

Probabilistic Databases: Models and Applications to Web Data

Pierre Senellart



Journées Télécom–UPS, 29 May 2015

Part I: Uncertainty in the Real World

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, etc.)
- ▶ Imperfect human judgment
- ▶ Lies, opinions, rumors

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, etc.)
- ▶ Imperfect human judgment
- ▶ Lies, opinions, rumors

Use case: Web information extraction

instance	iteration	date learned	confidence
<u>arabic, egypt</u>	406	08-sep-2011	(Seed) 100.0
<u>chinese, republic of china</u>	439	24-oct-2011	100.0
<u>chinese, singapore</u>	421	21-sep-2011	(Seed) 100.0
<u>english, britain</u>	439	24-oct-2011	100.0
<u>english, canada</u>	439	24-oct-2011	(Seed) 100.0
<u>english, england001</u>	439	24-oct-2011	100.0
<u>arabic, morocco</u>	422	23-sep-2011	100.0
<u>cantonese, hong kong</u>	406	08-sep-2011	100.0
<u>english, uk</u>	436	19-oct-2011	100.0
<u>english, south vietnam</u>	427	27-sep-2011	99.9
<u>french, morocco</u>	422	23-sep-2011	99.9
<u>greek, turkey</u>	430	07-oct-2011	99.9

Never-ending Language Learning (NELL, CMU),

<http://rtw.ml.cmu.edu/rtw/kbbrowser/>

Use case: Web information extraction

Google squared labs

comedy movies

Square it Add

Item Name	Language	Director	Release Date
<input type="checkbox"/> The Mask	English	Chuck Russell	29 July 1994
<input type="checkbox"/> Scary M	<input checked="" type="radio"/> English language for the mask www.infibeam.com - all 9 sources »	<input checked="" type="radio"/> Chuck Russell directed by for The Mask www.infibeam.com - all 9 sources »	
<input type="checkbox"/> Superba	Other possible values <input type="radio"/> English Language Low confidence language for Mask www.freebase.com	Other possible values <input type="radio"/> John R. Dilworth Low confidence director for The Mask www.freebase.com	
<input type="checkbox"/> Music	<input type="radio"/> english, french Low confidence languages for the mask www.dvdreview.com	<input type="radio"/> Fiorella Infascelli Low confidence directed by for The Mask www.freebase.com - all 2 sources »	
<input type="checkbox"/> Knocked	<input type="radio"/> Italian Language Low confidence language for The Mask www.freebase.com	<input type="radio"/> Charles Russell Low confidence directed by for The Mask www.freebase.com - all 2 sources »	

[Search for more values »](#)

Google Squared (terminated), screenshot from [Fink et al., 2011]

Use case: Web information extraction

Subject	Predicate	Object	Confidence
Elvis Presley	diedOnDate	1977-08-16	97.91%
Elvis Presley	isMarriedTo	Priscilla Presley	97.29%
Elvis Presley	influences	Carlo Wolff	96.25%

YAGO, <http://www.mpi-inf.mpg.de/yago-naga/yago>

Uncertainty in Web information extraction

- ▶ The information extraction system is **imprecise**
- ▶ The system has some **confidence** in the information extracted, which can be:
 - ▶ a **probability** of the information being true (e.g., conditional random fields)
 - ▶ an **ad-hoc** numeric confidence score
 - ▶ a **discrete** level of confidence (low, medium, high)
- ▶ What if this uncertain information is not seen as something final, but is used as a source of, e.g., a query answering system?

Different types of uncertainty

Two dimensions:

- ▶ Different types:
 - ▶ **Unknown** value: NULL in an RDBMS
 - ▶ **Alternative** between several possibilities: either A or B or C
 - ▶ **Imprecision on a numeric value**: a sensor gives a value that is an approximation of the actual value
 - ▶ **Confidence in a fact as a whole**: cf. information extraction
 - ▶ **Structural uncertainty**: the schema of the data itself is uncertain
- ▶ **Qualitative** (NULL) or **Quantitative** (95%, low-confidence, etc.) uncertainty

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.

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 **all different forms** of uncertainty
- ▶ Use **probabilities** to represent quantitative information on the confidence in the data
- ▶ Query data and retrieve **uncertain** results
- ▶ Allow adding, deleting, modifying data in an **uncertain** way
- ▶ Bonus (if possible): Keep as well **lineage/provenance** information, so as to ensure **traceability**

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)
- ▶ In other cases, we “cheat” and pretend that (normalized) **confidence scores** are probabilities: see this as a first-order approximation

Objective of this talk

- ▶ Present **data models** for uncertain data management in general, and probabilistic data management in particular:
 - ▶ relational
 - ▶ XML
- ▶ Briefly discuss **querying** of probabilistic data

Part II: Probabilistic Models of Uncertainty

- ▶ Probabilistic Relational Models
- ▶ Probabilistic XML

Possible worlds semantics

Possible world: A **regular** (deterministic) relational or XML database

Incomplete database: (Compact) representation of a **set of possible worlds**

Probabilistic database: (Compact) representation of a **probability distribution over possible worlds**, either:

- finite:** a set of possible worlds, each with their probability

- continuous:** more complicated, requires defining a σ -algebra, and a measure for the sets of this σ -algebra

Part II: Probabilistic Models of Uncertainty

- ▶ Probabilistic Relational Models
- ▶ Probabilistic XML

The relational model

- ▶ Data stored into **tables**
- ▶ Every table has a precise **schema** (**type** of columns)
- ▶ Adapted when the information is very **structured**

Patient	Examin. 1	Examin. 2	Diagnosis
A	23	12	α
B	10	23	β
C	2	4	γ
D	15	15	α
E	15	17	β

Codd tables, a.k.a. SQL NULLs

Patient	Examin. 1	Examin. 2	Diagnosis
A	23	12	α
B	10	23	\perp_1
C	2	4	γ
D	15	15	\perp_2
E	\perp_3	17	β

- ▶ Most **simple** form of incomplete database
- ▶ **Widely used** in practice, in DBMS since the mid-1970s!
- ▶ All NULLs (\perp) are considered **distinct**
- ▶ Possible world semantics: all (infinitely many under the **open world** assumption) possible completions of the table
- ▶ In SQL, **three-valued logic**, weird semantics:

```
SELECT * FROM Tel WHERE tel_nr = '333' OR tel_nr <> '333'
```

C-tables [Imielinski and Lipski, 1984]

Patient	Examin. 1	Examin. 2	Diagnosis	Condition
A	23	12	α	
B	10	23	\perp_1	
C	2	4	γ	
D	\perp_2	15	\perp_1	
E	\perp_3	17	β	$18 < \perp_3 < \perp_2$

- ▶ NULLs are labeled, and can be **reused** inside and across tuples
- ▶ **Arbitrary correlations** across tuples
- ▶ **Closed** under the relational algebra (Codd tables only closed under projection and union)
- ▶ Every set of possible worlds can be represented as a database with c-tables

Tuple-independent databases (TIDs)

[Lakshmanan et al., 1997, Dalvi and Suciu, 2007]

Patient	Examin. 1	Examin. 2	Diagnosis	Probability
A	23	12	α	0.9
B	10	23	β	0.8
C	2	4	γ	0.2
C	2	14	γ	0.4
D	15	15	α	0.6
D	15	15	β	0.4
E	15	17	β	0.7
E	15	17	α	0.3

- ▶ Allow representation of the **confidence** in each row of the table
- ▶ Impossible to express **dependencies** across rows
- ▶ Very simple model, well understood

Block-independent databases (BIDs)

[Barbará et al., 1992, Ré and Suciu, 2007]

Patient	Examin. 1	Examin. 2	Diagnosis	Probability
A	23	12	α	0.9
B	10	23	β	0.8
C	2	4	γ	0.2
C	2	14	γ	0.4
D	15	15	β	0.6
D	15	15	α	0.4
E	15	17	β	0.7
E	15	17	α	0.3

- ▶ The table has a **primary key**: tuples sharing a primary key are mutually exclusive (probabilities must sum up to ≤ 1)
- ▶ Simple **dependencies** (exclusion) can be expressed, but not more complex ones

Probabilistic c-tables [Green and Tannen, 2006]

Patient	Examin. 1	Examin. 2	Diagnosis	Condition
A	23	12	α	w_1
B	10	23	β	w_2
C	2	4	γ	w_3
C	2	14	γ	$\neg w_3 \wedge w_4$
D	15	15	β	w_5
D	15	15	α	$\neg w_5 \wedge w_6$
E	15	17	β	w_7
E	15	17	α	$\neg w_7$

- ▶ The w_i 's are **Boolean random variables**
- ▶ Each w_i has a probability of being true (e.g., $\Pr(w_1) = 0.9$)
- ▶ The w_i 's are independent
- ▶ Any **finite** probability distribution of tables can be represented using probabilistic c-tables

Two actual PRDBMS: Trio and MayBMS

Two main probabilistic relational DBMS:

Trio [Widom, 2005] Various **uncertainty operators**: unknown value, uncertain tuple, choice between different possible values, with probabilistic annotations. See example later on.

MayBMS [Koch, 2009] Implementation of the **probabilistic c-tables** model. In addition, uncertain tables can be constructed using a REPAIR-KEY operator, similar to BIDs.

Two actual PRDBMS: Trio and MayBMS

Two m

```
test=# select * from R;
dummy | weather | ground | p
-----+-----+-----+-----
dummy | rain    | wet    | 0.35
dummy | rain    | dry    | 0.05
dummy | no rain | wet    | 0.1
dummy | no rain | dry    | 0.5
(4 rows)
```

M

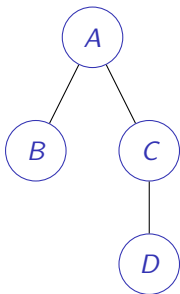
```
test=# create table S as
repair key Dummy in R weight by P;
SELECT
test=# select Ground, conf() from S group by Ground;
ground | conf
-----+-----
dry    | 0.55
wet    | 0.45
(2 rows)
```

own
ible
ter on.
bles
d using

Part II: Probabilistic Models of Uncertainty

- ▶ Probabilistic Relational Models
- ▶ Probabilistic XML

The semistructured model and XML



```
<a>
  <b>...</b>
  <c>
    <d>...</d>
  </c>
</a>
```

- ▶ **Tree-like** structuring of data
- ▶ **No** (or less) schema **constraints**
- ▶ Allow mixing **tags** (structured data) and text (unstructured content)
- ▶ Particularly adapted to **tagged** or **heterogeneous** content

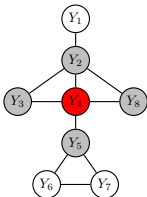
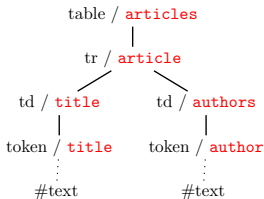
Why Probabilistic XML?

- ▶ Extensive literature about probabilistic relational databases [Dalvi et al., 2009, Widom, 2005, Koch, 2009]
- ▶ Different typical querying languages: 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]

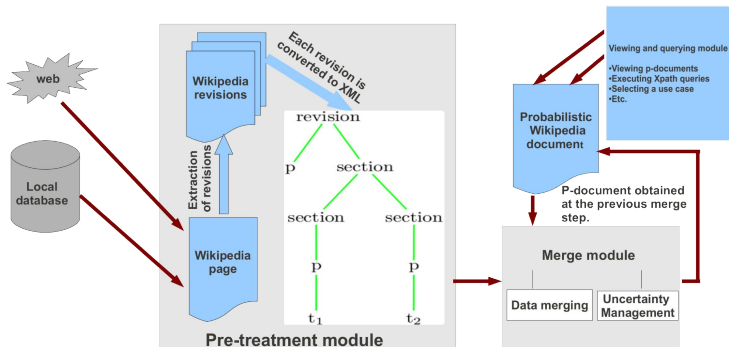
Web information extraction [Senellart et al., 2008]



- ▶ Annotate HTML Web pages with possible **labels**
- ▶ Labels can be learned from a **corpus of annotated documents**
- ▶ **Conditional random fields for XML:** estimate **probabilities of annotations** given annotations of neighboring nodes
- ▶ Provides **probabilistic labeling** of Web pages

Uncertain version control

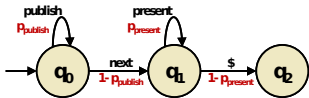
[Abdessalem et al., 2011, Ba et al., 2013]



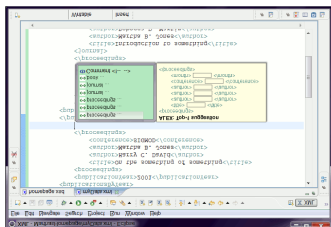
Use trees with probabilistic annotations to represent the **uncertainty in the correctness** of a document under open version control (e.g., Wikipedia articles)

Probabilistic summaries of XML corpora

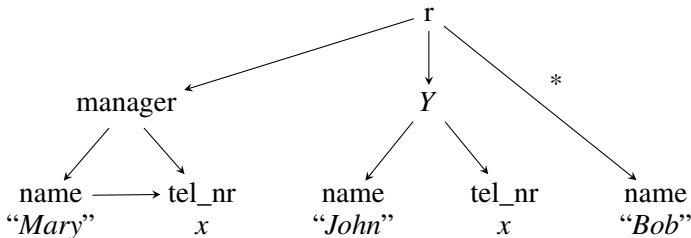
[Abiteboul et al., 2012a,b]



- ▶ Transform an XML schema (deterministic top-down tree automaton) into a **probabilistic generator** (probabilistic tree automaton) of XML documents
- ▶ Probability distribution **optimal** with respect to a given corpus
- ▶ **Application:** Optimal **auto-completions** in an XML editor

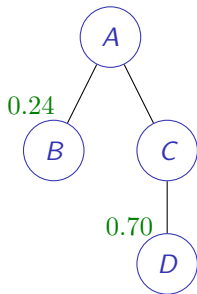


Incomplete XML [Barceló et al., 2009]



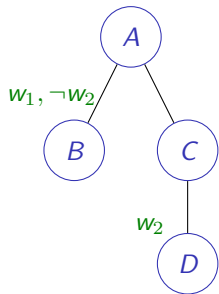
- ▶ Models all XML documents where these patterns exist (i.e., this subtree can be matched)
- ▶ Can be used for query answering, etc.

Simple probabilistic annotations



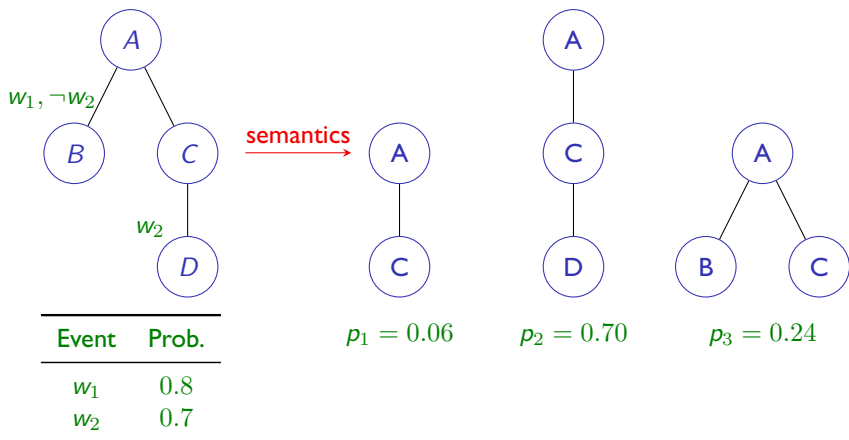
- ▶ **Probabilities** associated to tree nodes
- ▶ Express parent/child dependencies
- ▶ Impossible to express more complex dependencies
- ▶ \Rightarrow some **sets of possible worlds** are not expressible this way!

Annotations with event variables



Event	Prob.
w_1	0.8
w_2	0.7

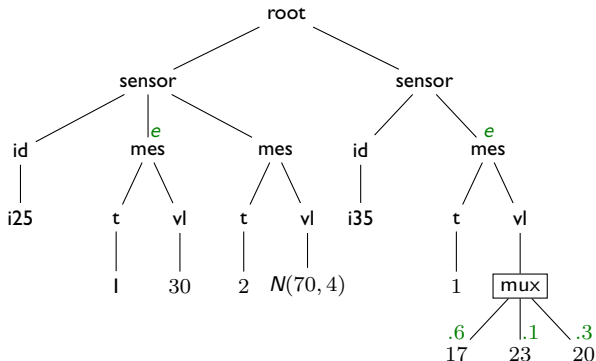
Annotations with event variables



- ▶ Expresses **arbitrarily complex** dependencies
- ▶ Obviously, analogous to probabilistic c-tables

A general probabilistic XML model

[Abiteboul et al., 2009]



- ▶ e : event “it did not rain” at time l
- ▶ mux: mutually exclusive options
- ▶ $N(70, 4)$: normal distribution

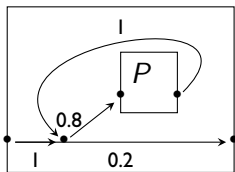
- ▶ Compact representation of a **set of possible worlds**
- ▶ Two kinds of dependencies: global (e) and local (mux)
- ▶ Generalizes **all previously proposed models** of the literature

Recursive Markov chains [Benedikt et al., 2010]

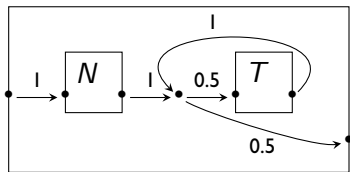
<!ELEMENT directory (person*)>

<!ELEMENT person (name,phone*)>

D: directory



P: person



- ▶ Probabilistic model that **extends** PXML with local dependencies
- ▶ Allows generating documents of **unbounded** width or depth

Part III: Querying Probabilistic Databases

- ▶ Semantics
- ▶ Lineage computation and #P-Hardness
- ▶ Special tractable case within Probabilistic XML

Part III: Querying Probabilistic Databases

- ▶ **Semantics**
- ▶ Lineage computation and #P-Hardness
- ▶ Special tractable case within Probabilistic XML

Semantics Of Query Answering: Example

Person

name	city	probability
Ivan	Moscow	0.3
Jean	Paris	0.8
Pedro	Madrid	0.4

Query:

```
SELECT name FROM Person
```

Semantics Of Query Answering: Example

Person

name	city	probability
Ivan	Moscow	0.3
Jean	Paris	0.8
Pedro	Madrid	0.4

Query:
SELECT name FROM Person

$$\text{Pr} = 0.3 * 0.8 * 0.4$$



name	city
Ivan	Moscow
Jean	Paris
Pedro	Madrid

$$\text{Pr} = 0.3 * 0.2 * 0.4$$

name	city
Ivan	Moscow
Pedro	Madrid

...

Semantics Of Query Answering: Example

Person

name	city	probability
Ivan	Moscow	0.3
Jean	Paris	0.8
Pedro	Madrid	0.4

Query:

SELECT name FROM Person

$$\text{Pr} = 0.3 * 0.8 * 0.4$$

name	city
Ivan	Moscow
Jean	Paris
Pedro	Madrid

$$\text{Pr} = 0.3 * 0.2 * 0.4$$

name	city
Ivan	Moscow
Pedro	Madrid

...

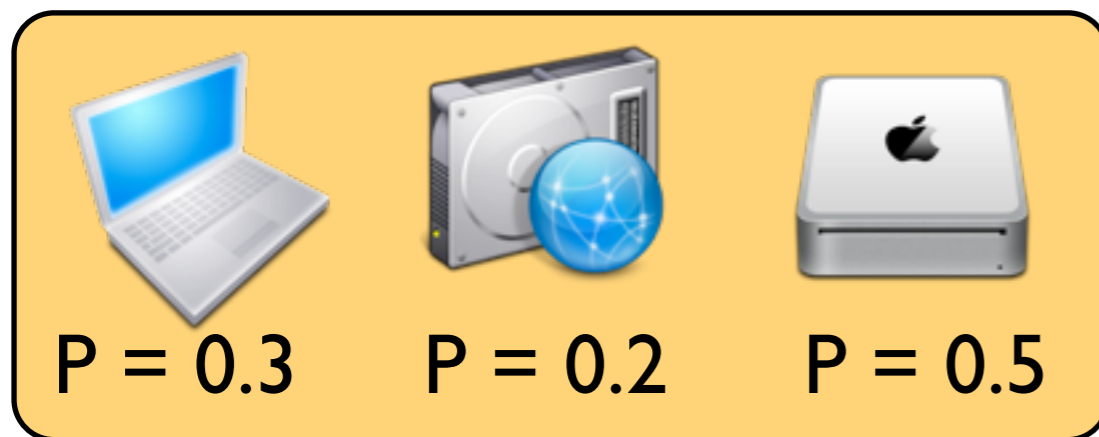
Possible answers: $(\{\text{Ivan, Juan, Pedro}\}, 0.3 * 0.8 * 0.4),$
 $(\{\text{Ivan, Pedro}\}, 0.3 * 0.2 * 0.4), \dots$

Possible tuples: $(\text{Ivan}, 0.3), (\text{Jean}, 0.8), (\text{Pedro}, 0.4)$

Semantics Of Query Answering

Possible Answers Semantics

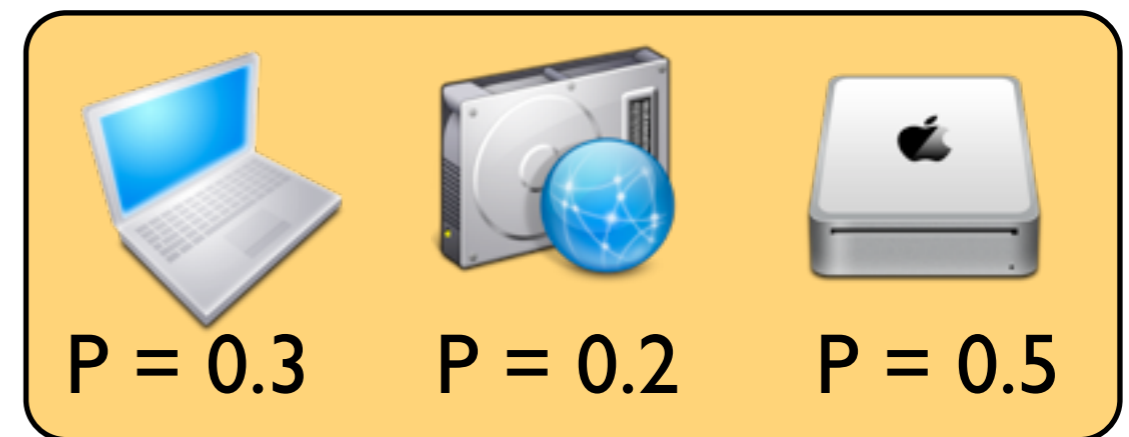
Probabilistic DB:



A yellow rounded rectangle containing three icons: a laptop, a server tower with a blue globe, and a white Apple iMac. Below each icon is a probability value: $P = 0.3$, $P = 0.2$, and $P = 0.5$.

Possible Tuples Semantics

Probabilistic DB:

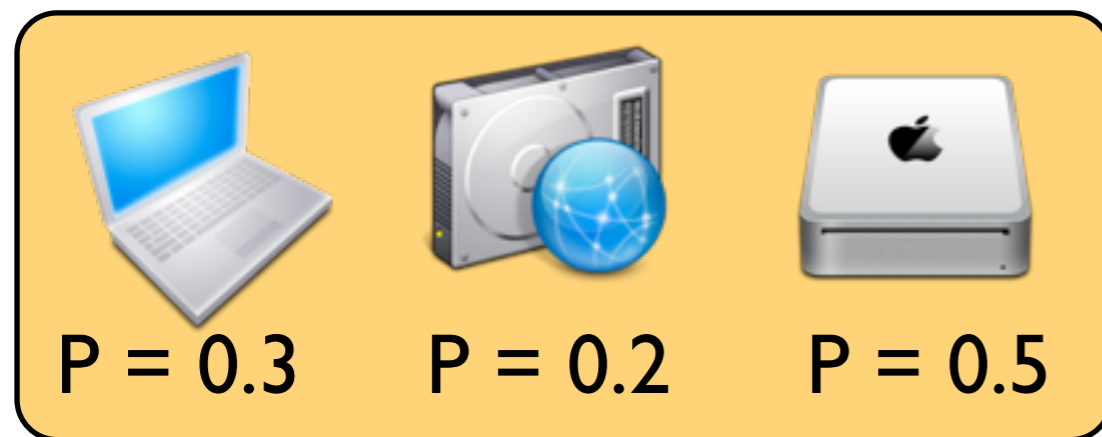


A yellow rounded rectangle containing three icons: a laptop, a server tower with a blue globe, and a white Apple iMac. Below each icon is a probability value: $P = 0.3$, $P = 0.2$, and $P = 0.5$.

Semantics Of Query Answering

Possible Answers Semantics

Probabilistic DB:



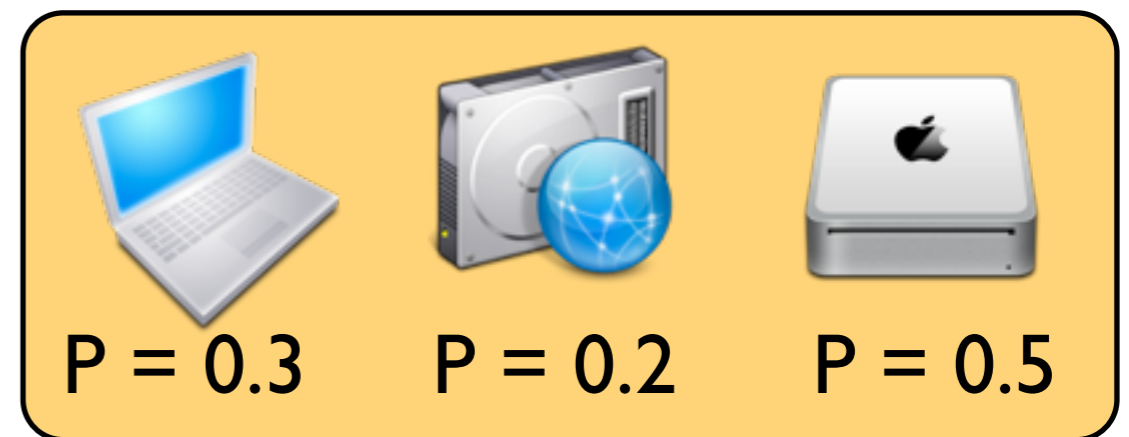
Q
↓
{a}

Q
↓

Q
↓
{a,b}

Possible Tuples Semantics

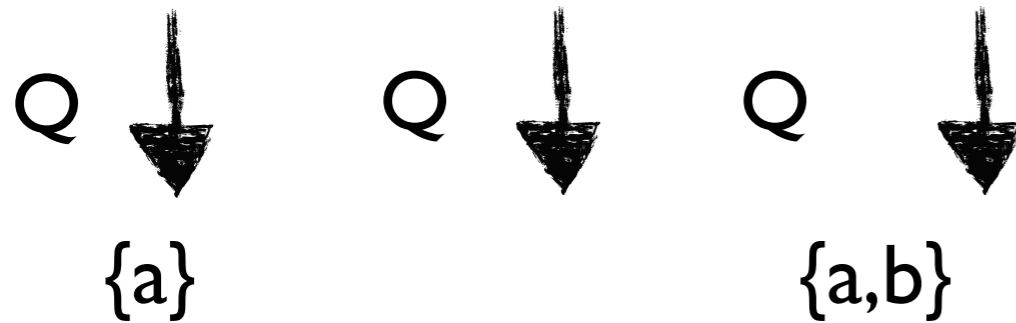
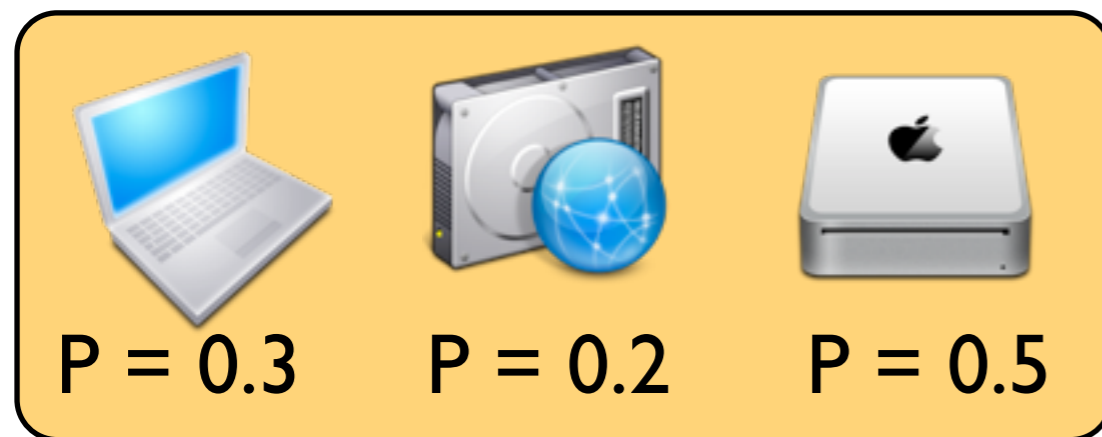
Probabilistic DB:



Semantics Of Query Answering

Possible Answers Semantics

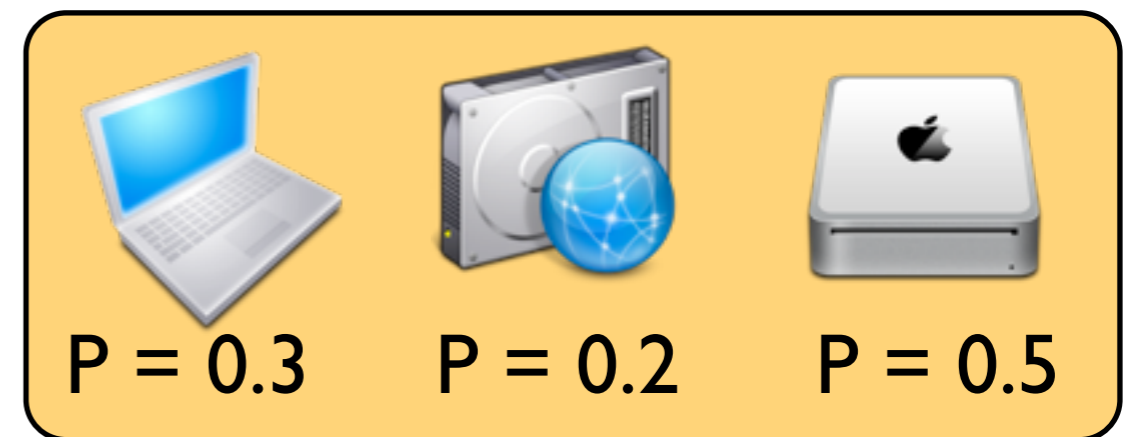
Probabilistic DB:



Answer: $(\{a\}, 0.3); (\{a,b\}, 0.5)$

Possible Tuples Semantics

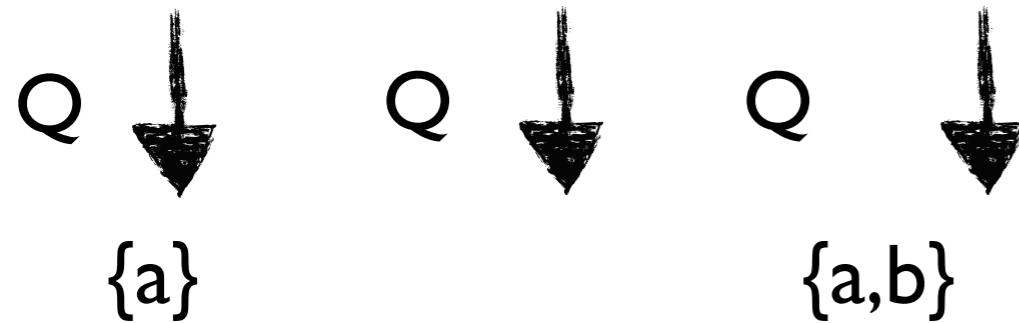
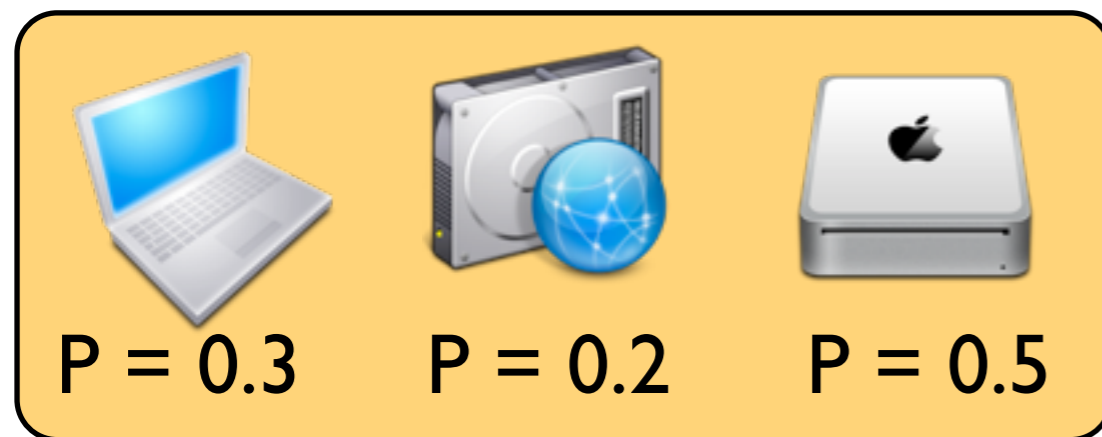
Probabilistic DB:



Semantics Of Query Answering

Possible Answers Semantics

Probabilistic DB:

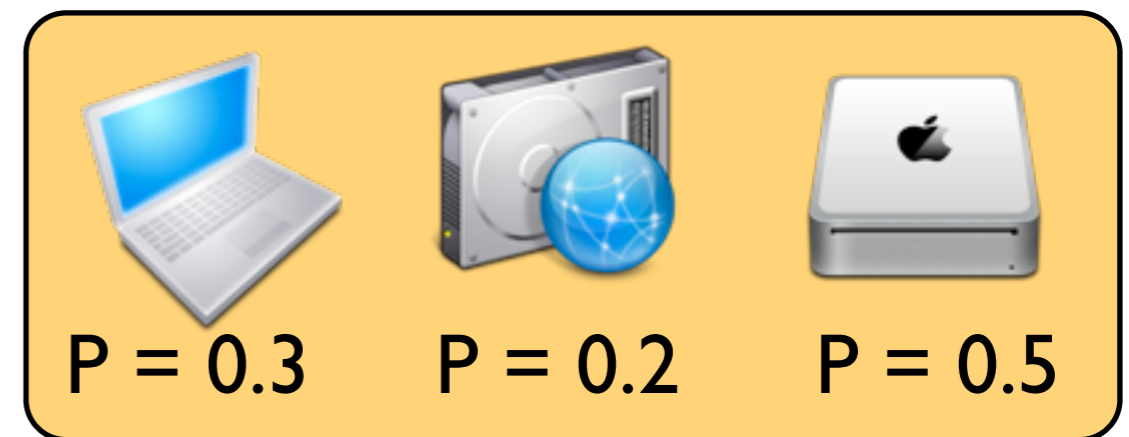


Answer: $(\{a\}, 0.3)$; $(\{a,b\}, 0.5)$

Probability distribution on
sets of tuples

Possible Tuples Semantics

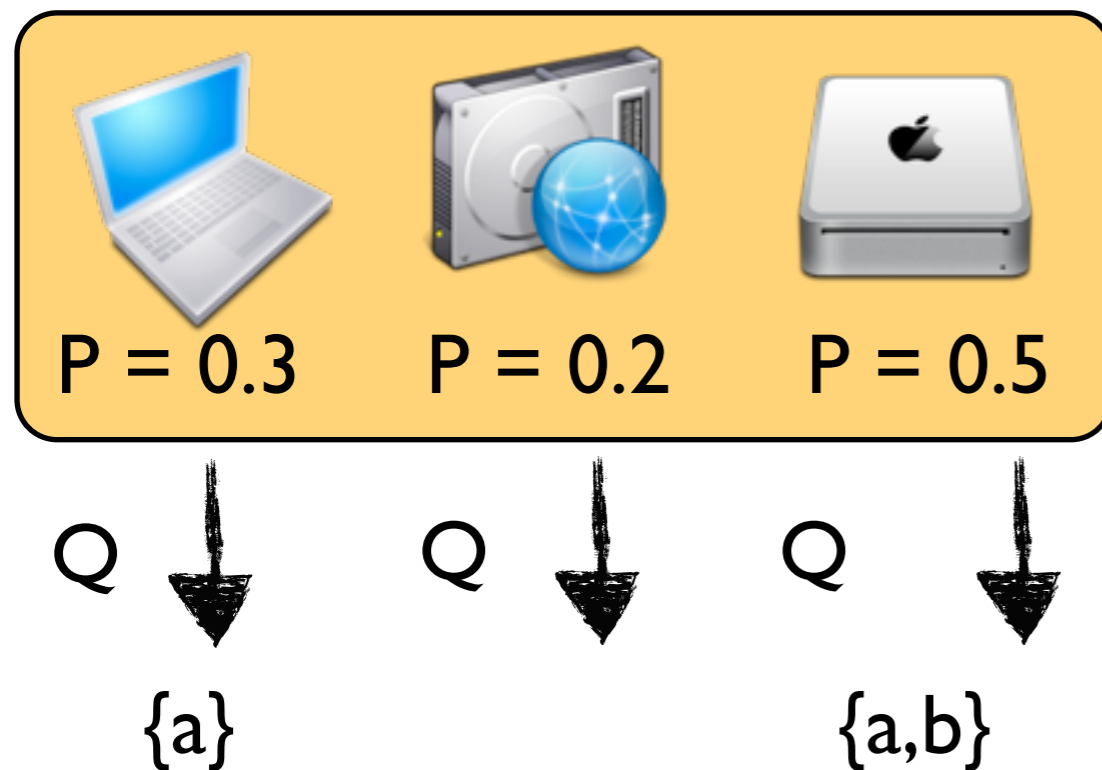
Probabilistic DB:



Semantics Of Query Answering

Possible Answers Semantics

Probabilistic DB:

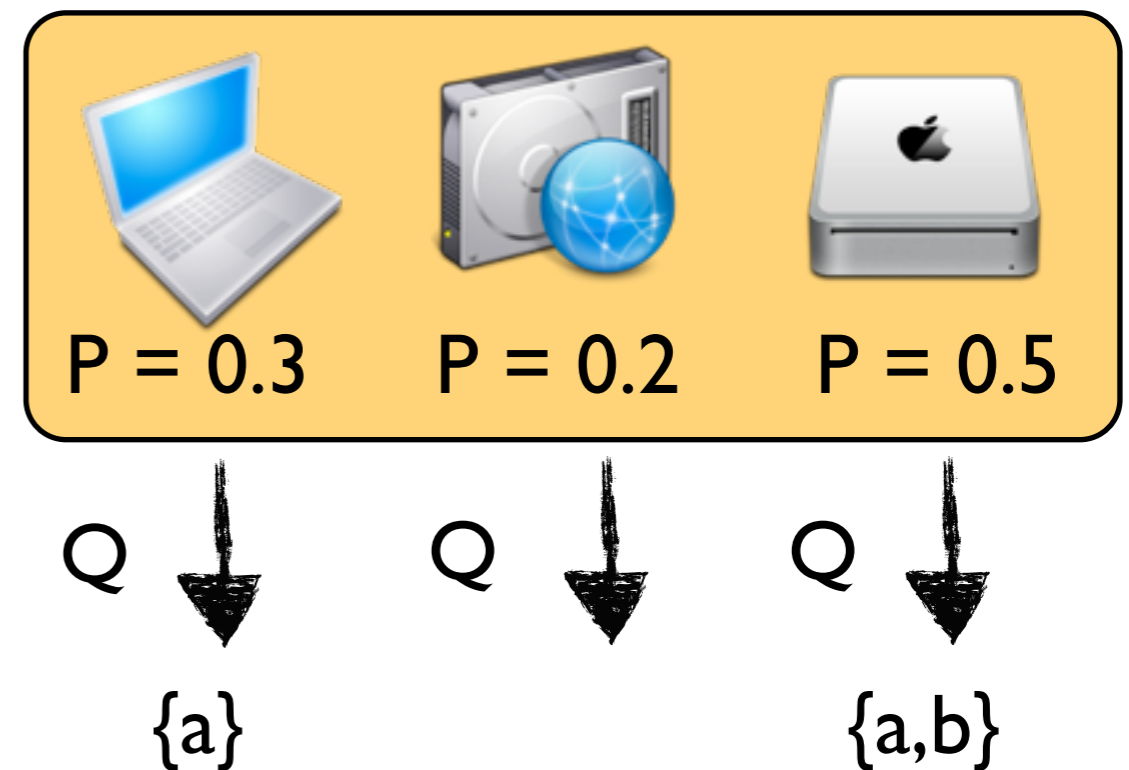


Answer: $(\{a\}, 0.3); (\{a,b\}, 0.5)$

Probability distribution on
sets of tuples

Possible Tuples Semantics

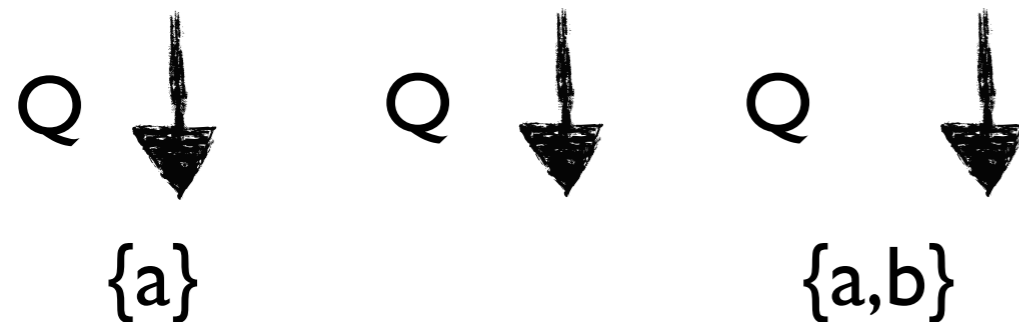
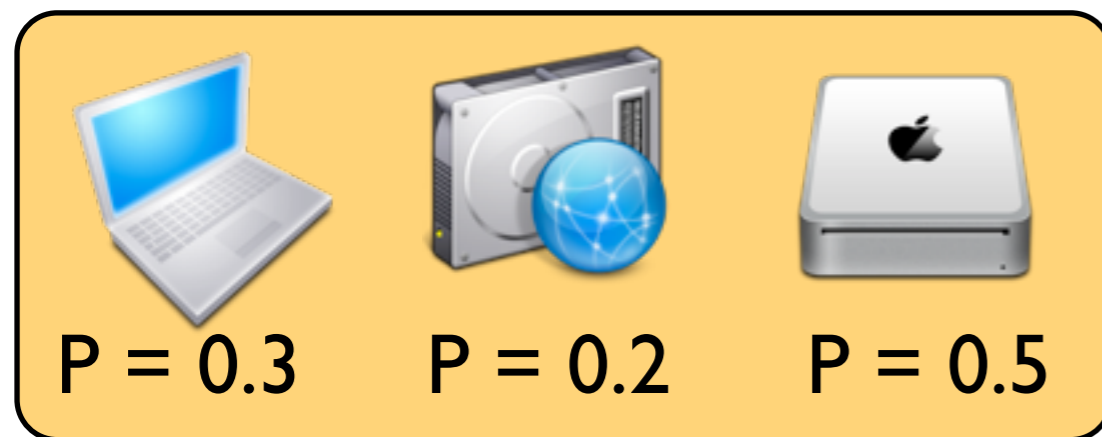
Probabilistic DB:



Semantics Of Query Answering

Possible Answers Semantics

Probabilistic DB:

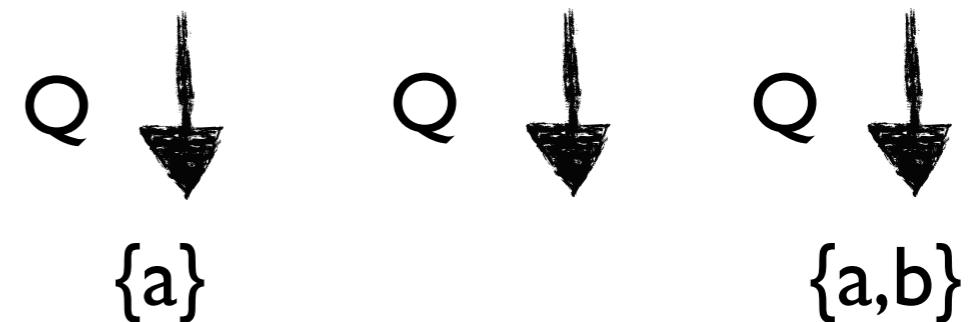
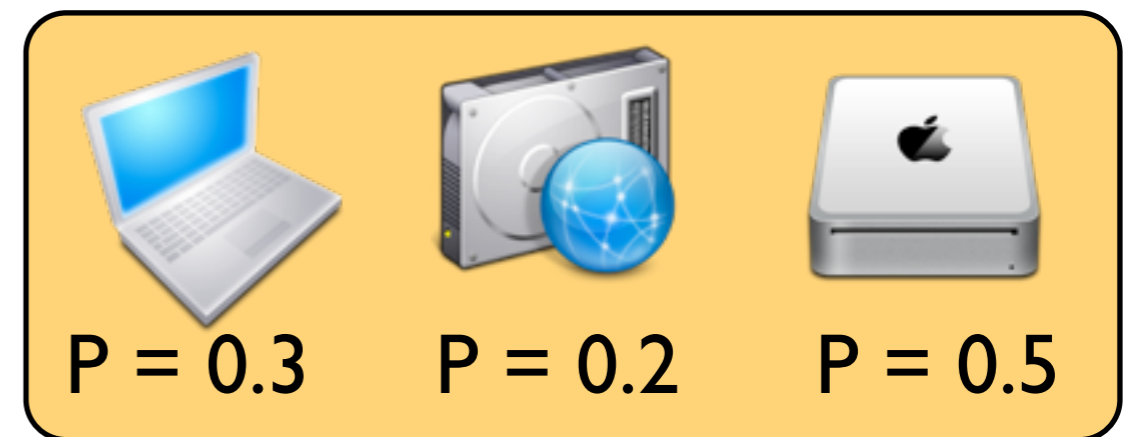


Answer: $(\{a\}, 0.3); (\{a,b\}, 0.5)$

Probability distribution on
sets of tuples

Possible Tuples Semantics

Probabilistic DB:

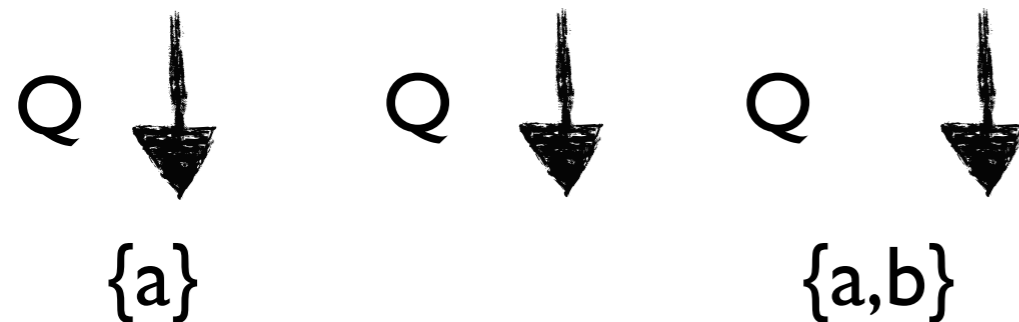
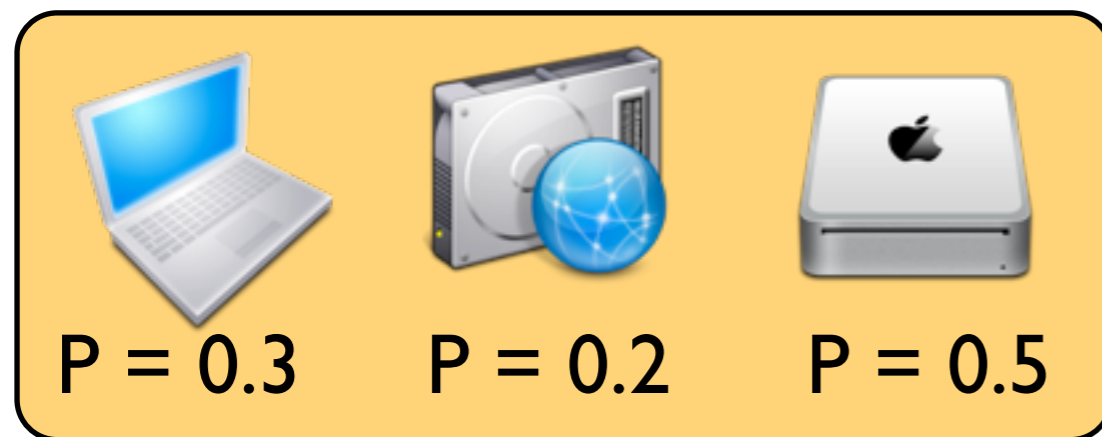


Answer: $(a, 0.8), (b, 0.5)$

Semantics Of Query Answering

Possible Answers Semantics

Probabilistic DB:

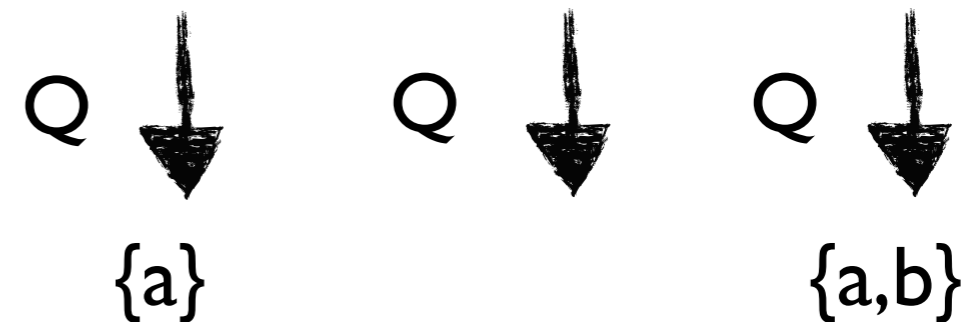
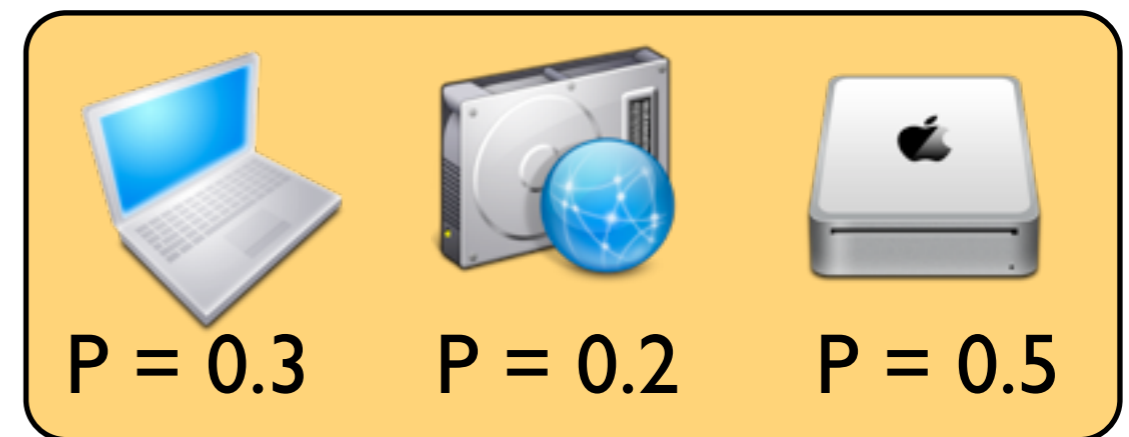


Answer: $(\{a\}, 0.3); (\{a,b\}, 0.5)$

Probability distribution on
sets of tuples

Possible Tuples Semantics

Probabilistic DB:



Answer: $(a, 0.8), (b, 0.5)$

Probability distribution on
tuples

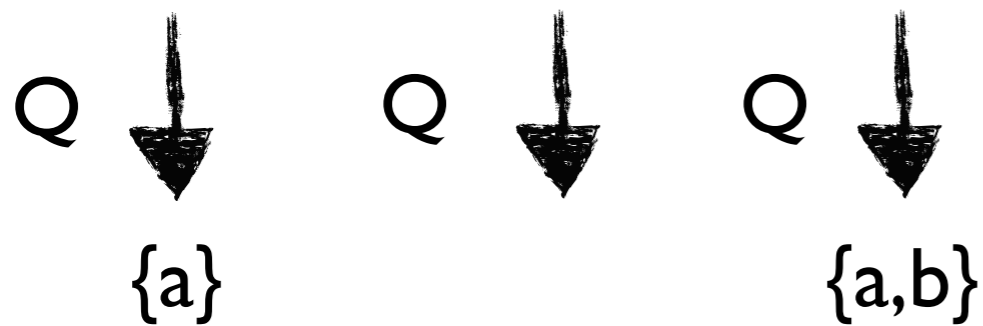
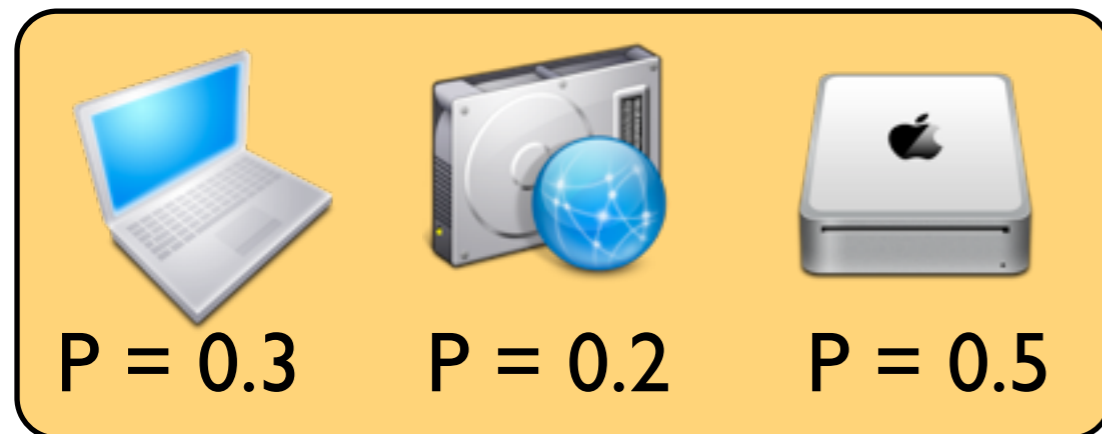
Possible Answer vs Possible Tuple Semantics

[Dalvi,Suciu'09]

- **Possible answers** semantics:
 - Precise
 - Can be used to compose queries
 - Difficult user interface
- **Possible tuples** semantics:
 - Less precise, but simple; sufficient for most apps
 - Cannot be used to compose queries
 - Simple user interface

Goals of Query Answering

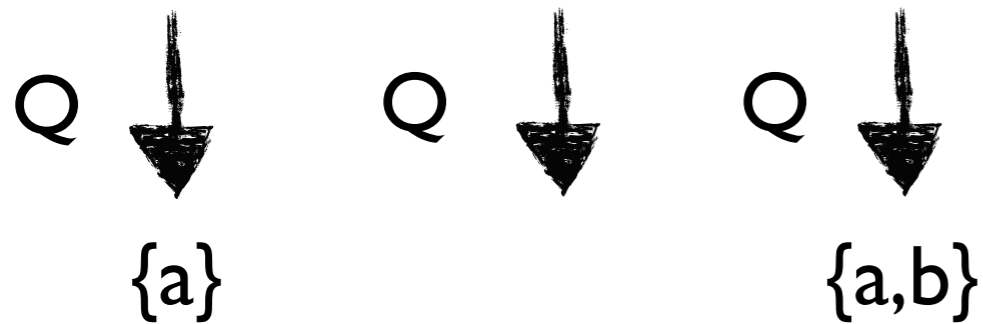
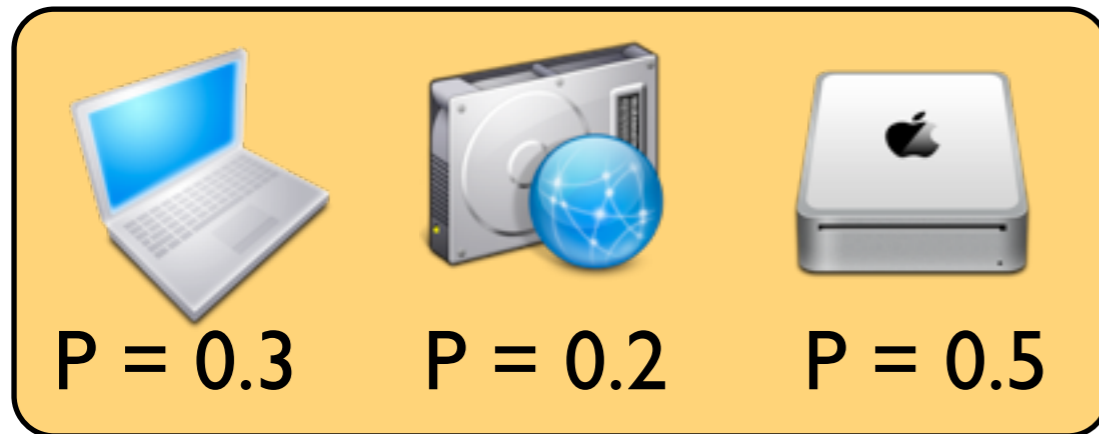
Probabilistic DB:



Answer: $(a, 0.8), (b, 0.5)$

Goals of Query Answering

Probabilistic DB:

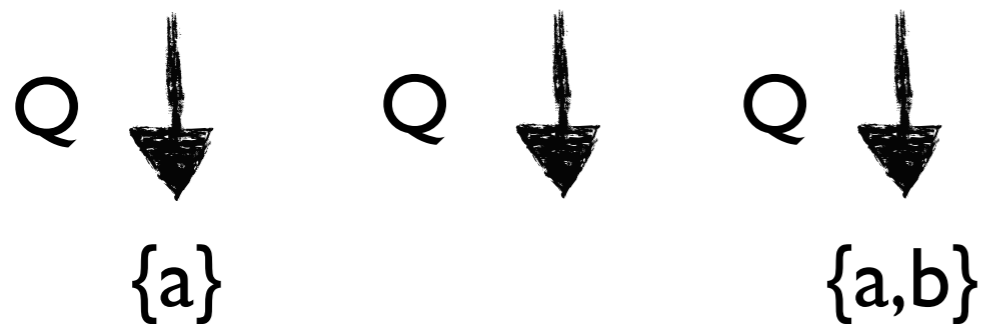
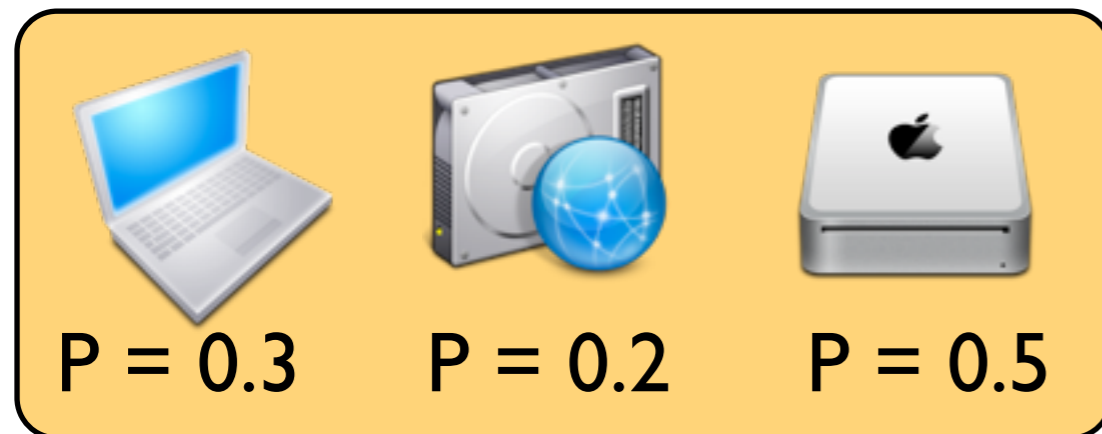


Answer: $(a, 0.8), (b, 0.5)$

- There may be EXP many worlds \rightarrow naive evaluation is exponential
- Can we do better?

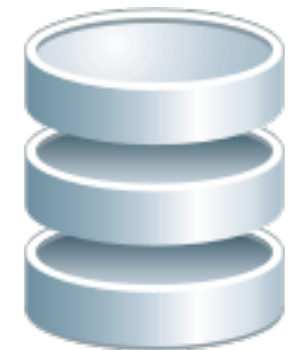
Goals of Query Answering

Probabilistic DB:



Answer: $(a, 0.8), (b, 0.5)$

Representation
of Prob DB:



semantics



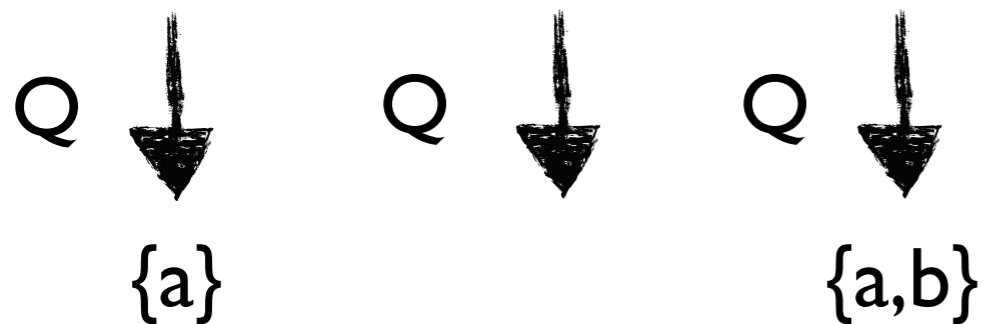
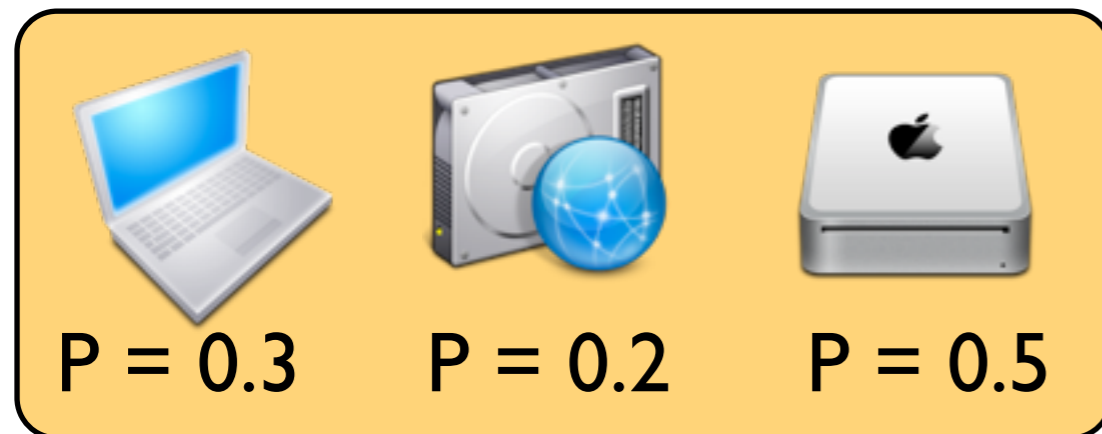
theory

practice

- There may be EXP many worlds \rightarrow naive evaluation is exponential
- Can we do better?

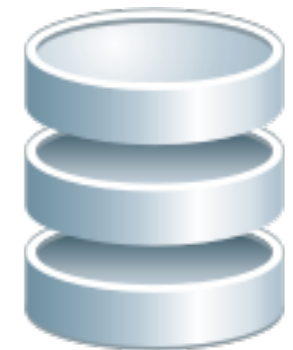
Goals of Query Answering

Probabilistic DB:



Answer: $(a, 0.8), (b, 0.5)$

Representation of Prob DB:



Q



$(a, 0.8), (b, 0.5)$

semantics



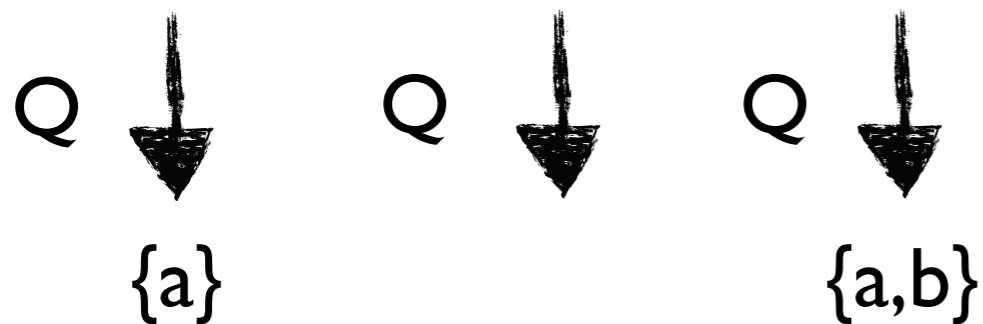
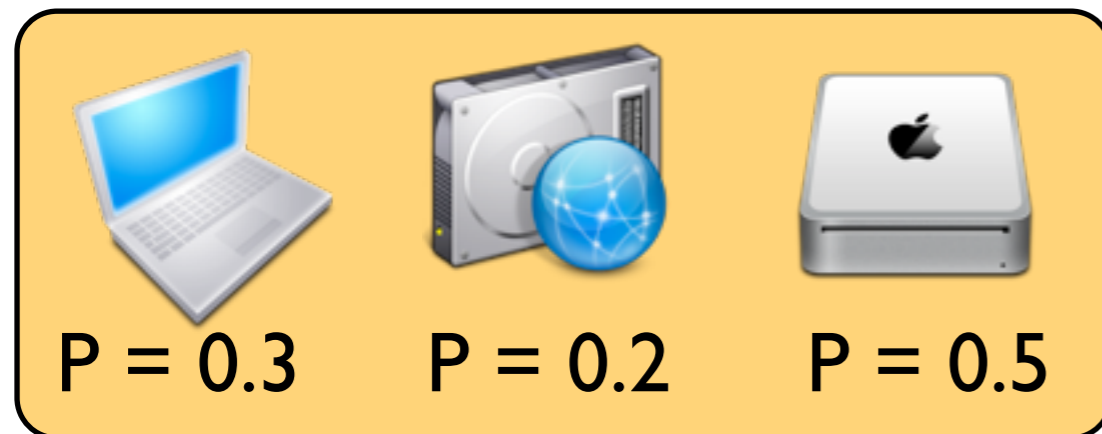
theory

practice

- There may be EXP many worlds \rightarrow naive evaluation is exponential
- Can we do better?

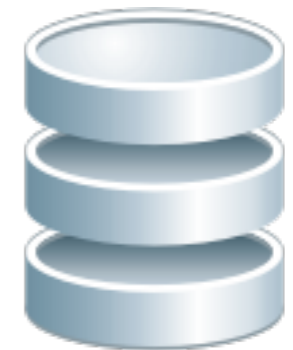
Goals of Query Answering

Probabilistic DB:



Answer: $(a, 0.8), (b, 0.5)$

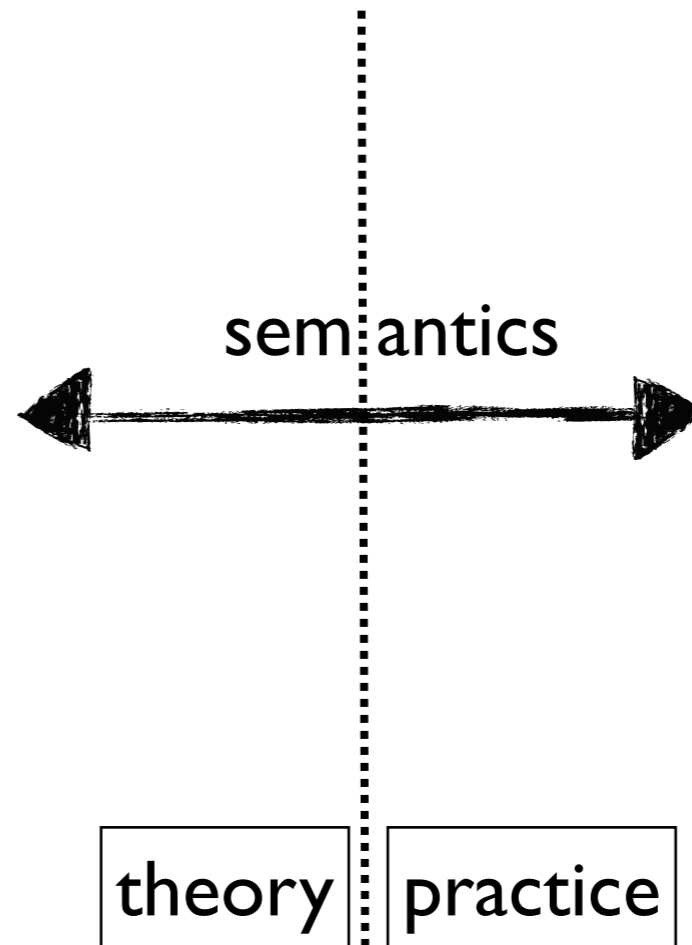
Representation of Prob DB:



Q



$(a, 0.8), (b, 0.5)$



- There may be EXP many worlds \rightarrow naive evaluation is exponential
- Can we do better?
- **Goal:** to find out how to query **representation system directly**

Part III: Querying Probabilistic Databases

- ▶ Semantics
- ▶ Lineage computation and #P-Hardness
- ▶ Special tractable case within Probabilistic XML

General Lineage: Examples of Operators (I)

Drivers

ID	person	car	Lineage
31	Jimmy	Toyota	$x \wedge y$
32	Jimmy	Honda	y
33	Hank	Honda	$x \vee z$

Saw

ID	witness	car	Lineage
21	Cathy	Honda	w

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Project = π_{person} (Drives)

Project

person	Lineage
Jimmy	$(x \wedge y) \vee y$
Hank	$x \vee z$

Select = $\sigma_{\text{car}=\text{"honda"}}$ (Drives)

Select

person	car	Lineage
Jimmy	Honda	y
Hank	Honda	$x \vee z$

General Lineage: Examples of Operators (I)

Drivers

ID	person	car	Lineage
31	Jimmy	Toyota	$x \wedge y$
32	Jimmy	Honda	y
33	Hank	Honda	$x \vee z$

Saw

ID	witness	car	Lineage
21	Cathy	Honda	w

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Project = π_{person} (Drives)

Project

person	Lineage
Jimmy	$(x \wedge y) \vee y$
Hank	$x \vee z$

Select = $\sigma_{\text{car}=\text{"honda"}}$ (Drives)

Select

person	car	Lineage
Jimmy	Honda	y
Hank	Honda	$x \vee z$

General Lineage: Examples of Operators (2)

Drivers

ID	person	car	Lineage
31	Jimmy	Toyota	$x \wedge y$
32	Jimmy	Honda	y
33	Hank	Honda	$x \vee z$

Saw

ID	witness	car	Lineage
21	Cathy	Honda	w

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Join = $\text{Saw} \bowtie_{\text{car}} \text{Drives}$

Several = $\pi_{\text{person}}(\sigma_{\text{person}=\text{"Hank"}}(\text{Saw} \bowtie_{\text{car}} \text{Drives}))$

Join

person	car	witness	Lineage
Jimmy	Honda	Cathy	$y \wedge w$
Hank	Honda	Cathy	$(x \vee z) \wedge w$

Several

person	Lineage
Hank	$(x \vee z) \wedge w$

General Lineage: Examples of Operators (2)

Drivers

ID	person	car	Lineage
31	Jimmy	Toyota	$x \wedge y$
32	Jimmy	Honda	y
33	Hank	Honda	$x \vee z$

Saw

ID	witness	car	Lineage
21	Cathy	Honda	w

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Join = $\text{Saw} \bowtie_{\text{car}} \text{Drives}$

Several = $\pi_{\text{person}}(\sigma_{\text{person}=\text{"Hank"}}(\text{Saw} \bowtie_{\text{car}} \text{Drives}))$

Join

person	car	witness	Lineage
Jimmy	Honda	Cathy	$y \wedge w$
Hank	Honda	Cathy	$(x \vee z) \wedge w$

Several

person	Lineage
Hank	$(x \vee z) \wedge w$

General Lineage: Examples of Operators (3)

Saw-day

ID	witness	car	Lineage
31	Cathy	Honda	z
32	Bob	BMW	$y \wedge w$

Saw-night

ID	witness	car	Lineage
21	Cathy	Honda	w

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Union = Saw-day \cup Saw-night

Difference = Saw-day \setminus Saw-night

Union

witness	car	Lineage
Cathy	Honda	$z \vee w$
Bob	BMW	$y \wedge w$

Difference

witness	car	Lineage
Cathy	Honda	$z \wedge (\neg w)$
Bob	BMW	$y \wedge w$

General Lineage: Examples of Operators (3)

Saw-day

ID	witness	car	Lineage
31	Cathy	Honda	z
32	Bob	BMW	$y \wedge w$

Saw-night

ID	witness	car	Lineage
21	Cathy	Honda	w

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Union = Saw-day \cup Saw-night

Difference = Saw-day \setminus Saw-night

Union

witness	car	Lineage
Cathy	Honda	$z \vee w$
Bob	BMW	$y \wedge w$

Difference

witness	car	Lineage
Cathy	Honda	$z \wedge (\neg w)$
Bob	BMW	$y \wedge w$

Query Probabilities from Lineage

Join = Saw \bowtie_{car} Drives

$\Pr(x \text{ is true}) = 0.2$ $\Pr(z \text{ is true}) = 0.8$
 $\Pr(y \text{ is true}) = 0.4$ $\Pr(w \text{ is true}) = 0.5$

Join

person	car	witness	Lineage
Jimmy	Honda	Cathy	$y \wedge w$
Hank	Honda	Cathy	$(x \vee z) \wedge w$

Theorem: SPJUD-query evaluation over PrRBDs with boolean-formulas lineage is **#P-hard**, i.e. intractable

Query Probabilities from Lineage

Join = Saw \bowtie_{car} Drives

$\Pr(x \text{ is true}) = 0.2$ $\Pr(z \text{ is true}) = 0.8$
 $\Pr(y \text{ is true}) = 0.4$ $\Pr(w \text{ is true}) = 0.5$

Join

person	car	witness	Lineage
Jimmy	Honda	Cathy	$y \wedge w$
Hank	Honda	Cathy	$(x \vee z) \wedge w$

- $\Pr(\text{Jimmy} \in (\text{Saw} \bowtie_{\text{car}} \text{Drives})) = \Pr(y \wedge w) = \Pr(y) \times \Pr(w) = 0.4 \times 0.5 = 0.2$

Theorem: SPJUD-query evaluation over PrRBDs with boolean-formulas lineage is **#P-hard**, i.e. intractable

Query Probabilities from Lineage

Join = Saw \bowtie_{car} Drives

$$\begin{aligned} \Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5 \end{aligned}$$

Join

person	car	witness	Lineage
Jimmy	Honda	Cathy	$y \wedge w$
Hank	Honda	Cathy	$(x \vee z) \wedge w$

- $\Pr(\text{Jimmy} \in (\text{Saw} \bowtie_{\text{car}} \text{Drives})) = \Pr(y \wedge w) = \Pr(y) \times \Pr(w) = 0.4 \times 0.5 = 0.2$
- $\Pr(\text{Hank} \in (\text{Saw} \bowtie_{\text{car}} \text{Drives})) = \Pr((x \vee z) \wedge w)$
 $= \Pr(x \vee z) \times \Pr(w)$
 $= [\Pr(x) + \Pr(z) - \Pr(x \wedge z)] \times 0.5$
 $= [\Pr(x) + \Pr(z) - \Pr(x) \times \Pr(z)] \times 0.5$
 $= [0.2 + 0.8 - 0.2 \times 0.8] \times 0.5 = 0.42$

Theorem: SPJUD-query evaluation over PrRBDs with boolean-formulas lineage is **#P-hard**, i.e. intractable

Query Probabilities from Lineage

Join = Saw \bowtie_{car} Drives

$$\begin{aligned}\Pr(x \text{ is true}) &= 0.2 & \Pr(z \text{ is true}) &= 0.8 \\ \Pr(y \text{ is true}) &= 0.4 & \Pr(w \text{ is true}) &= 0.5\end{aligned}$$

Join

person	car	witness	Lineage
Jimmy	Honda	Cathy	$y \wedge w$
Hank	Honda	Cathy	$(x \vee z) \wedge w$

- $\Pr(\text{Jimmy} \in (\text{Saw} \bowtie_{\text{car}} \text{Drives})) = \Pr(y \wedge w) = \Pr(y) \times \Pr(w) = 0.4 \times 0.5 = 0.2$
- $\Pr(\text{Hank} \in (\text{Saw} \bowtie_{\text{car}} \text{Drives})) = \Pr((x \vee z) \wedge w)$

In general:

$$\Pr(\text{lineage}) = \Pr(\varphi)$$

where φ is a prop. formula

$$= \Pr(x \vee z) \times \Pr(w)$$

$$= [\Pr(x) + \Pr(z) - \Pr(x \wedge z)] \times 0.5$$

$$= [\Pr(x) + \Pr(z) - \Pr(x) \times \Pr(z)] \times 0.5$$

$$= [0.2 + 0.8 - 0.2 \times 0.8] \times 0.5 = 0.42$$

Theorem:

SPJUD-query evaluation over PrRBDs with boolean-formulas lineage is **#P-hard**, i.e. intractable

#P Functions

- Probability computation is a **function** and not a decision problem
- Usually complexity is studied for **decision** problems: $P(x) = \text{yes/no}$
- Complexity classes for probability computation are for classes of functions
- **#P functions**: $f(x) = n$
 - there is a PTIME non-deterministic Turing machine M_f
 - $n =$ the number of accepting runs of M_f on x , i.e., of $M_f(x)$
- #P functions are **counting** counterparts of **NP** decision problems
- Example of #P-complete function:
#2DNF: count number of evaluations for 2DNF propositional formulas
- #P-comp. functions are counter counterparts of NP-comp. problems

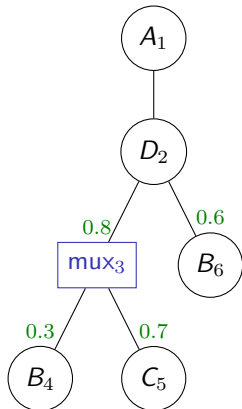
Part III: Querying Probabilistic Databases

- ▶ Semantics
- ▶ Lineage computation and #P-Hardness
- ▶ Special tractable case within Probabilistic XML

Algorithm for TP over local dependencies

[Kimelfeld and Sagiv, 2007]

Bottom-up dynamic programming algorithm. Query: /A//B



	A_1	D_2	mux_3	B_4	C_5	B_6
/B					0	
//B					0	
/A//B				0	0	0

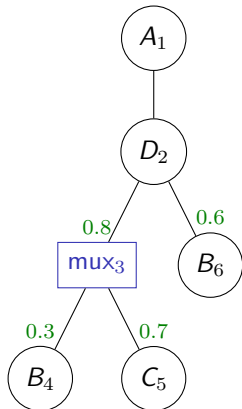
mux convex sum

ordinary inclusion-exclusion

Algorithm for TP over local dependencies

[Kimelfeld and Sagiv, 2007]

Bottom-up dynamic programming algorithm. Query: /A//B



	A_1	D_2	mux_3	B_4	C_5	B_6
/B			0.3		0	
//B			0.3		0	
/A//B			0	0	0	0

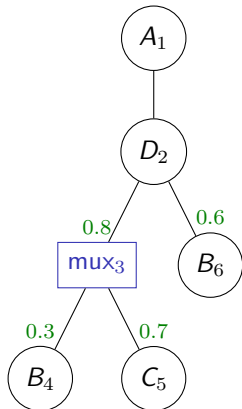
mux convex sum

ordinary inclusion-exclusion

Algorithm for TP over local dependencies

[Kimelfeld and Sagiv, 2007]

Bottom-up dynamic programming algorithm. Query: /A//B



	A_1	D_2	mux_3	B_4	C_5	B_6
/B		0	0.3		0	
//B		0.696	0.3		0	
/A//B		0	0	0	0	0

mux convex sum

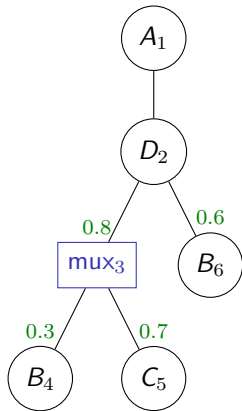
ordinary inclusion-exclusion

$$\begin{aligned}\Pr(D_2 \models //B) &= 1 - (1 - 0.8 \times \Pr(\text{mux}_3 \models /B)) \times (1 - 0.6 \times \Pr(B_6 \models /B)) \\ &= 1 - (1 - 0.8 \times 0.3) \times (1 - 0.6) = 0.696\end{aligned}$$

Algorithm for TP over local dependencies

[Kimelfeld and Sagiv, 2007]

Bottom-up dynamic programming algorithm. Query: /A//B



	A_1	D_2	mux_3	B_4	C_5	B_6
/B	0	0	0.3		0	
//B	0.696	0.696	0.3		0	
/A//B	0.696	0	0	0	0	0

mux convex sum

ordinary inclusion-exclusion

Part IV: To go further

Systems

Trio <http://infolab.stanford.edu/trio/>, useful to see lineage computation

MayBMS <http://maybms.sourceforge.net/>, full-fledged probabilistic relational DBMS, on top of PostgreSQL, usable for actual applications.

ProApproX <http://www.infres.enst.fr/~souihli/Publications.html> to play with various approximation and exact query evaluation methods for probabilistic XML.

Reading material

- ▶ An influential paper on **incomplete databases** [Imielinski and Lipski, 1984]
- ▶ A book on **probabilistic relational databases**, focused around TIDs/BIDs and MayBMS [Suciu et al., 2011]
- ▶ An in-depth presentation of **MayBMS** [Koch, 2009]
- ▶ A gentle presentation of relational and XML probabilistic **models** [Kharlamov and Senellart, 2011]
- ▶ A survey of **probabilistic XML** [Kimelfeld and Senellart, 2013]

Merci.

Wabdam

- Talel Abdessalem, M. Lamine Ba, and Pierre Senellart. A probabilistic XML merging tool. In *Proc. EDBT*, pages 538–541, Uppsala, Sweden, March 2011. Demonstration.
- 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, 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.
- 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.

- M. Lamine Ba, Talel Abdesslem, and Pierre Senellart. Uncertain version control in open collaborative editing of tree-structured documents. In *Proc. DocEng*, Florence, Italy, September 2013.
- Daniel Barbará, Hector Garcia-Molina, and Daryl Porter. The management of probabilistic data. *IEEE Transactions on Knowledge and Data Engineering*, 4(5):487–502, 1992.
- Pablo Barceló, Leonid Libkin, Antonella Poggi, and Cristina Sirangelo. XML with incomplete information: models, properties, and query answering. In *Proc. PODS*, pages 237–246, New York, NY, 2009. ACM.
- 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.
- Omar Benjelloun, Anish Das Sarma, Alon Y. Halevy, and Jennifer Widom. Uldbs: Databases with uncertainty and lineage. In *VLDB*, pages 953–964, 2006.

- Nilesh Dalvi, Christopher Ré, and Dan Suciu. Probabilistic databases: Diamonds in the dirt. *Communications of the ACM*, 52(7), 2009.
- Nilesh N. Dalvi and Dan Suciu. Efficient query evaluation on probabilistic databases. In *VLDB*, pages 864–875, 2004.
- Nilesh N. Dalvi and Dan Suciu. Efficient query evaluation on probabilistic databases. *VLDB Journal*, 16(4), 2007.
- Robert Fink, Andrew Hogue, Dan Olteanu, and Swaroop Rath. *SPROUT²*: a squared query engine for uncertain web data. In *SIGMOD*, 2011.
- José Galindo, Angelica Urrutia, and Mario Piattini. *Fuzzy Databases: Modeling, Design And Implementation*. IGI Global, 2005.
- Todd J. Green and Val Tannen. Models for incomplete and probabilistic information. In *Proc. EDBT Workshops, IIDB*, Munich, Germany, March 2006.
- Tomasz Imielinski and Witold Lipski. Incomplete information in relational databases. *Journal of the ACM*, 31(4):761–791, 1984.

- Evgeny Kharlamov and Pierre Senellart. Modeling, querying, and mining uncertain XML data. In Andrea Tagarelli, editor, *XML Data Mining: Models, Methods, and Applications*. IGI Global, 2011.
- B. Kimelfeld and Y. Sagiv. Matching twigs in probabilistic XML. In *Proc. VLDB*, Vienna, Austria, September 2007.
- 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.
- Christoph Koch. MayBMS: A system for managing large uncertain and probabilistic databases. In Charu Aggarwal, editor, *Managing and Mining Uncertain Data*. Springer-Verlag, 2009.
- Laks V. S. Lakshmanan, Nicola Leone, Robert B. Ross, and V. S. Subrahmanian. ProbView: A flexible probabilistic database system. *ACM Transactions on Database Systems*, 22(3), 1997.
- Christopher Ré and Dan Suciu. Materialized views in probabilistic databases: for information exchange and query optimization. In *Proc. VLDB*, 2007.

Pierre Senellart, Avin Mittal, Daniel Muschick, Rémi Gilleron, and Marc Tommasi. Automatic wrapper induction from hidden-Web sources with domain knowledge. In *Proc. WIDM*, pages 9–16, Napa, USA, October 2008.

Dan Suciu, Dan Olteanu, Christopher Ré, and Christoph Koch. *Probabilistic Databases*. Morgan & Claypool, 2011.

Jennifer Widom. Trio: A system for integrated management of data, accuracy, and lineage. In *Proc. CIDR*, Asilomar, CA, USA, January 2005.

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.