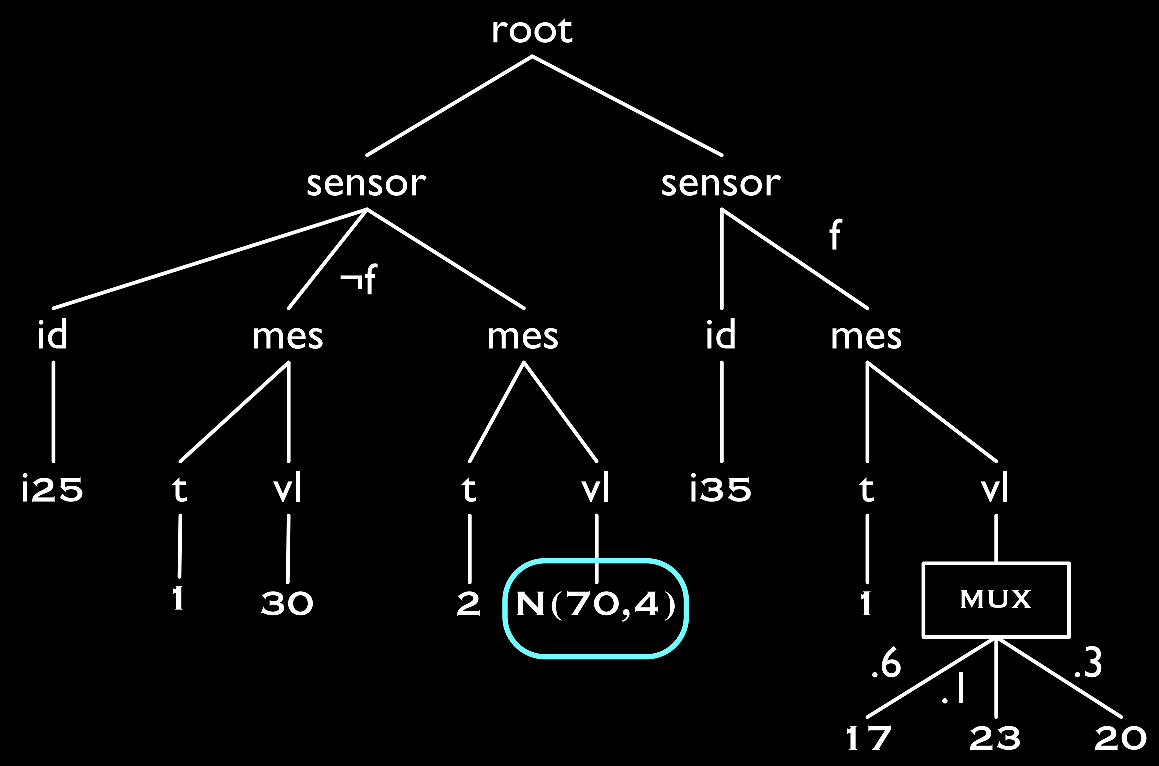
- Finite case:
  - finite sets of trees
  - where every tree has a non-zero probability



- Continuous case:
  - infinite sets of trees
  - where some (infinite) subsets of trees have non-zero probability measure

Discrete vs Continuous Models

- Finite case:
  - finite sets of trees
  - where every tree has a non-zero probability
- Continuous case:
  - infinite sets of trees
  - where some (infinite) subsets of trees have non-zero probability measure

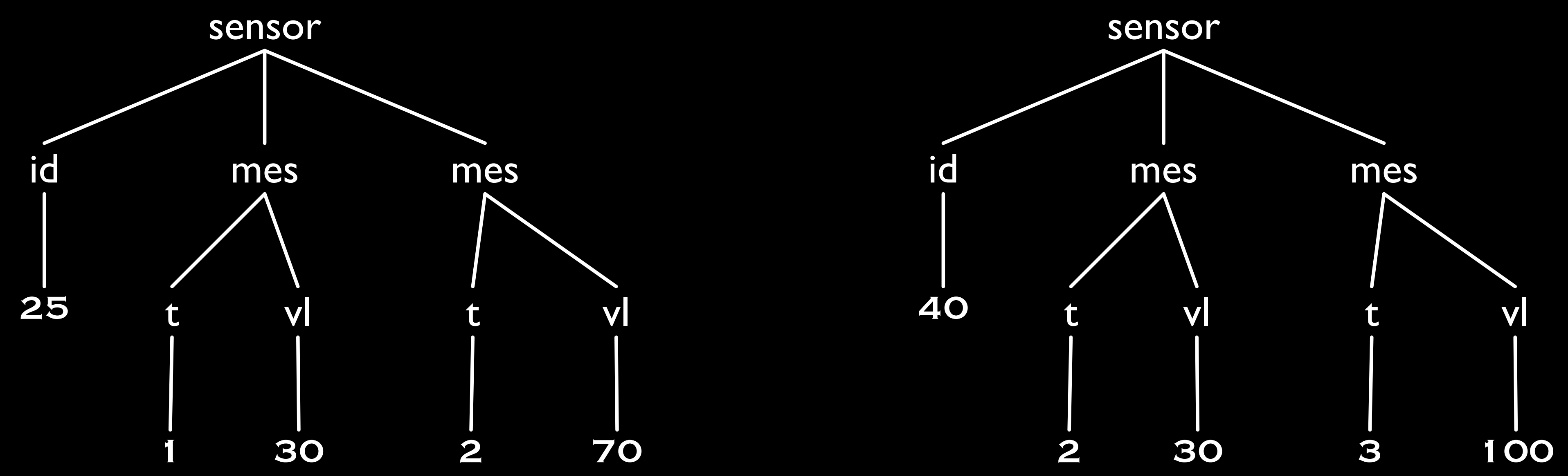


How to measure infinite sets of trees?

Measuring Infinite Sets of Trees

1. Take a set S of trees with
   - real values on the leaves / share the same structure

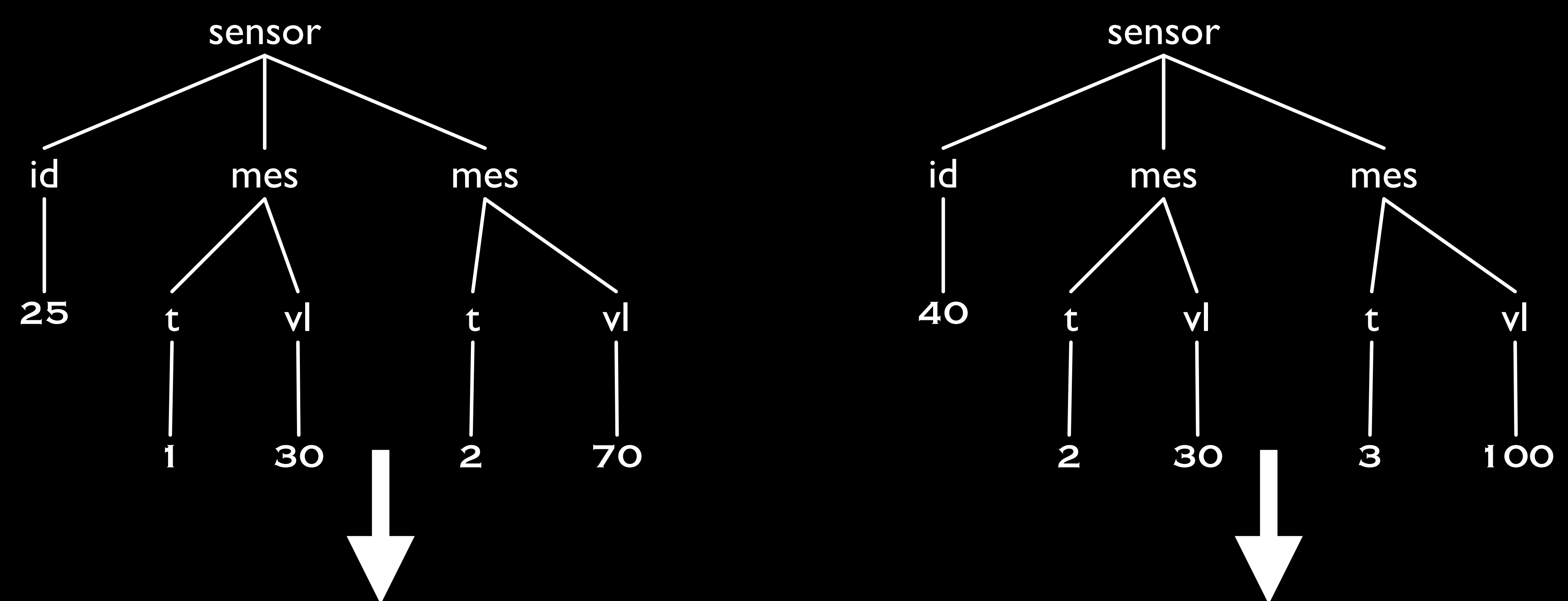
S:



Measuring Infinite Sets of Trees

1. Take a set S of trees with
   - real values on the leaves / share the same structure
2. collect labels of leaves as tuples of values
   $\Rightarrow$ Subset L(S) of R^n

S:

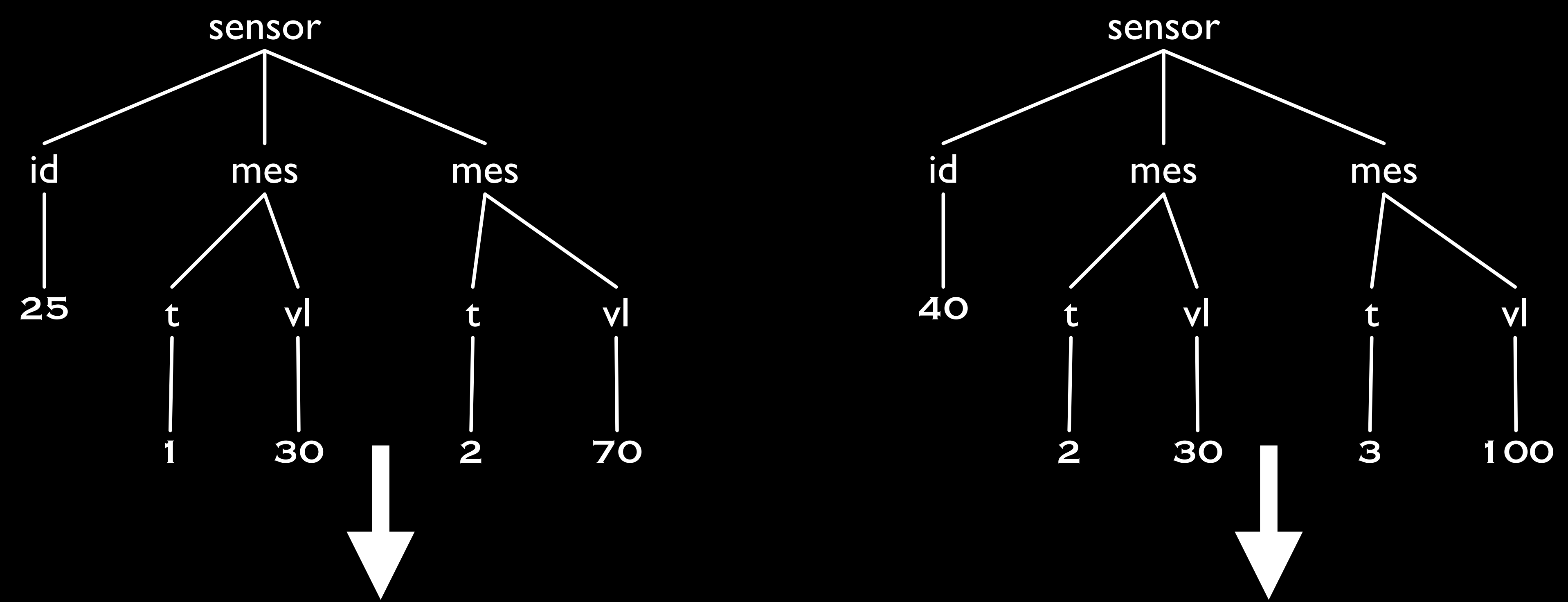


L(S): (25, 1, 30, 2, 70) (40, 2, 30, 3, 100)

Measuring Infinite Sets of Trees

3. Take a standard measure M on Borel subsets of R^n
4. Use the measure M on L(S)
5. Lift M from sets of tuples L(S) to sets of trees S

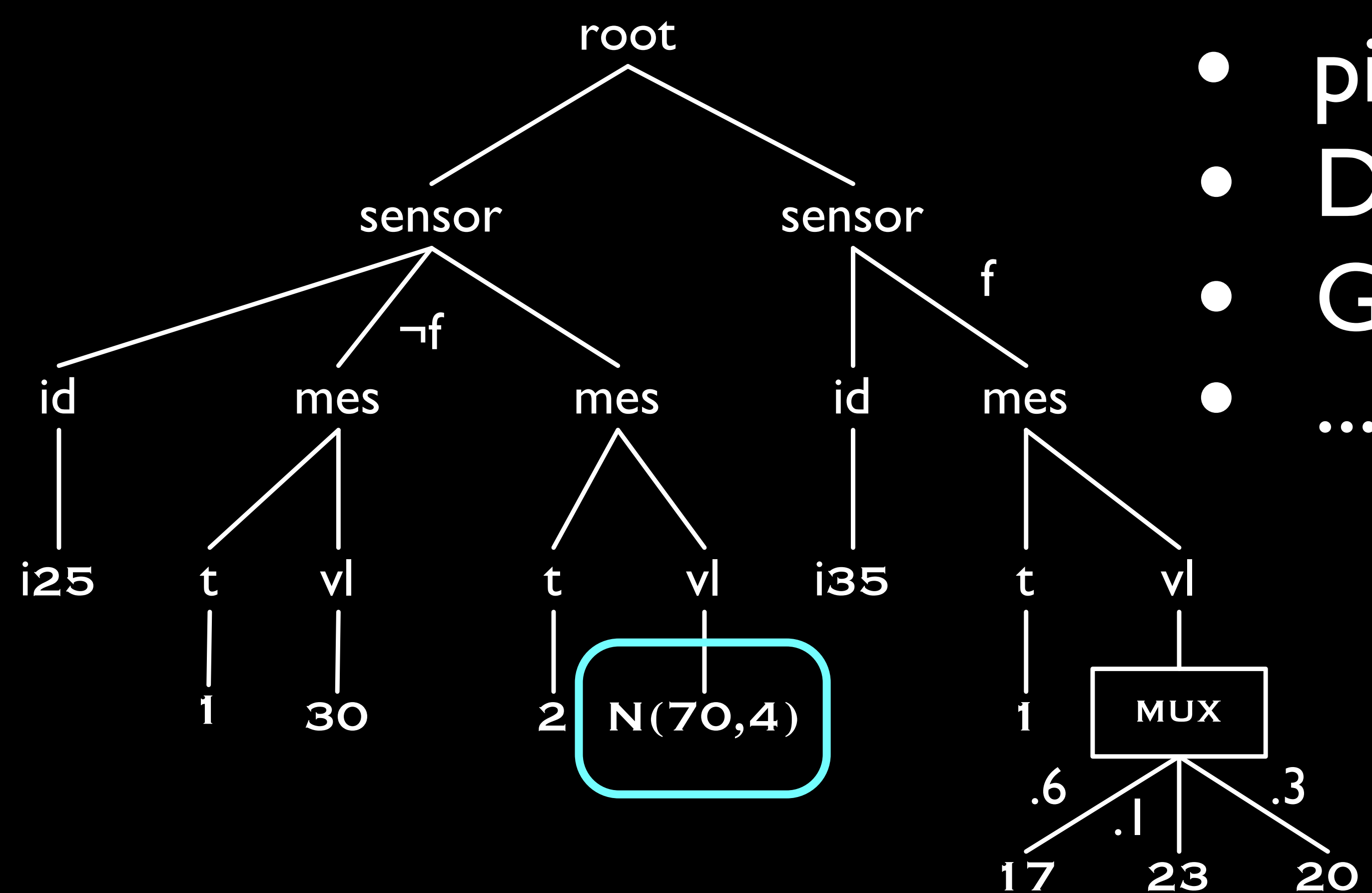
S:



L(S): (25, 1, 30, 2, 70) (40, 2, 30, 3, 100)

Continuous PXML Documents

- Extension of discrete PXML with distribution functions attached to leaves



- piecewise polynomials
- Diracs
- Gaussian
- ...

Aggregation of CPXML: Probability Computation

- Tractable for
  1. Data: CPXML with distributional nodes
  2. Query: SP with monoid functions
- Bottom-up algorithms based on convex sums and convolutions
- Works when distributions on the leaves are closed under convolutions and convex sums
  - piecewise polynomials (SUM, MIN/MAX) PTIME

Summing Up

- Comprehensive picture of complexity for discrete PXML aggregation:
  - PXML models with local, global dependencies
  - SP, TP, TPJ queries
  - COUNT, SUM, MIN, COUNTD, AVG
- Continuous PXML model:
  - formal semantics
  - initial study of aggregation

- Thank you

References

- [Kimelfeld&al:2007] - Benny Kimelfeld, Yehoshua Sagiv: Matching Twigs in Probabilistic XML. VLDB 2007: 27-38
- [Senellart&al:2007] -Pierre Senellart, Serge Abiteboul: On the complexity of managing probabilistic XML data. PODS 2007: 283-292
- [Re&al:2007] - C. Ré and D. Suciu. Efficient evaluation of HAVING queries on a probabilistic database. DBPL 2007
- [Cohen&al:2008] - Sara Cohen, Benny Kimelfeld, Yehoshua Sagiv: Incorporating constraints in probabilistic XML. PODS 2008: 109-118