

# Aggregate Queries for Discrete and Continuous Probabilistic XML

S. Abiteboul,<sup>1</sup> T-H. H. Chan,<sup>2</sup> E. Kharlamov,<sup>1,3</sup> W. Nutt,<sup>3</sup> P. Senellart<sup>4</sup>

<sup>1</sup>INRIA Saclay – Île-de-France    <sup>3</sup>Free University of Bozen-Bolzano

<sup>2</sup>The University of Hong Kong    <sup>4</sup>Télécom ParisTech

ICDT, March 2010

# Outline

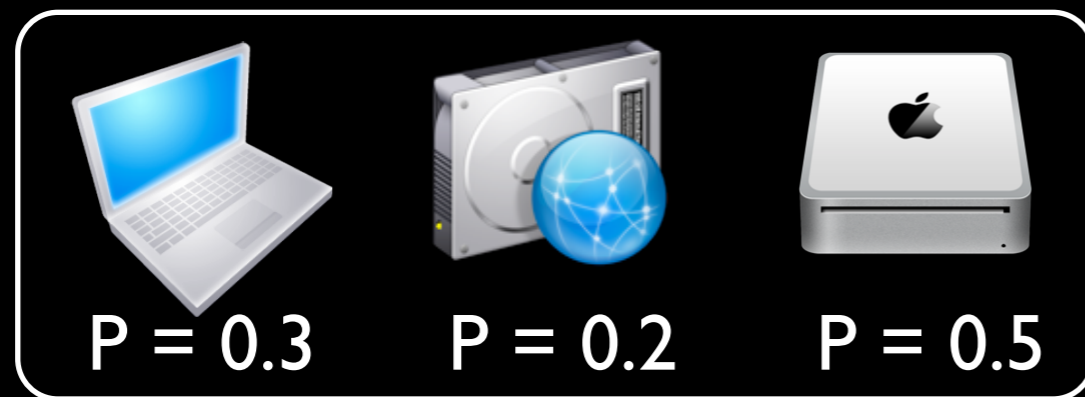
1. Probabilistic data
2. Problem definition
3. Aggregating discrete Probabilistic XML
4. Aggregating continuous Probabilistic XML

# 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
- ...

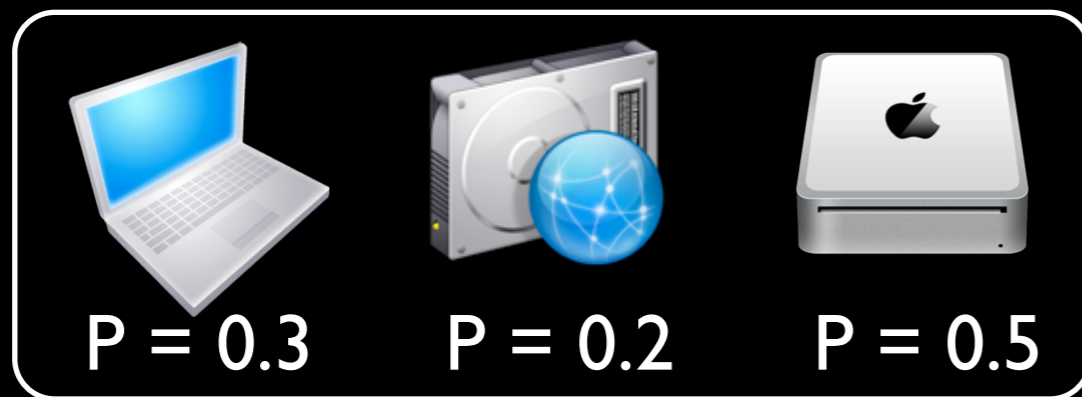
# Probabilistic Database

Probabilistic DB:



# Probabilistic Database

Probabilistic DB:



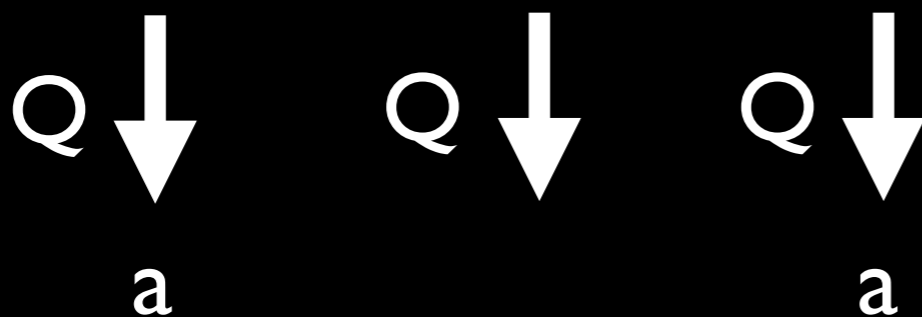
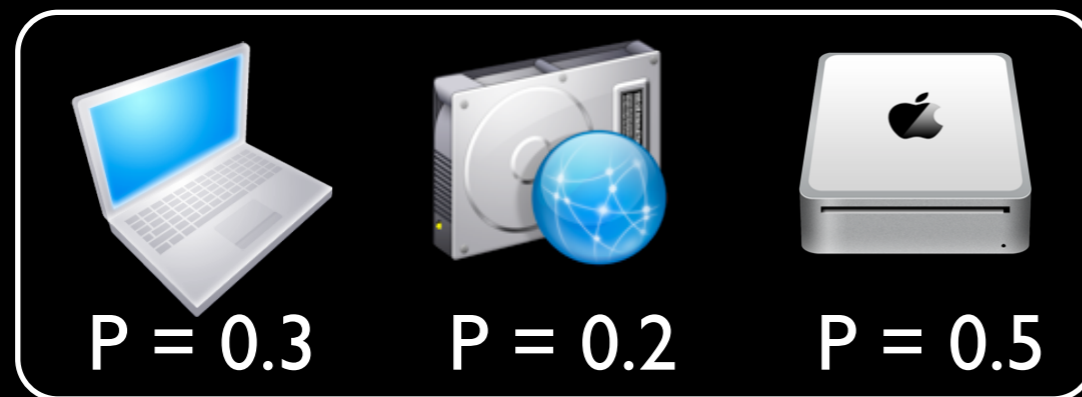
Q ↓  
a

Q ↓

Q ↓  
a

# Probabilistic Database

Probabilistic DB:

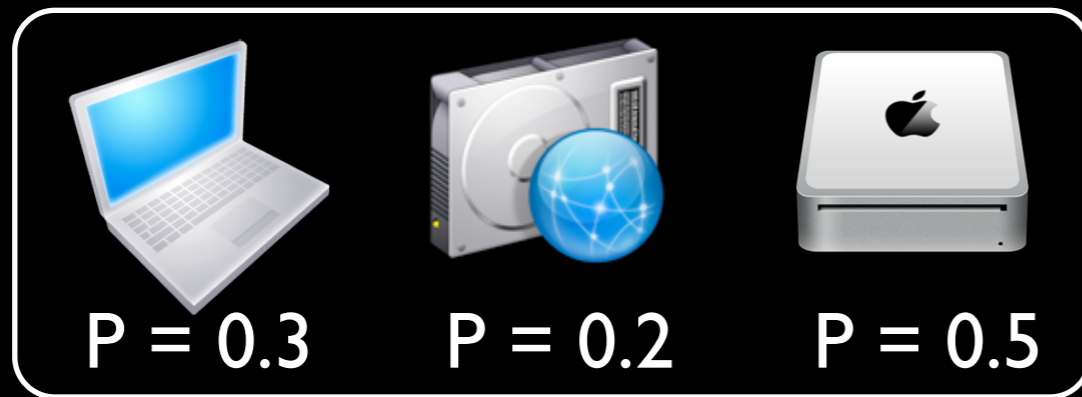


---

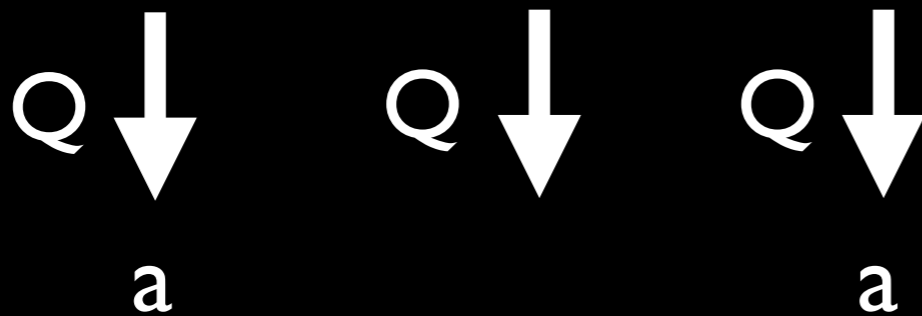
Answer:  $(a, 0.8)$

# Probabilistic Database

Probabilistic DB:



Representation  
of Prob DB:

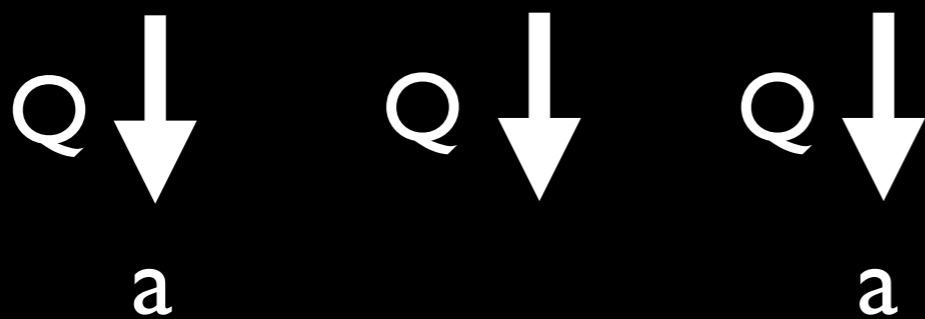
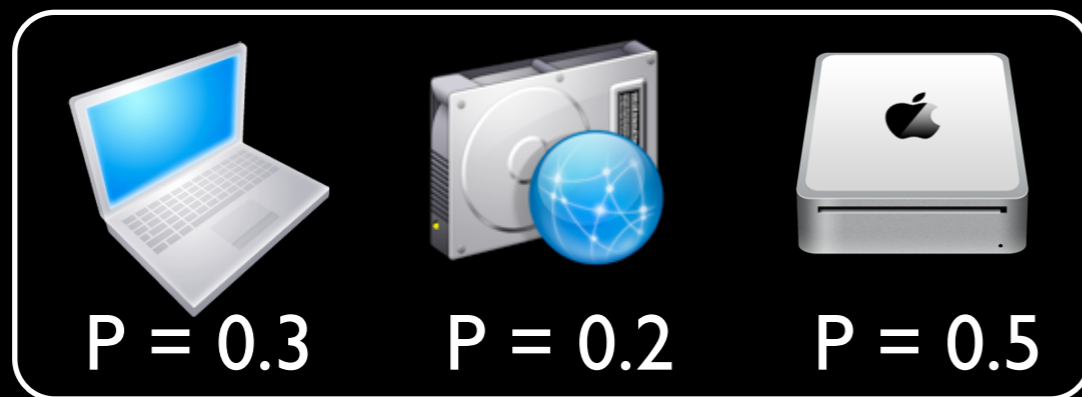


---

Answer: (a, 0.8)

# Probabilistic Database

Probabilistic DB:



---

Answer: (a, 0.8)

Representation  
of Prob DB:

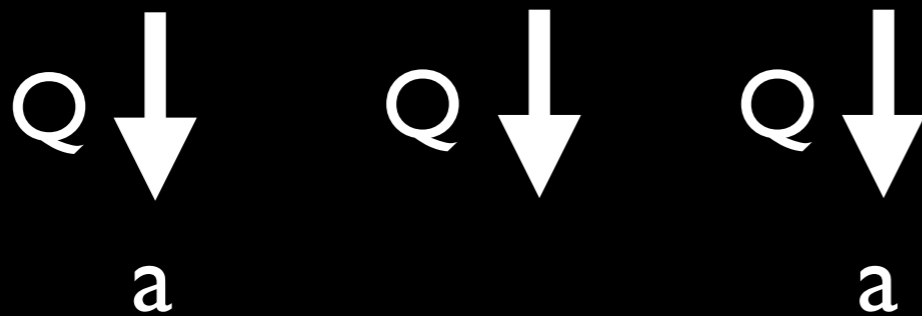
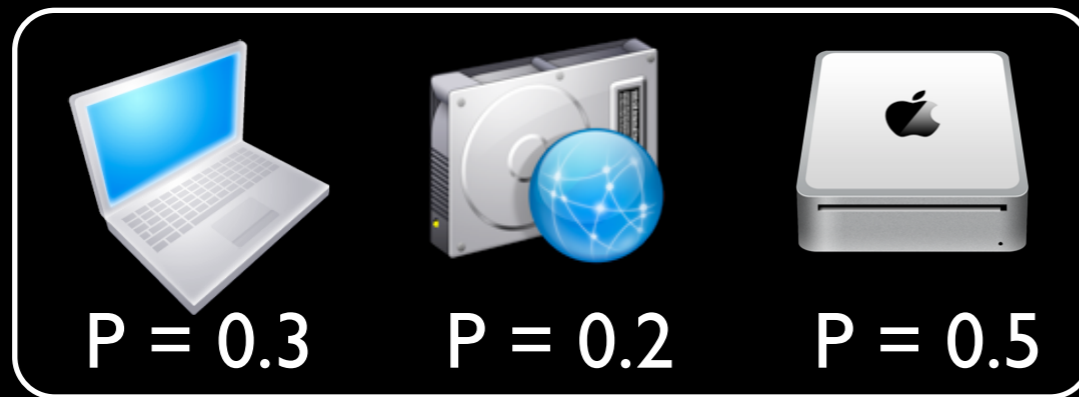


Q

(a, 0.8)

# Probabilistic Database

Probabilistic DB:



---

Answer: (a, 0.8)

Representation  
of Prob DB:



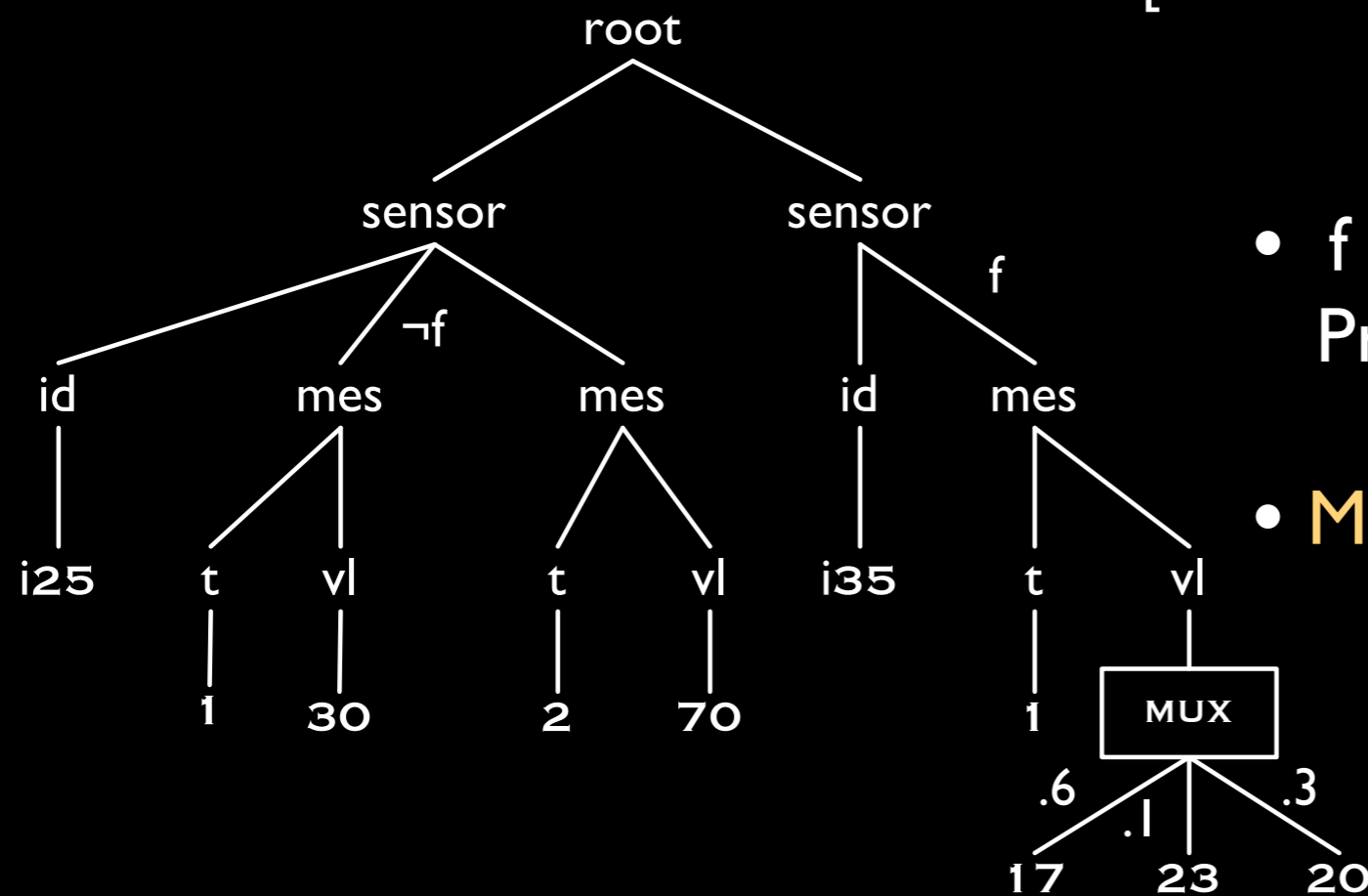
Q

(a, 0.8)

# PXML with Events and Distributional Nodes

[Kimelfed&al:2007]

[Senellart&al:2007]



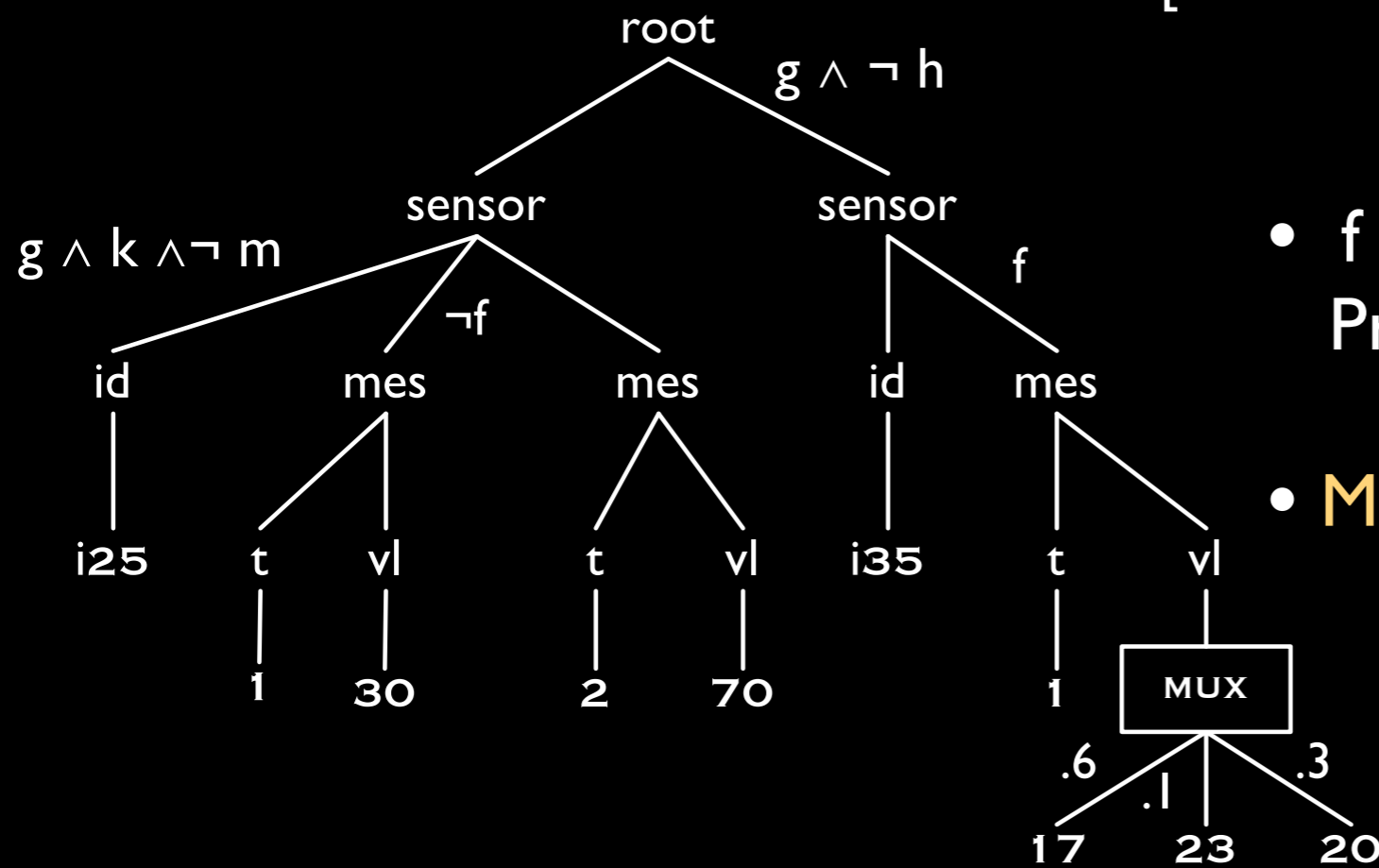
- **f - event**: “weather is fine”  
 $\Pr(f) = .4$

- **MUX** - mutually exclusive options

# PXML with Events and Distributional Nodes

[Kimelfed&al:2007]

[Senellart&al:2007]

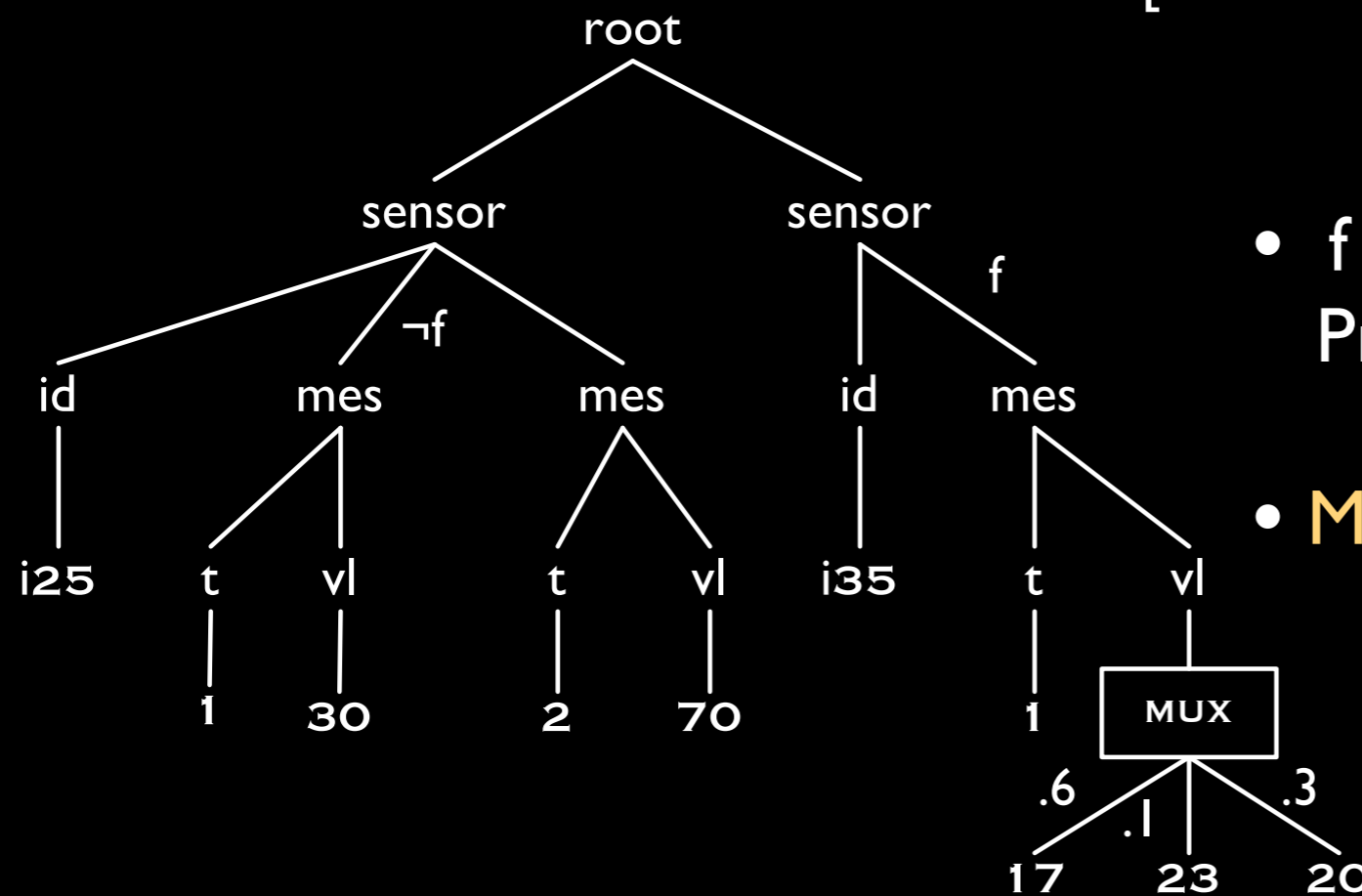


- **f - event**: “weather is fine”  
Pr(f) = .4
- **MUX** - mutually exclusive options

# PXML with Events and Distributional Nodes

[Kimelfed&al:2007]

[Senellart&al:2007]



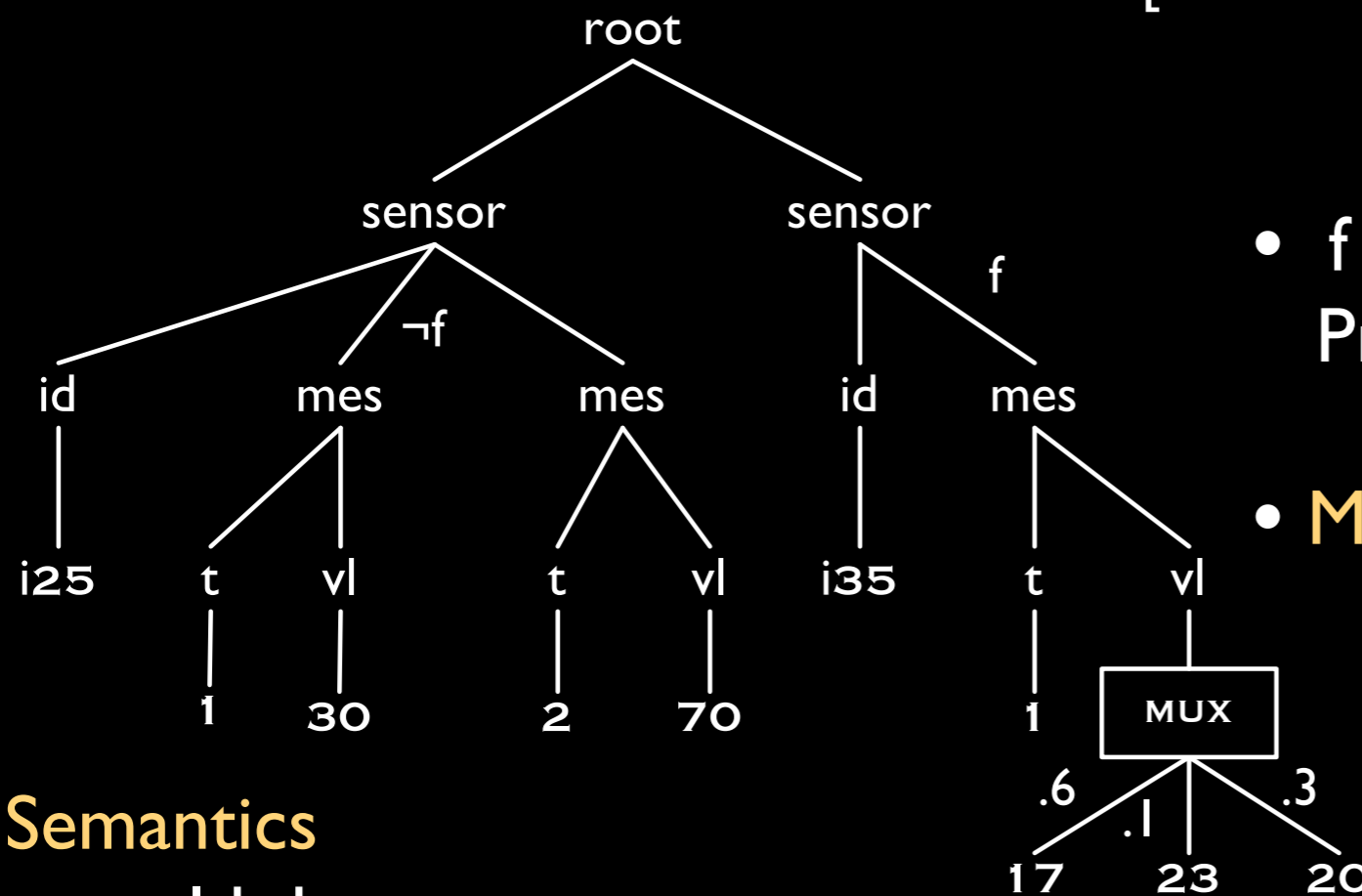
- **f - event**: “weather is fine”  
 $\Pr(f) = .4$

- **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  $d$ :

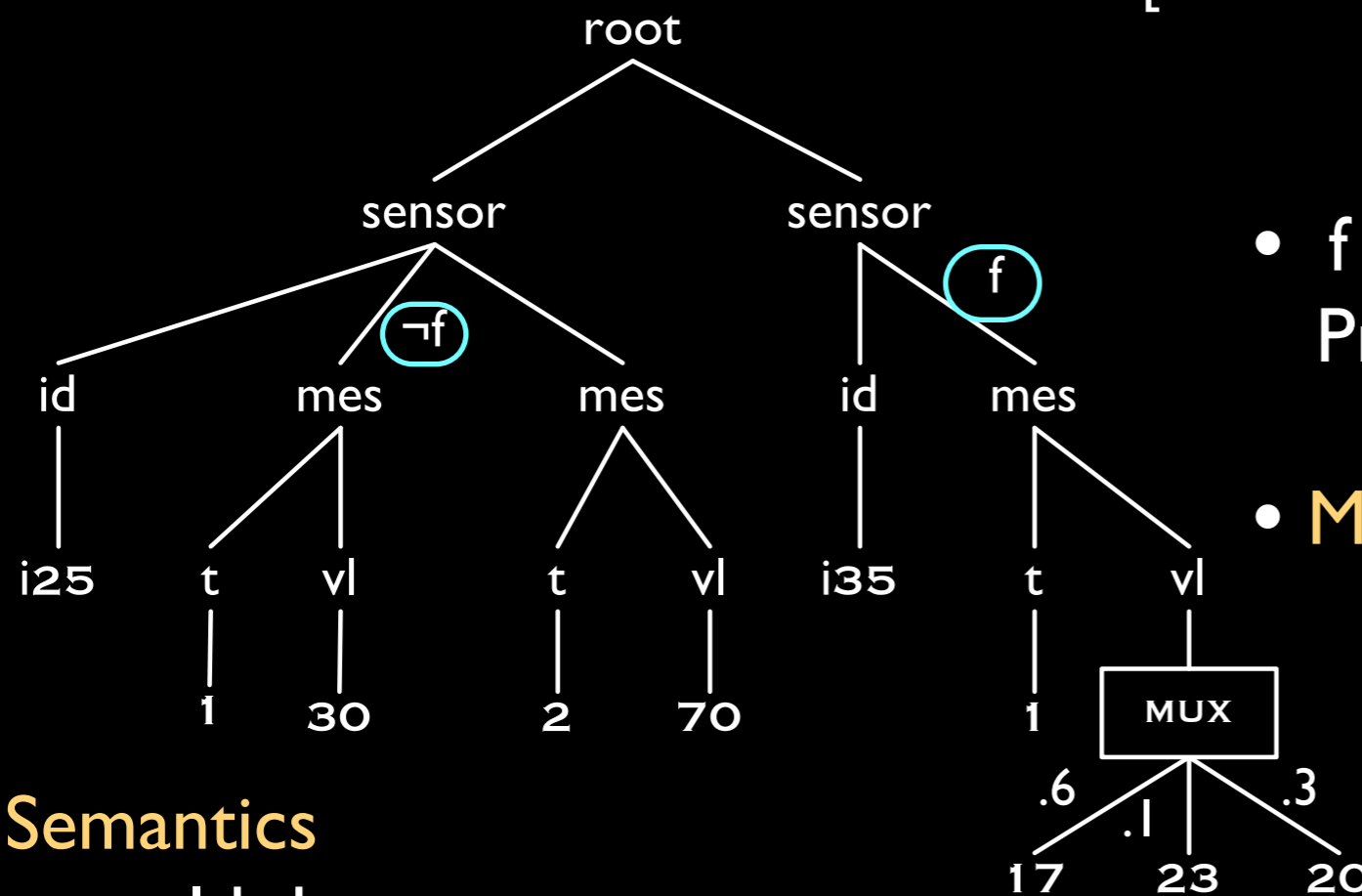
- $f = \text{true}$ ,  $\Pr(f) = 0.4$
- **MUX**: 23,  $\Pr(23) = 0.1$

$$\Pr(d) = 0.4 \times 0.1$$

# 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  $d$ :

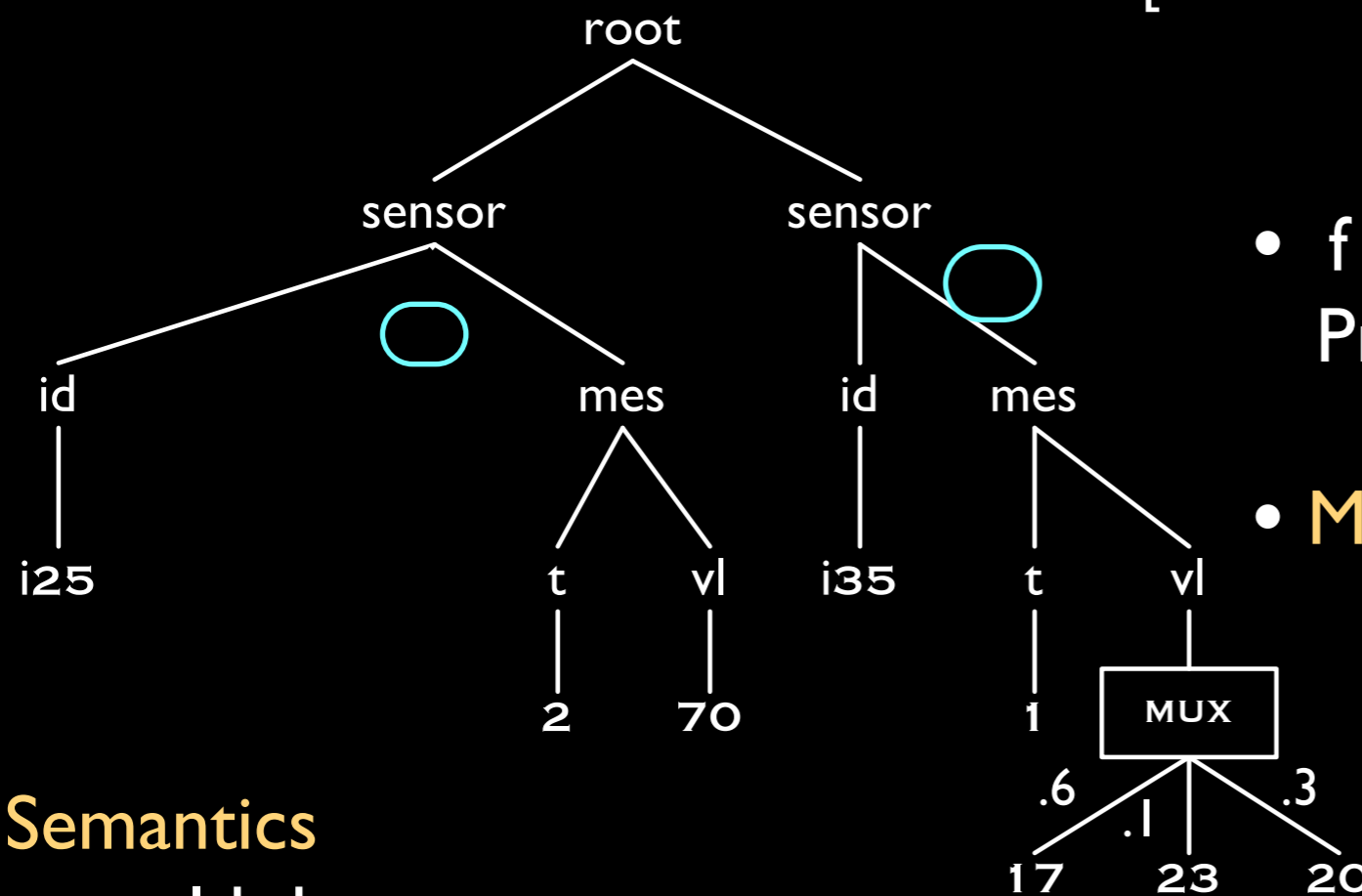
- **f = true**,  $\Pr(f) = 0.4$
- **MUX: 23**,  $\Pr(23) = 0.1$

$$\Pr(d) = 0.4 \times 0.1$$

# 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  $d$ :

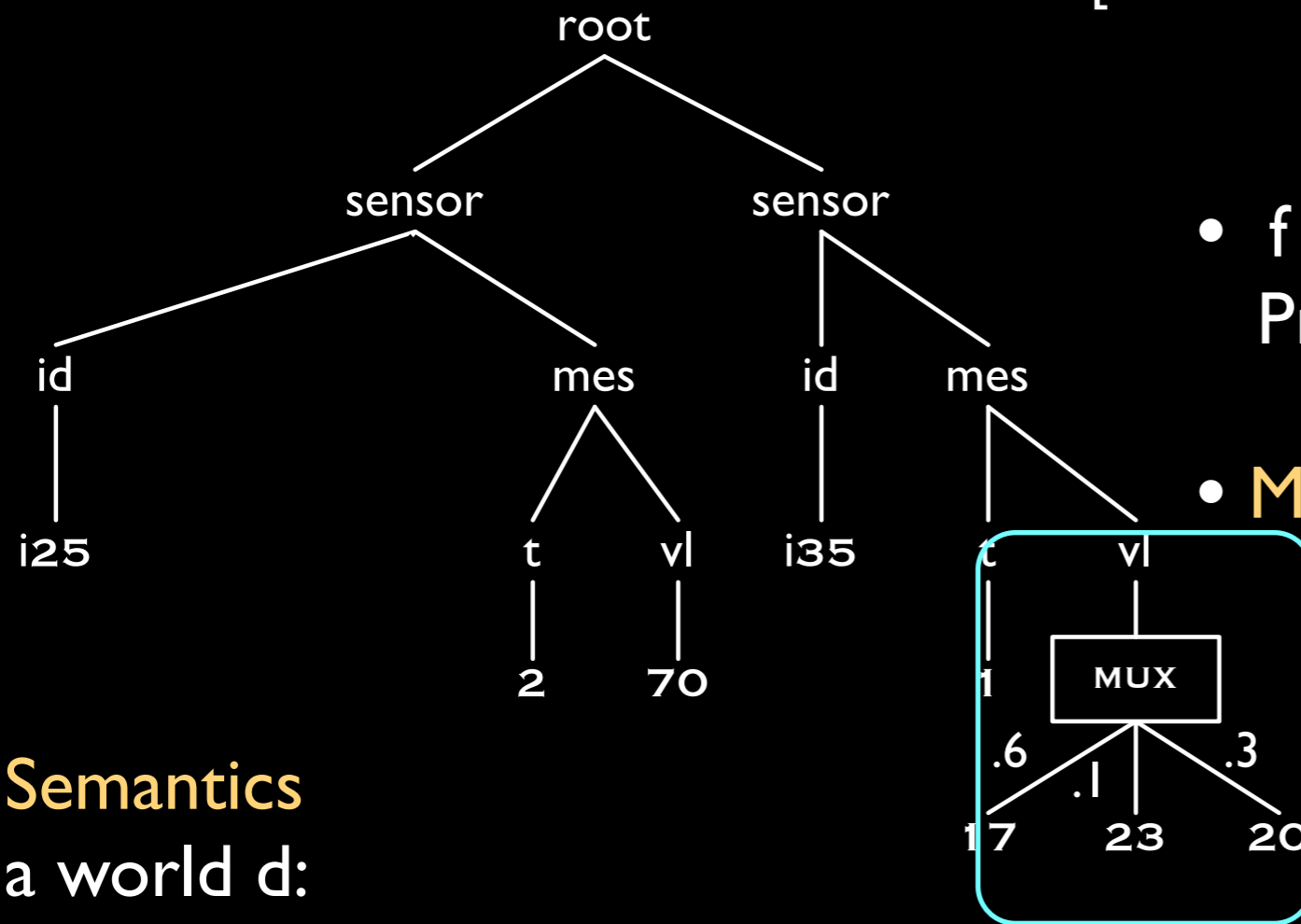
- **f = true**,  $\Pr(f) = 0.4$
- **MUX: 23**,  $\Pr(23) = 0.1$

$$\Pr(d) = 0.4 \times 0.1$$

# 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  $d$ :

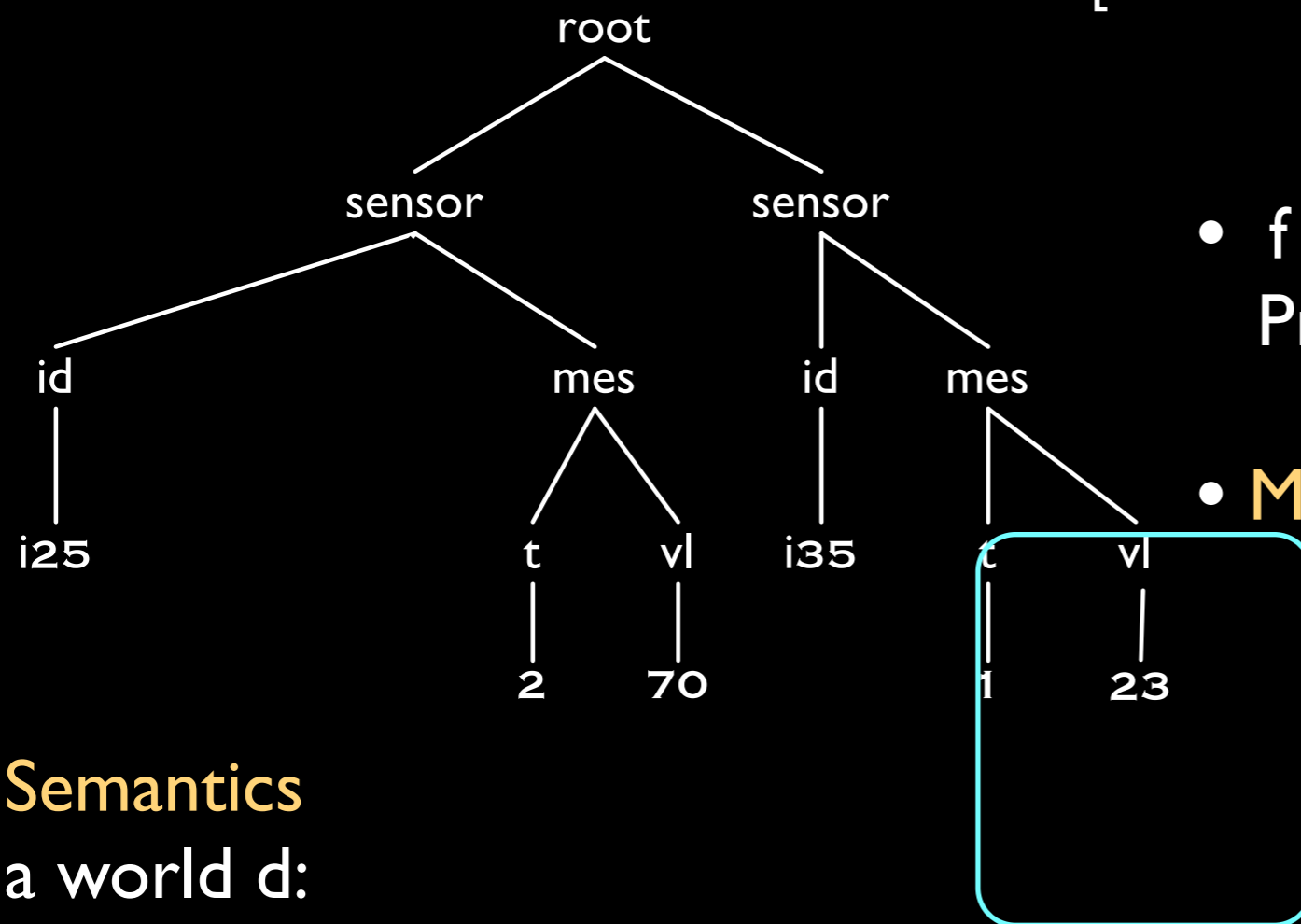
- $f = \text{true}$ ,  $\Pr(f) = 0.4$
- **MUX: 23**,  $\Pr(23) = 0.1$

$$\Pr(d) = 0.4 \times 0.1$$

# 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  $d$ :

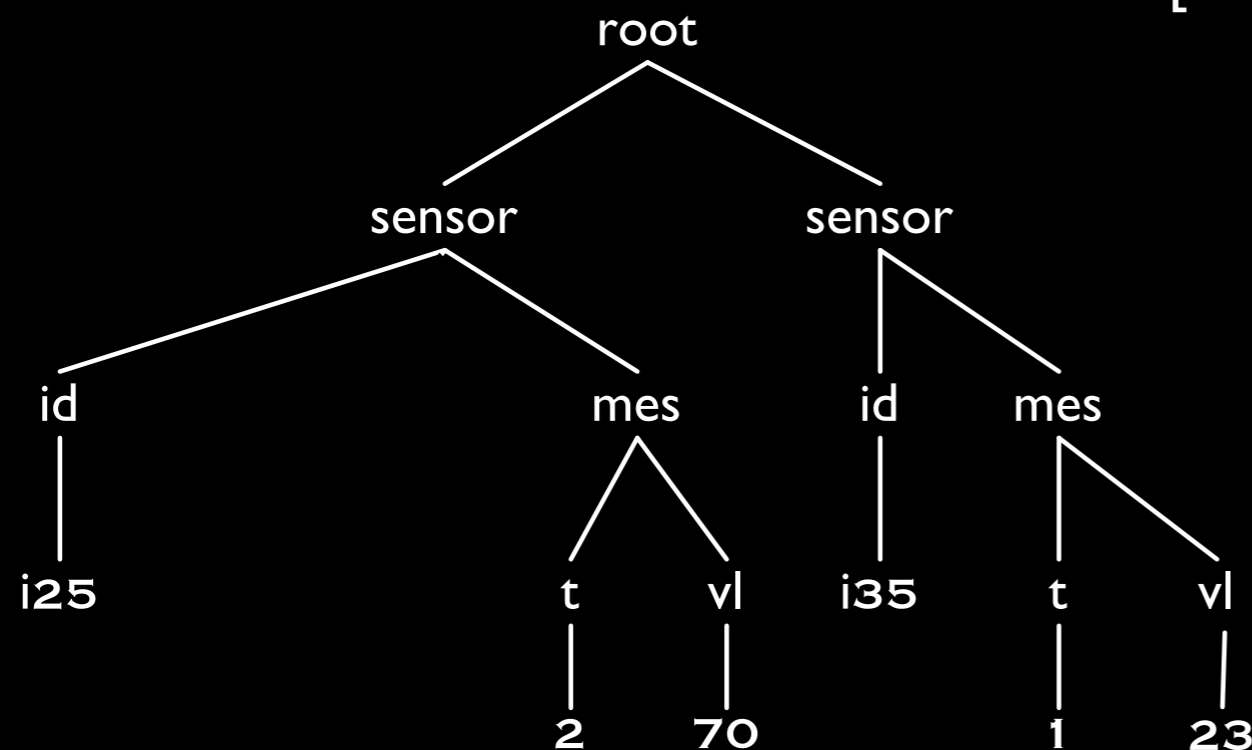
- $f = \text{true}$ ,  $\Pr(f) = 0.4$
- **MUX: 23**,  $\Pr(23) = 0.1$

$\Pr(d) = 0.4 \times 0.1$

# 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  $d$ :

- $f = \text{true}$ ,  $\Pr(f) = 0.4$
- **MUX**: 23,  $\Pr(23) = 0.1$

$\Pr(d) = 0.4 \times 0.1$

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

# What is Known?

- Answering simple XPath queries [Kimelfed&al:2007]  
[Senellart&al:2007]
- Distributional nodes: PTIME
- Events:  $FP^{\#P}$ -complete
- Simple XPath over Mux-Det PXML with HAVING constraints: [Cohen&al:2008]  
[Re&al:2007]
- PTIME for COUNT and MIN
- NP-hard for SUM and AVG

# What is Known?

- Answering simple XPath queries [Kimelfed&al:2007]  
[Senellart&al:2007]
- Distributional nodes: PTIME
- Events:  $FP^{\#P}$ -complete
- Simple XPath over Mux-Det PXML with HAVING constraints: [Cohen&al:2008]  
[Re&al:2007]
- 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

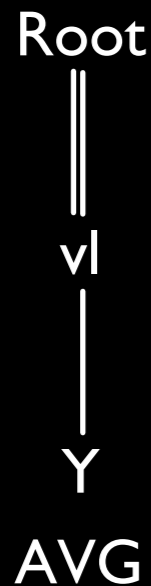
# Aggregate Queries

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?
- ⇒ we want to answer queries with **aggregate** functions:  
MIN/MAX, TopK, COUNT, SUM, COUNTD, AVG

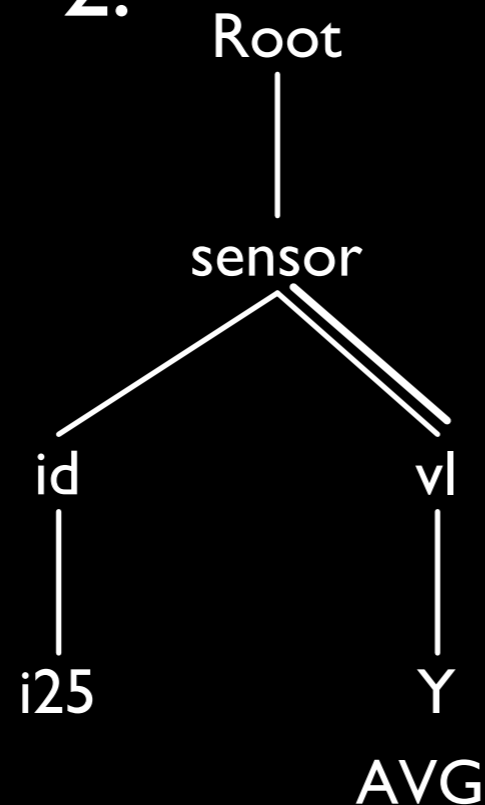
# Query Models

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?

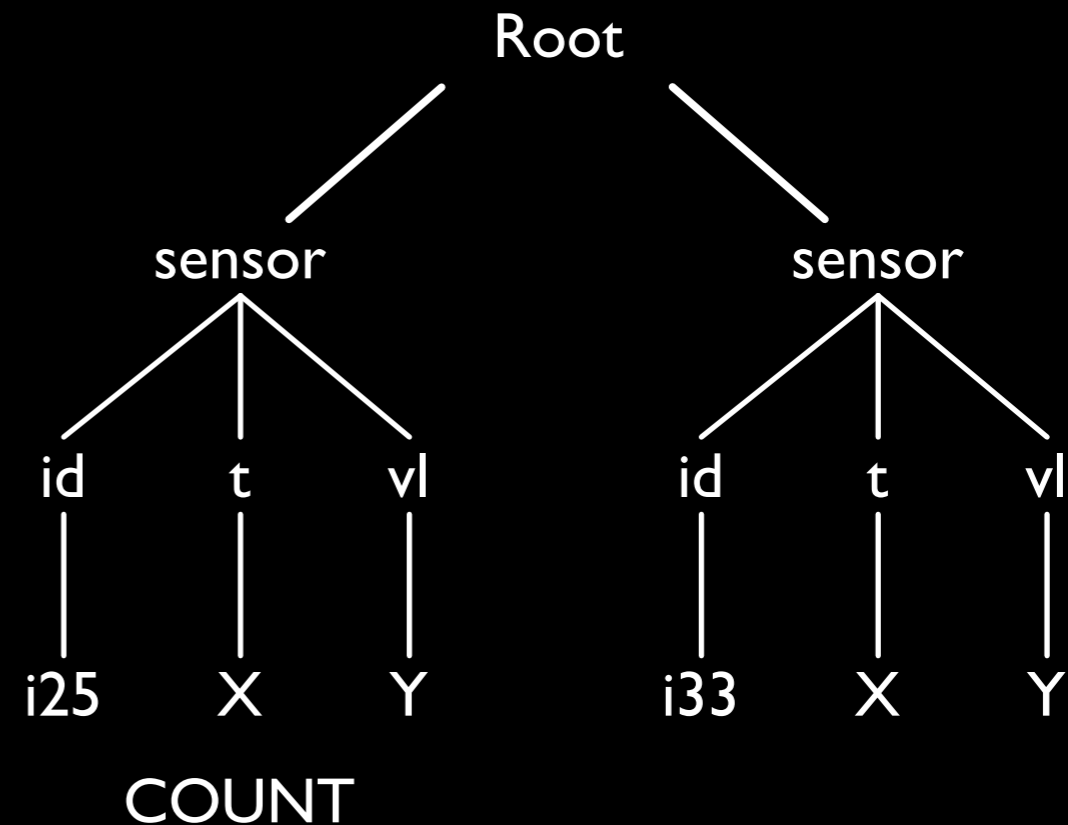
1.



2.



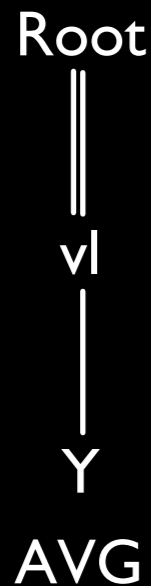
3.



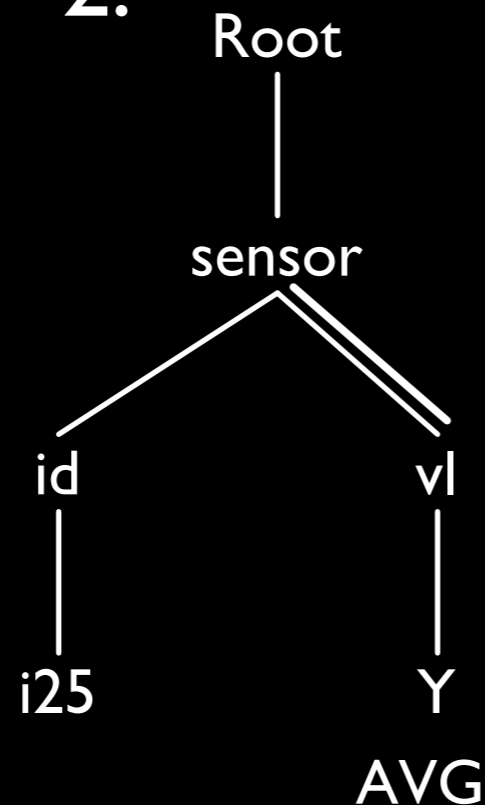
# Query Models

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?

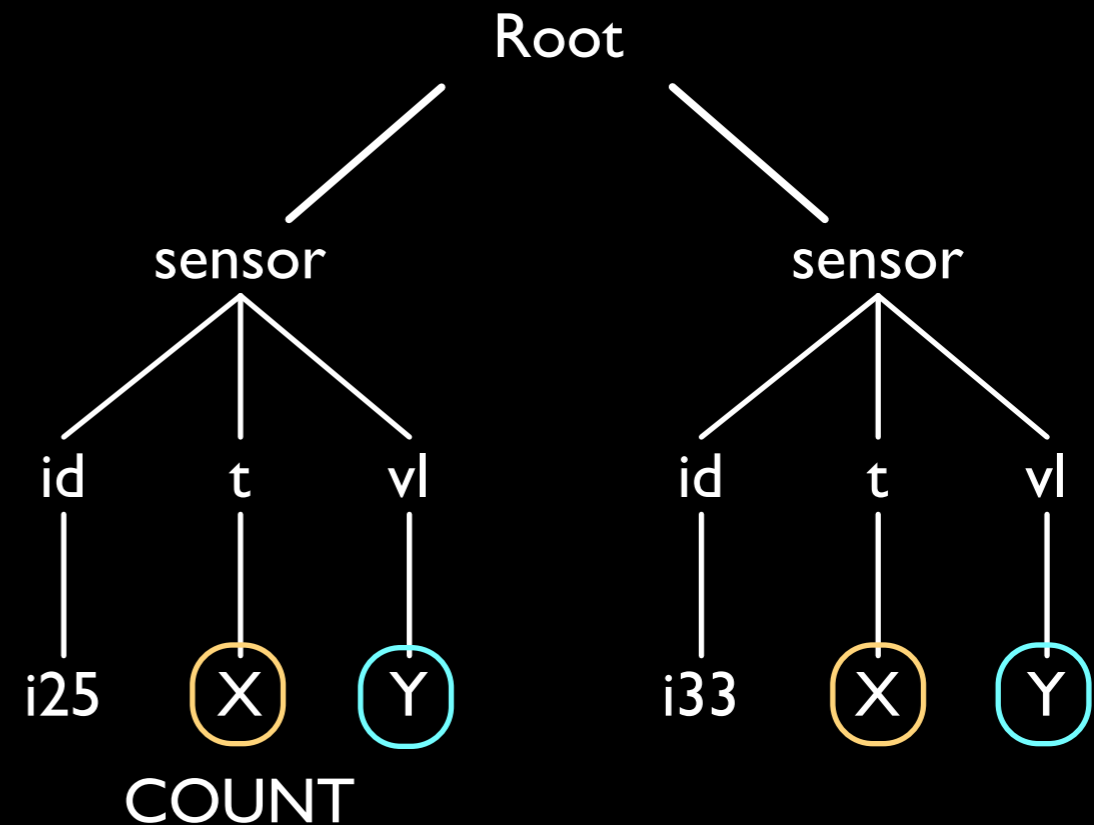
1.



2.



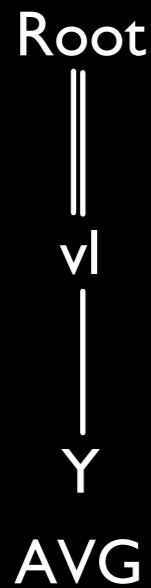
3.



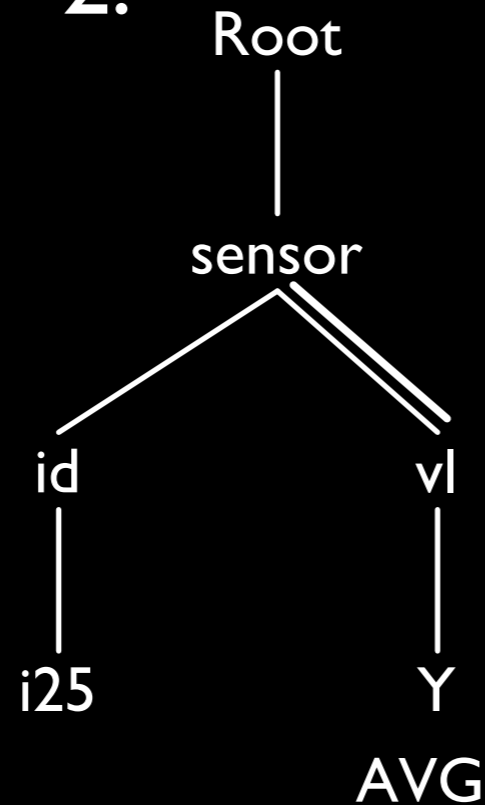
# Query Models

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

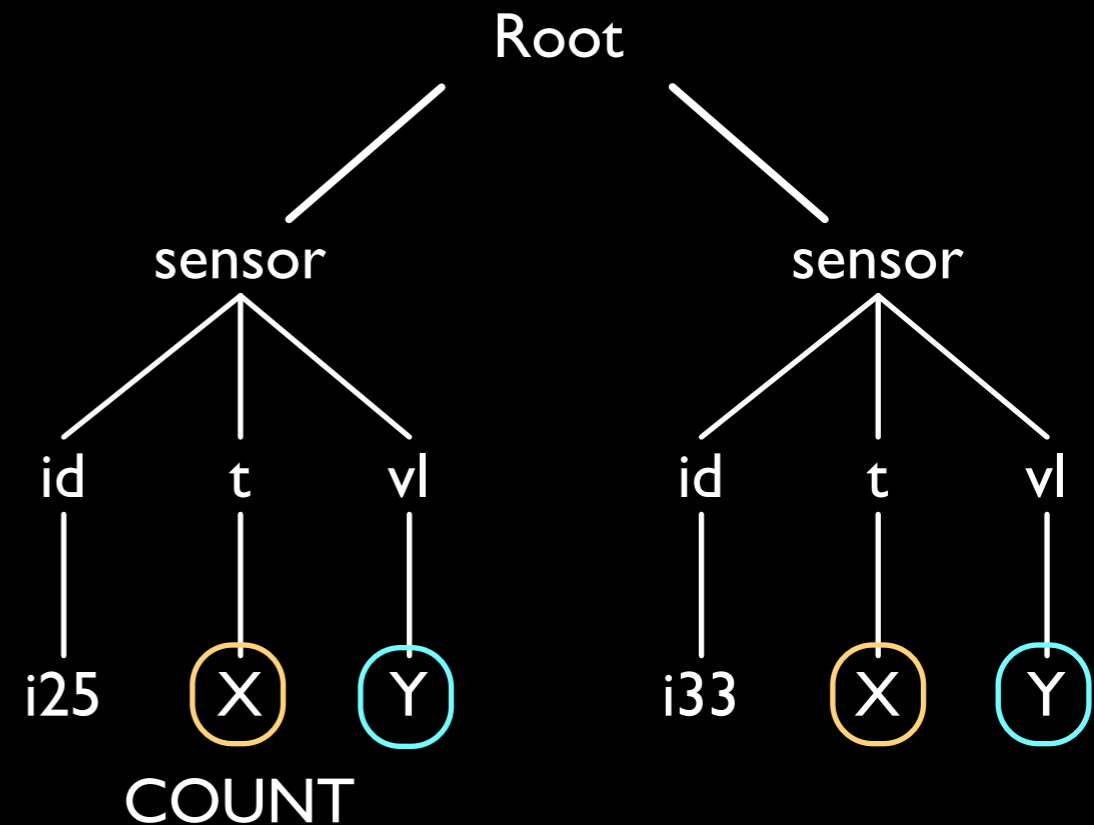
1.



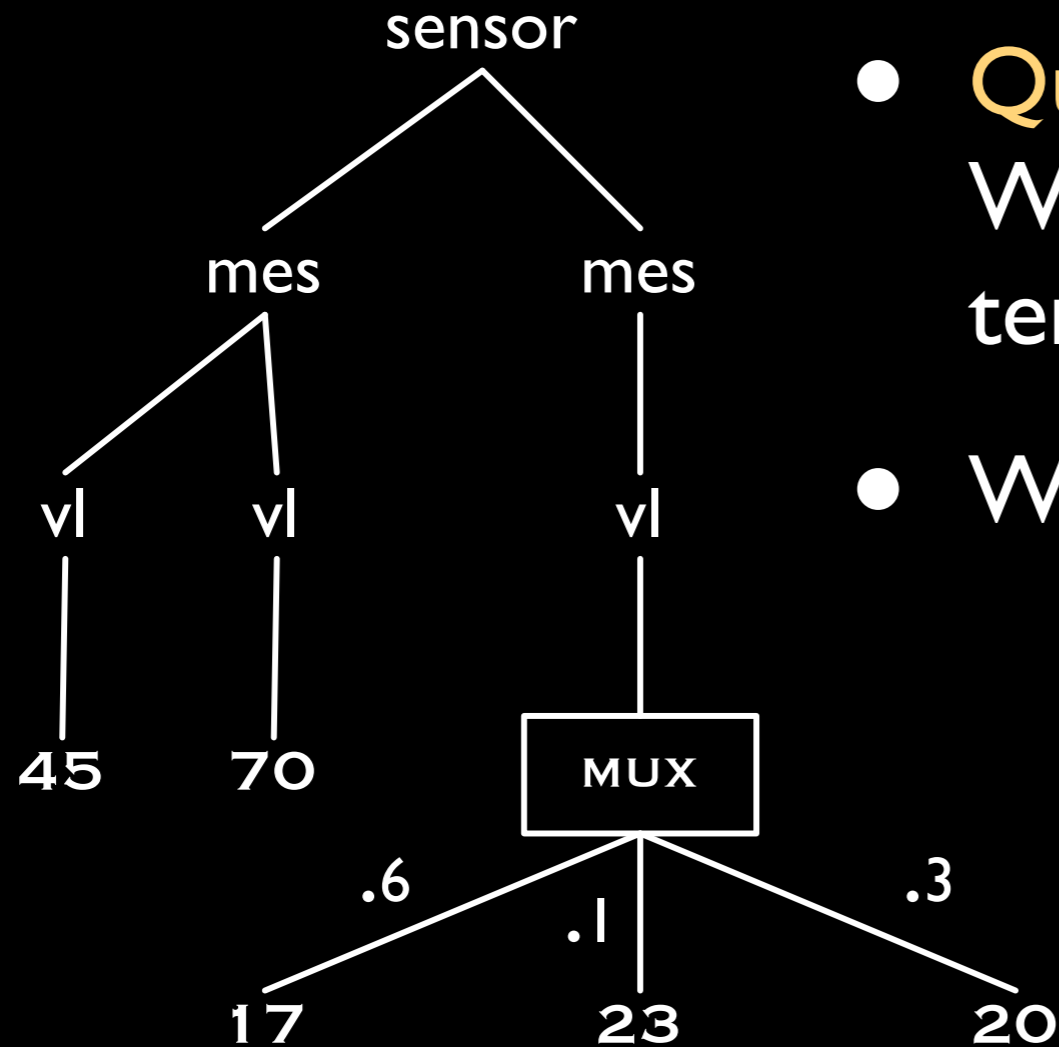
2.



3.



# Semantics of AQs



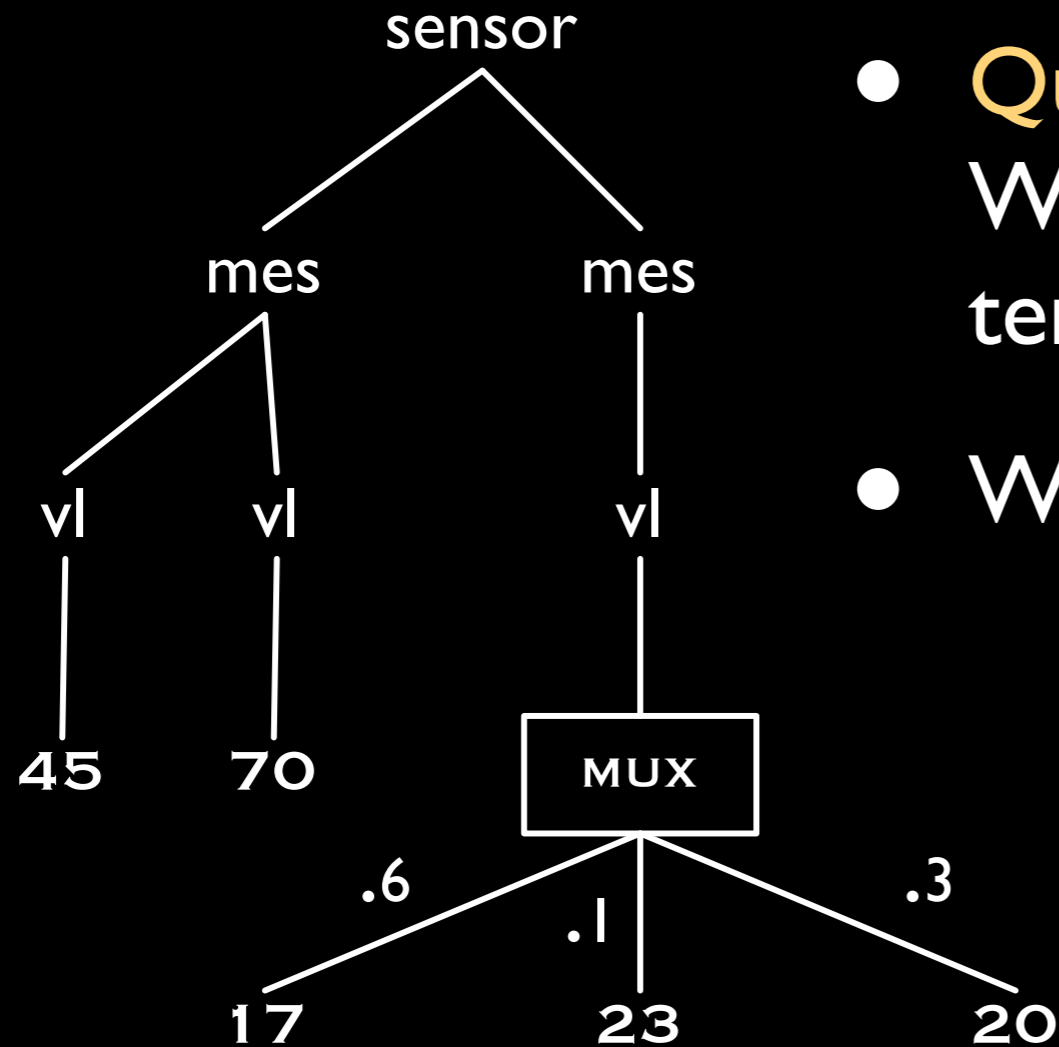
- **Query:**  
What is the **average** temperature?
- What should be an **answer**?

$$\text{AVG}(d17) = 44, \text{Pr}(d17) = .6$$

$$\text{AVG}(d23) = 46, \text{Pr}(d23) = .1$$

$$\text{AVG}(d20) = 45, \text{Pr}(d20) = .3$$

# Semantics of AQs



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

$$\text{AVG}(d17) = 44, \text{Pr}(d17) = .6$$

$$\text{AVG}(d23) = 46, \text{Pr}(d23) = .1$$

$$\text{AVG}(d20) = 45, \text{Pr}(d20) = .3$$

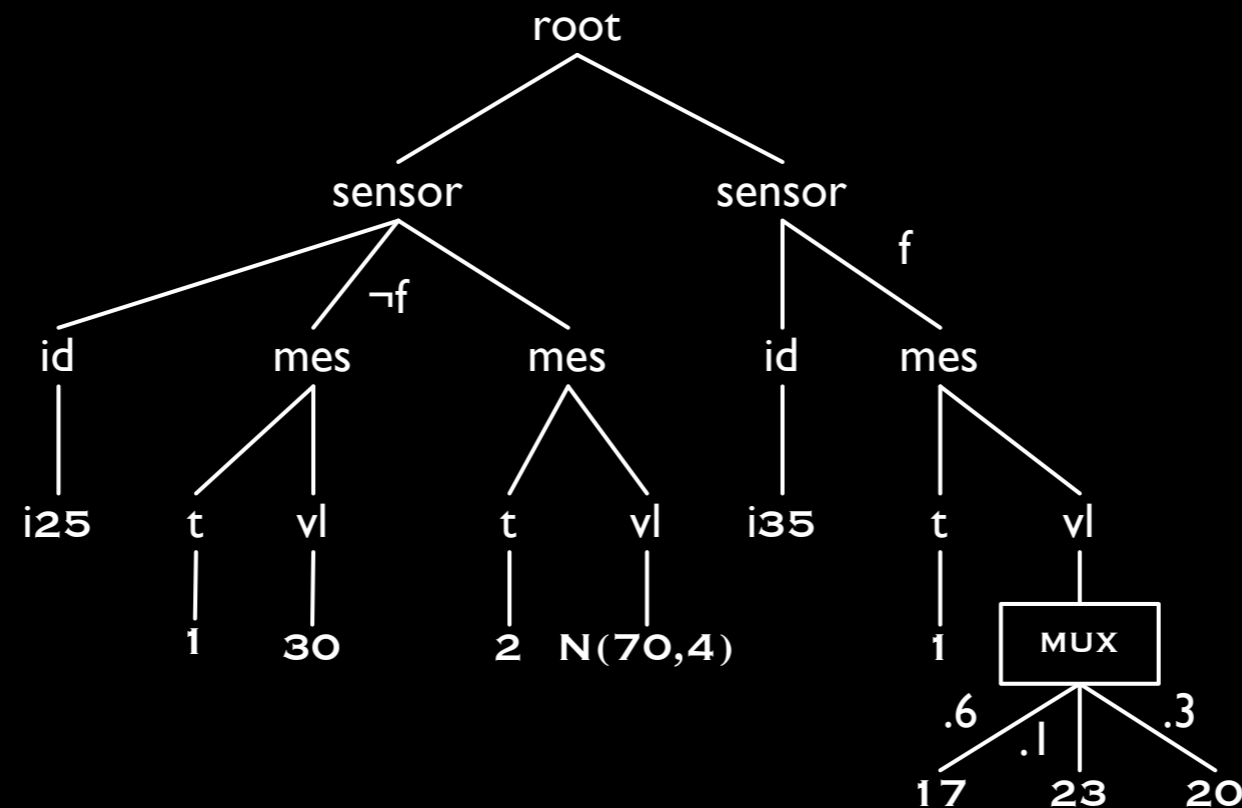
**Distribution** of aggregate values over all documents represented by the PXML document

# 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”

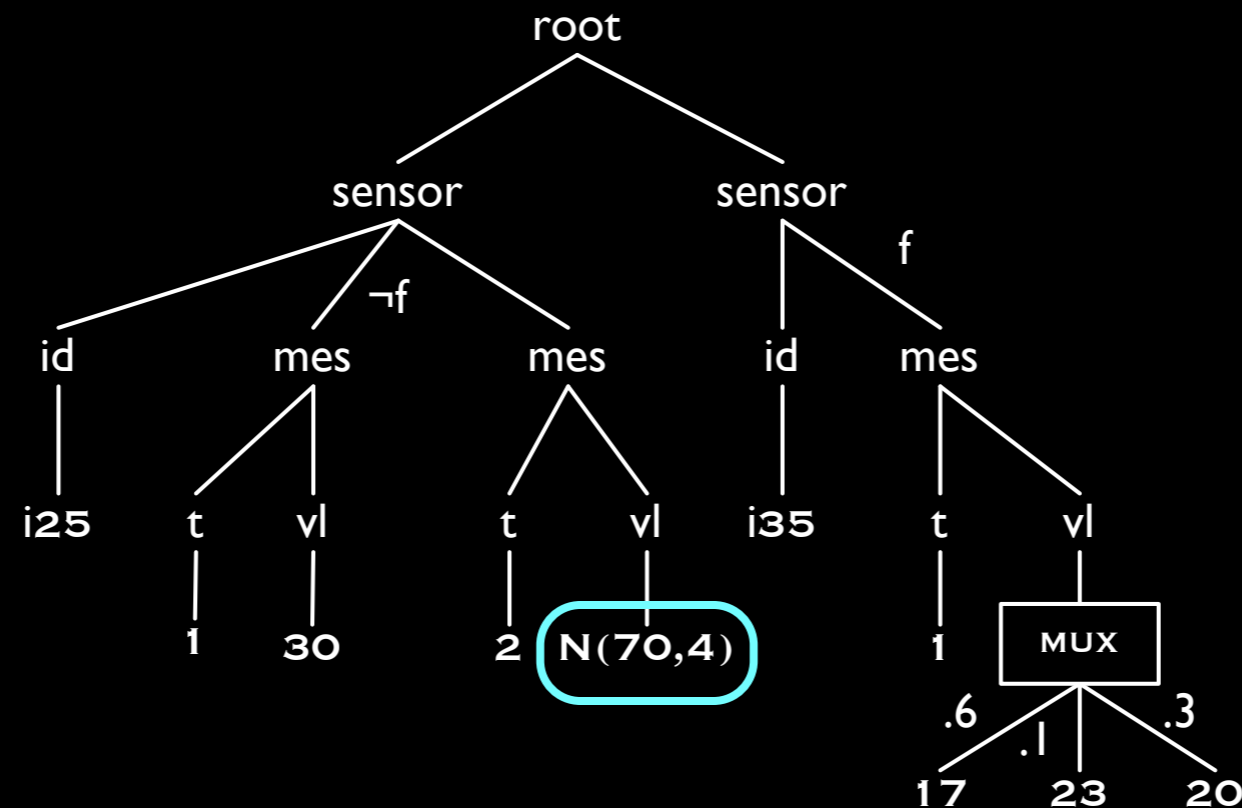
# Continuous PXML



- 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 <sup>#P</sup> -complete		
Moment Computation	COUNT, SUM: PTIME MIN, AVG COUNTD: FP <sup>#P</sup> -comp	FP <sup>#P</sup> -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 <sup>#P</sup> -complete		
Moment Computation	COUNT, SUM, <b>PTIME</b> MIN, AVG COUNTD: FP <sup>#P</sup> -comp	FP <sup>#P</sup> -complete	

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

# Aggregating PXML with Distributional Nodes

Aggregate Query Language			
Problems	Single Path	Tree Pattern	Tree Pat. Joins
Possible Answers	SUM,AVG, COUNTD: NP-complete		
Probability Computation	COUNT, MIN: PTIME		COUNT, MIN : NP
Probability SUM in  input  + output	SUM,AVG, COUNTD: FP <sup>#P</sup> -complete COUNT, MIN: PTIME		FP <sup>#P</sup> -complete
Moment Computation	PTIME	FP <sup>#P</sup> AVG: FP <sup>#P</sup> 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

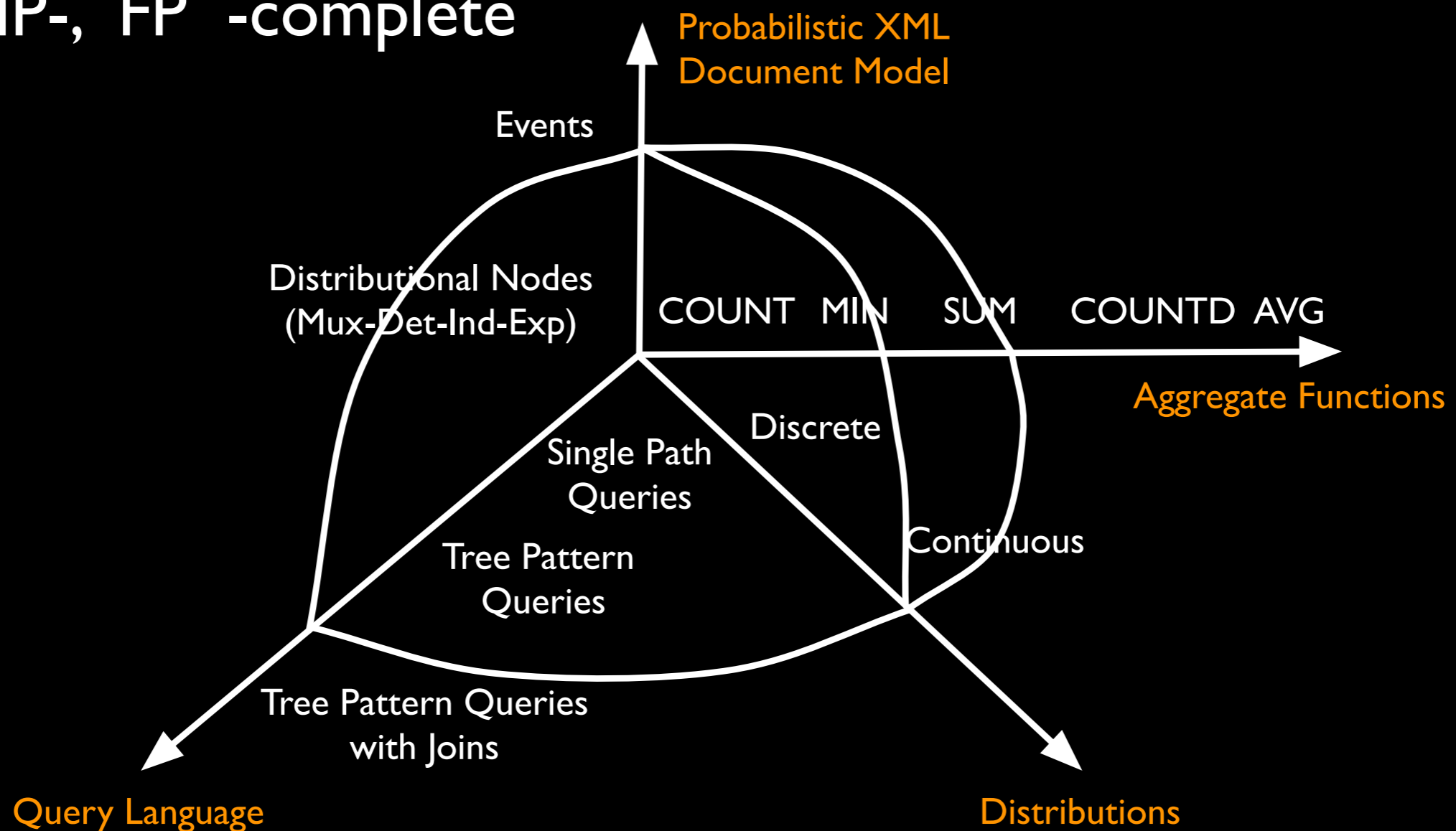
P-TIME algorithm to compute distributions:

**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  $\text{FP}^{\#P}$ -complete  
How good are **Monte-Carlo** methods?

- By Hoeffding bound, to achieve

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

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

$\Rightarrow$  for  $\alpha = \text{COUNTD}$

**quadratically** many samples are needed

# Approximating Query Answers

- Many problems are NP- or  $\text{FP}^{\#P}$ -complete  
How good are **Monte-Carlo** methods?

- By Hoeffding bound, to achieve

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

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

$\Rightarrow$  for  $\alpha = \text{COUNTD}$

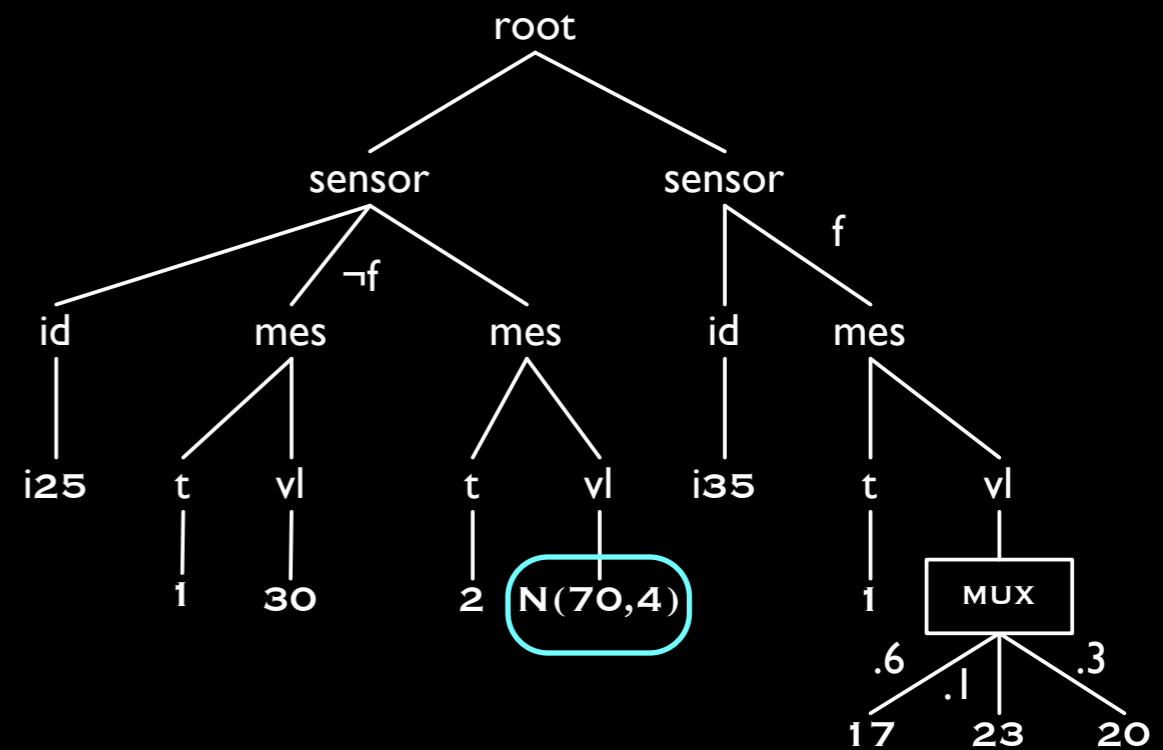
**quadratically** many samples are needed

# Outline

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

# Discrete vs Continuous Models

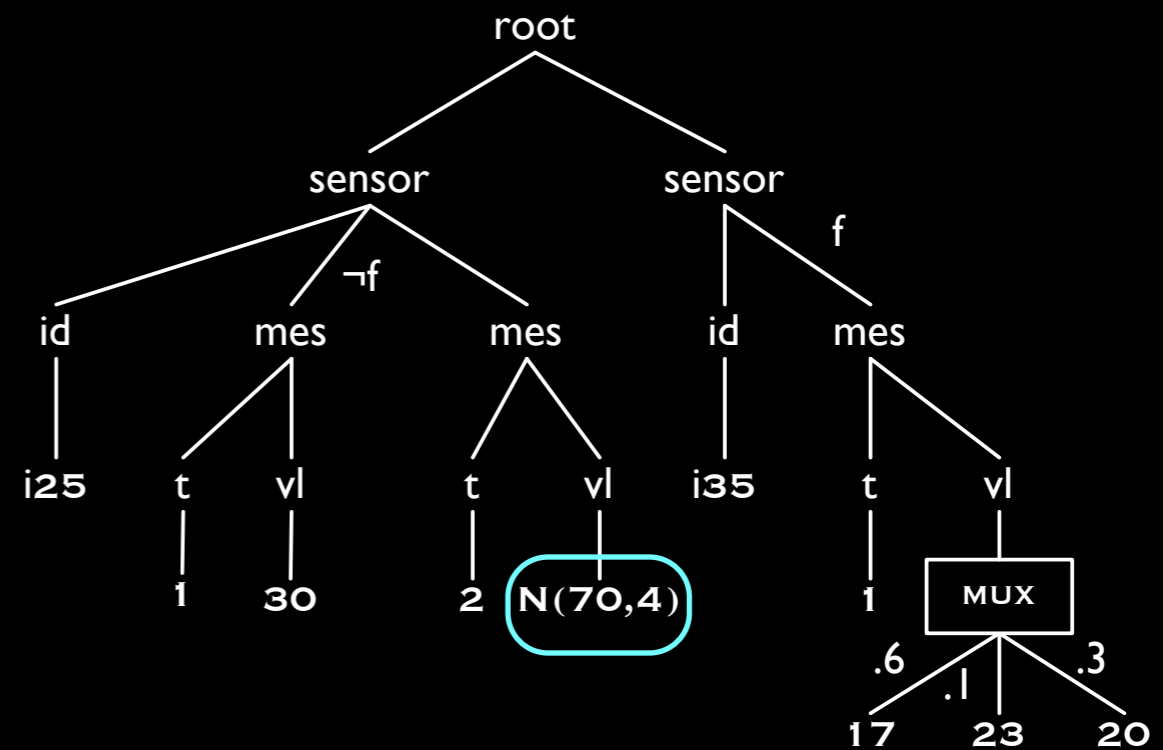
- Finite case:
  - **finite** sets of trees
  - where **every tree** has a non-zero probability



- 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

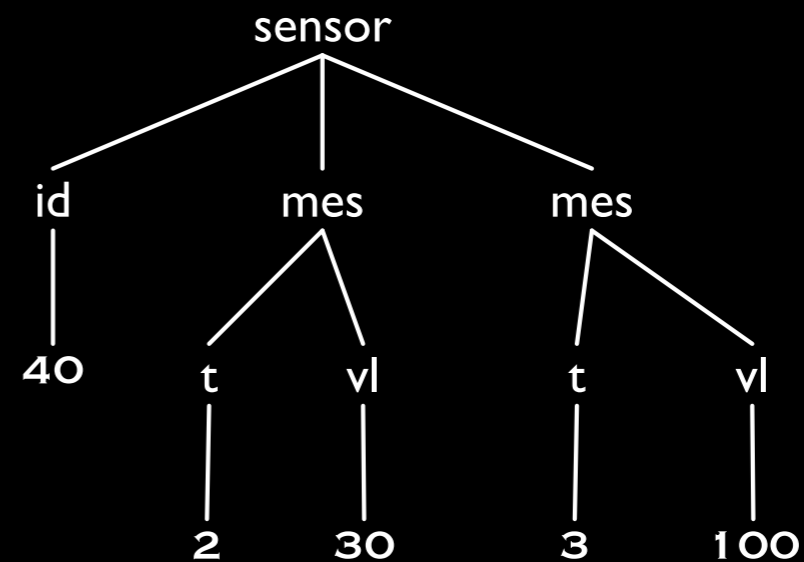
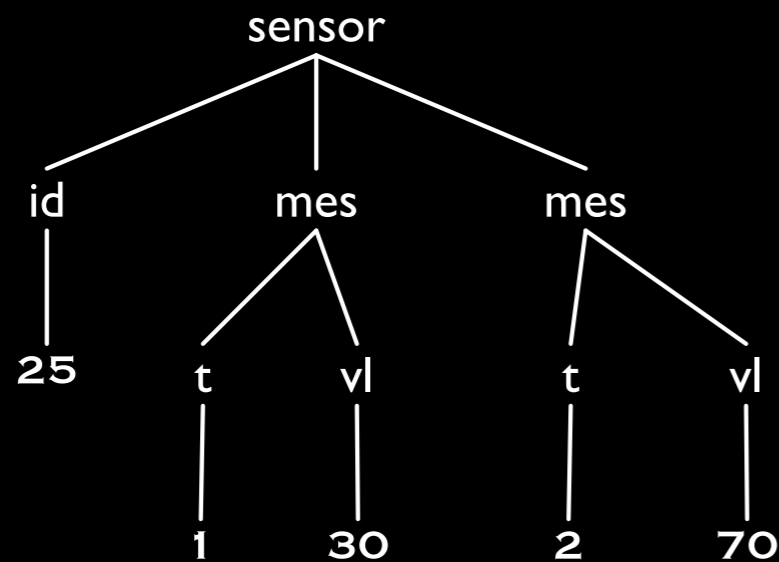
**How to measure infinite sets of trees?**

# Measuring Infinite Sets of Trees

I. Take a set  $S$  of trees with

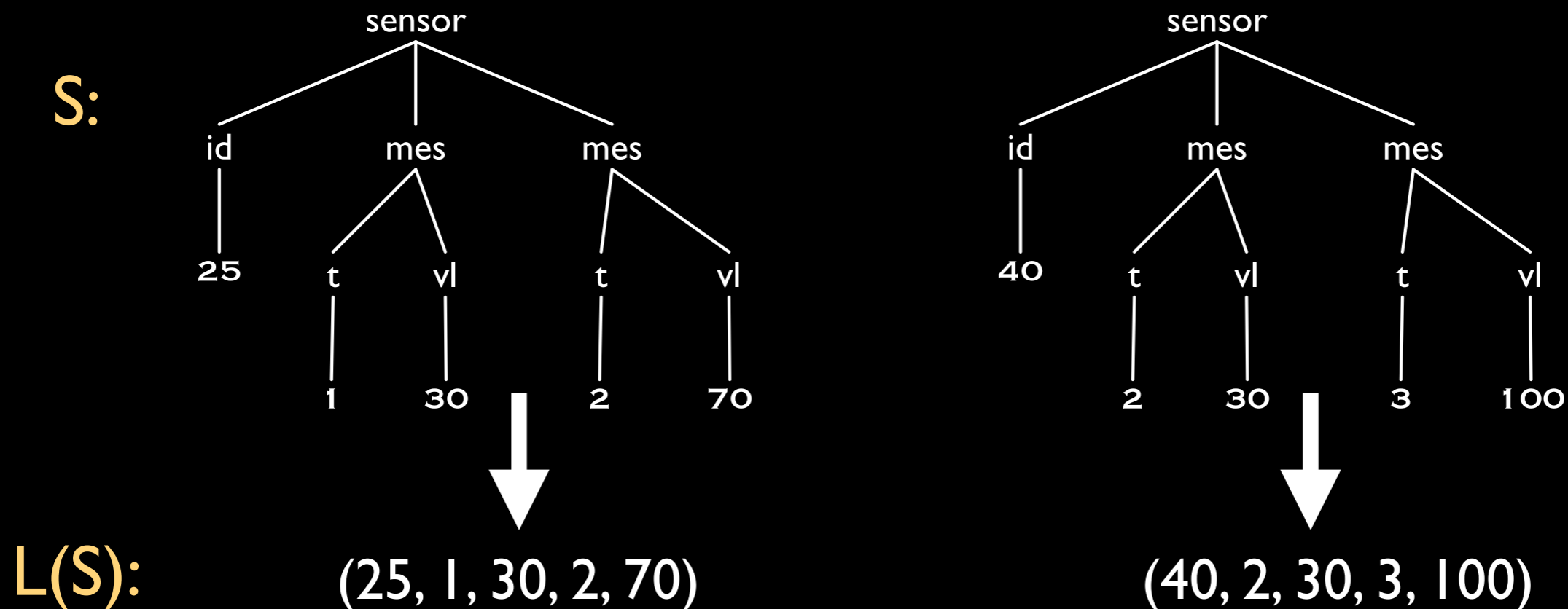
- **real values** on the leaves / **share** the same **structure**

**S:**



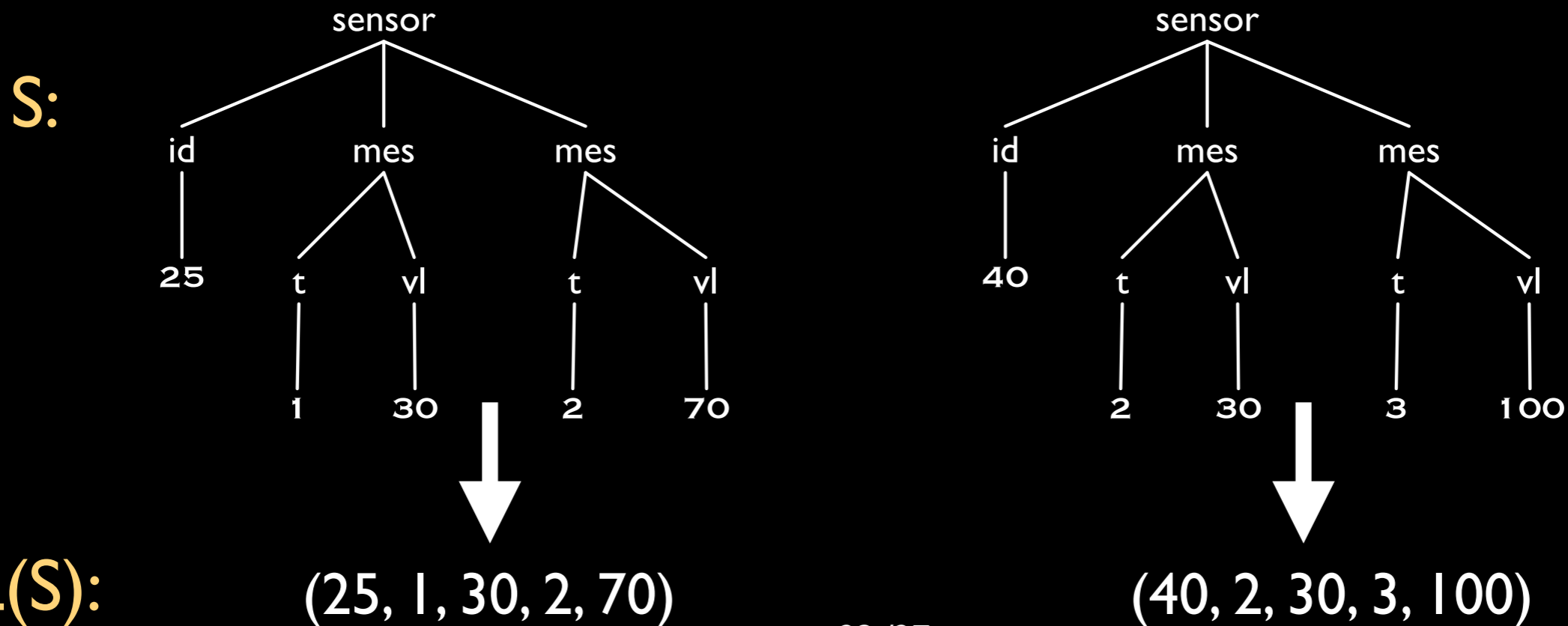
# Measuring Infinite Sets of Trees

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  $\mathbb{R}^n$



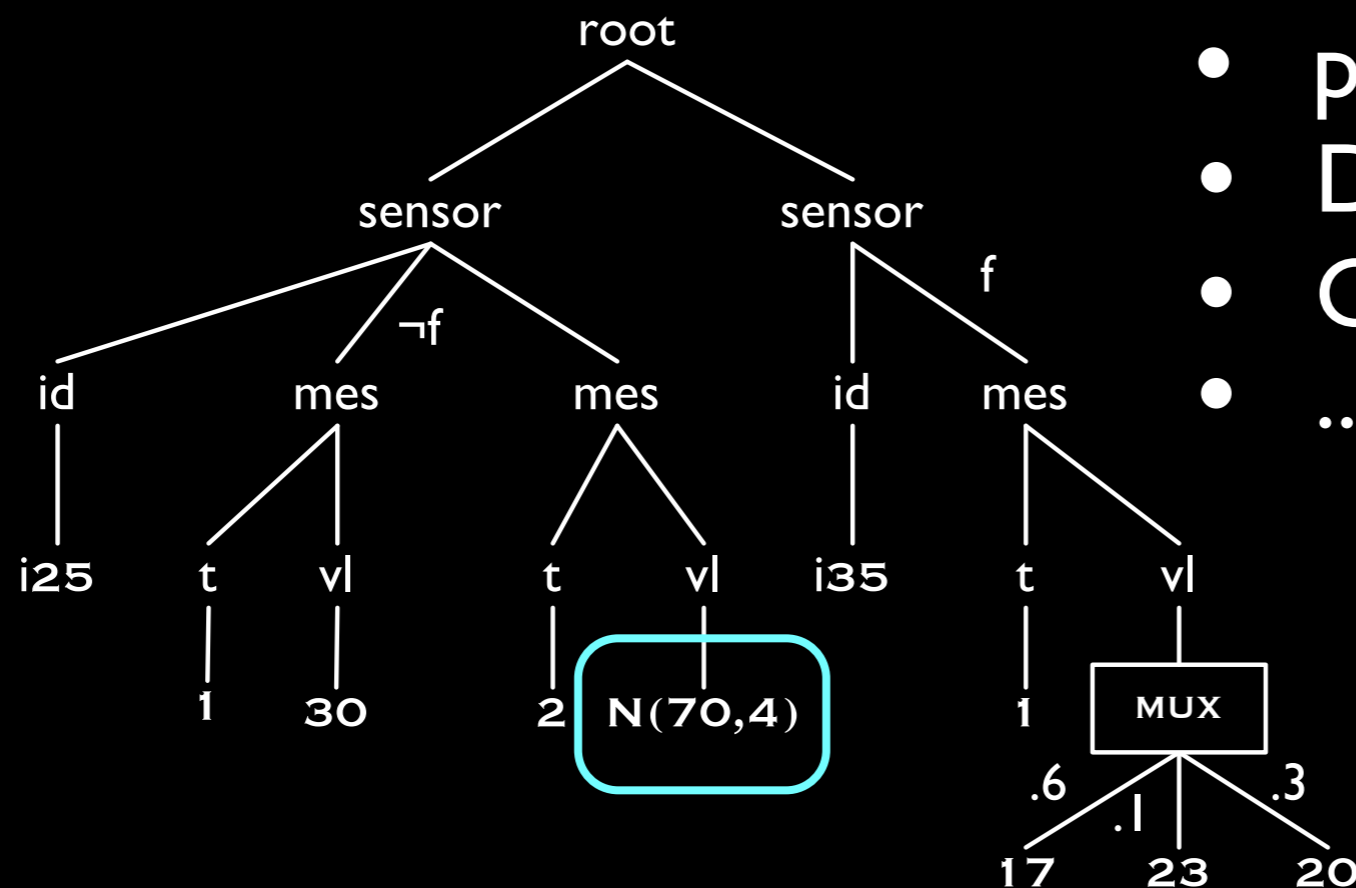
# Measuring Infinite Sets of Trees

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



# 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

# Webdam

Webdam Project:  
Foundations of Data Management  
<http://webdam.inria.fr>



DataRing Project: P2P Data Sharing  
for Online Communities  
[http://www.lina.univ-nantes.fr/  
projets/DataRing/](http://www.lina.univ-nantes.fr/projets/DataRing/)

- 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