

Challenges for View-Based Query Answering over Probabilistic XML

Bogdan Cautis

Télécom ParisTech

Evgeny Kharlamov

Free University of Bozen-Bolzano

University of Oxford



FREIE UNIVERSITÄT BOZEN

LIBERA UNIVERSITÀ DI BOLZANO

FREE UNIVERSITY OF BOZEN · BOLZANO



AMW, Santiago de Chile, May 2011

Uncertain Data

- is **commonplace**:
 - (Web) information **extraction**
 - **Processing** manually entered data (such as census forms)
 - Data **integration**, data **cleaning**
 - Managing scientific data; **sensor data**

Uncertain Data

- is **commonplace**:
 - (Web) information **extraction**
 - **Processing** manually entered data (such as census forms)
 - Data **integration**, data **cleaning**
 - Managing scientific data; **sensor data**
- **Probabilities** is a way to capture uncertainty

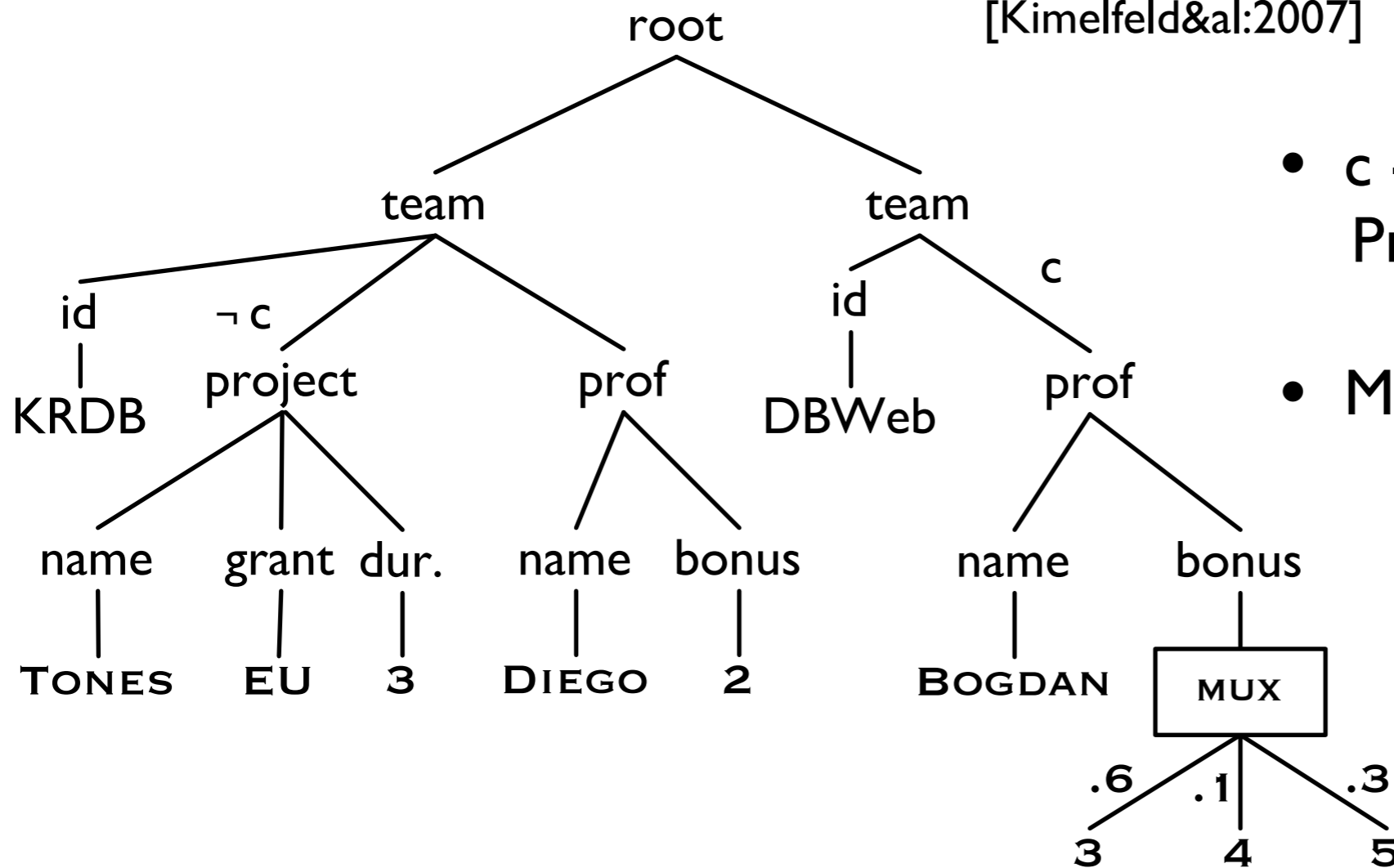
Uncertain Data

- is **commonplace**:
 - (Web) information **extraction**
 - **Processing** manually entered data (such as census forms)
 - Data **integration**, data **cleaning**
 - Managing scientific data; **sensor data**
- **Probabilities** is a way to capture uncertainty
- Our focus is probabilistic **XML**
since we see the Web as the prime source of uncertainty

PrXML Data Model

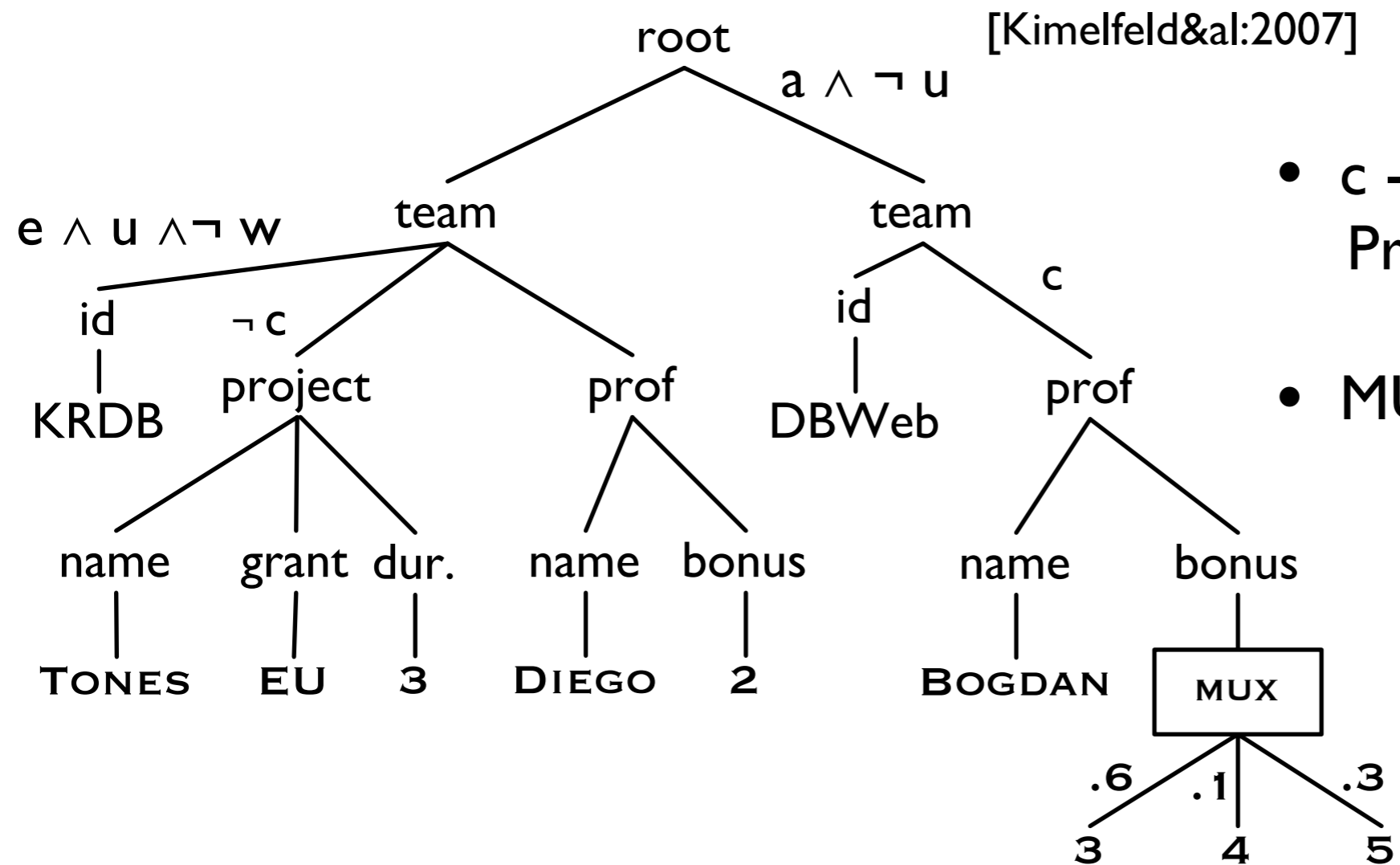
[Kimelfeld&al:2007]

[Senellart&al:2007]



- c - event: “current”
Pr(c) = 0.4
- MUX - mutually exclusive options

PrXML Data Model



[Kimelfeld&al:2007]

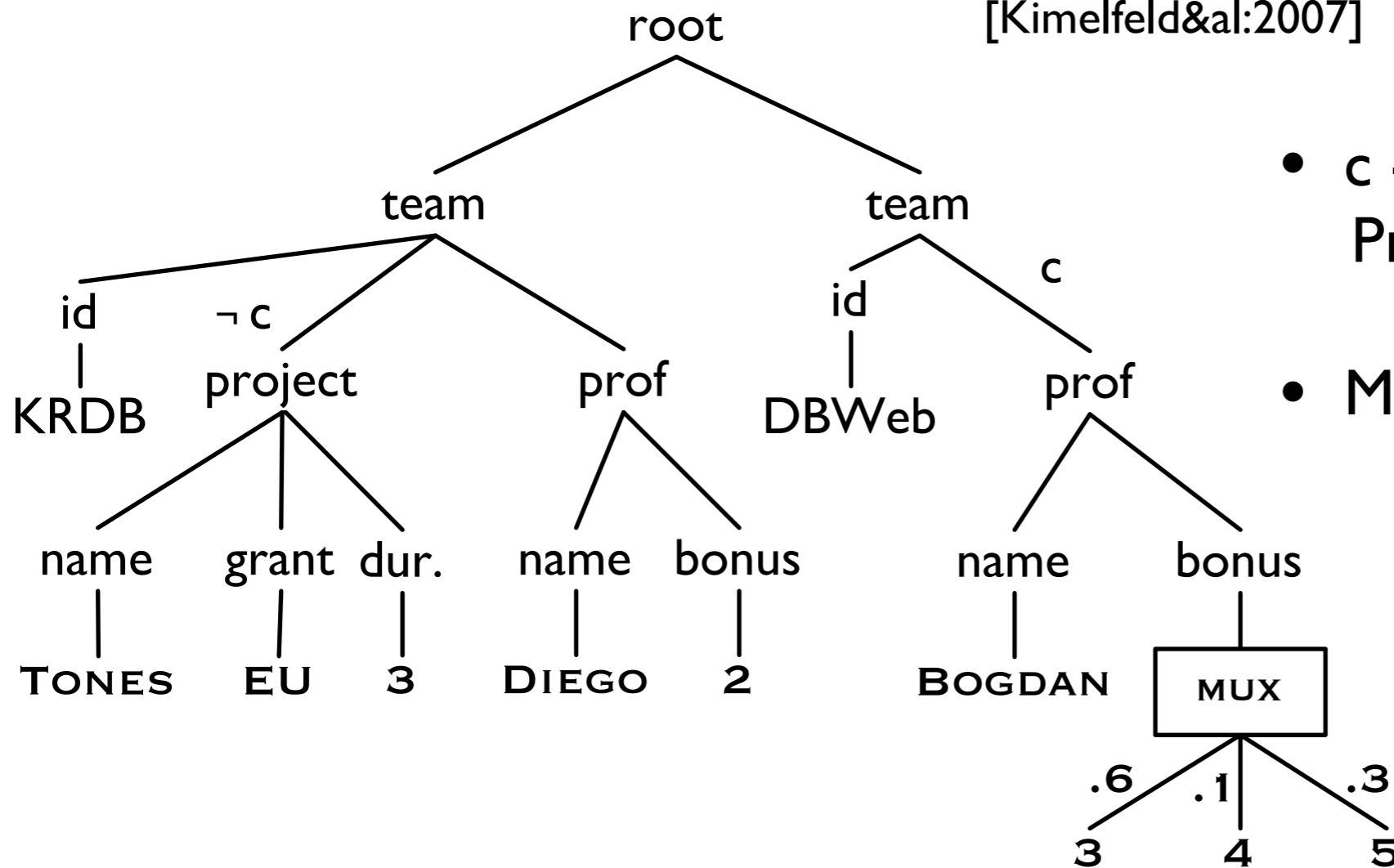
[Senellart&al:2007]

- **c** - event: “current”
Pr(c) = 0.4
- **MUX** - mutually exclusive options

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]

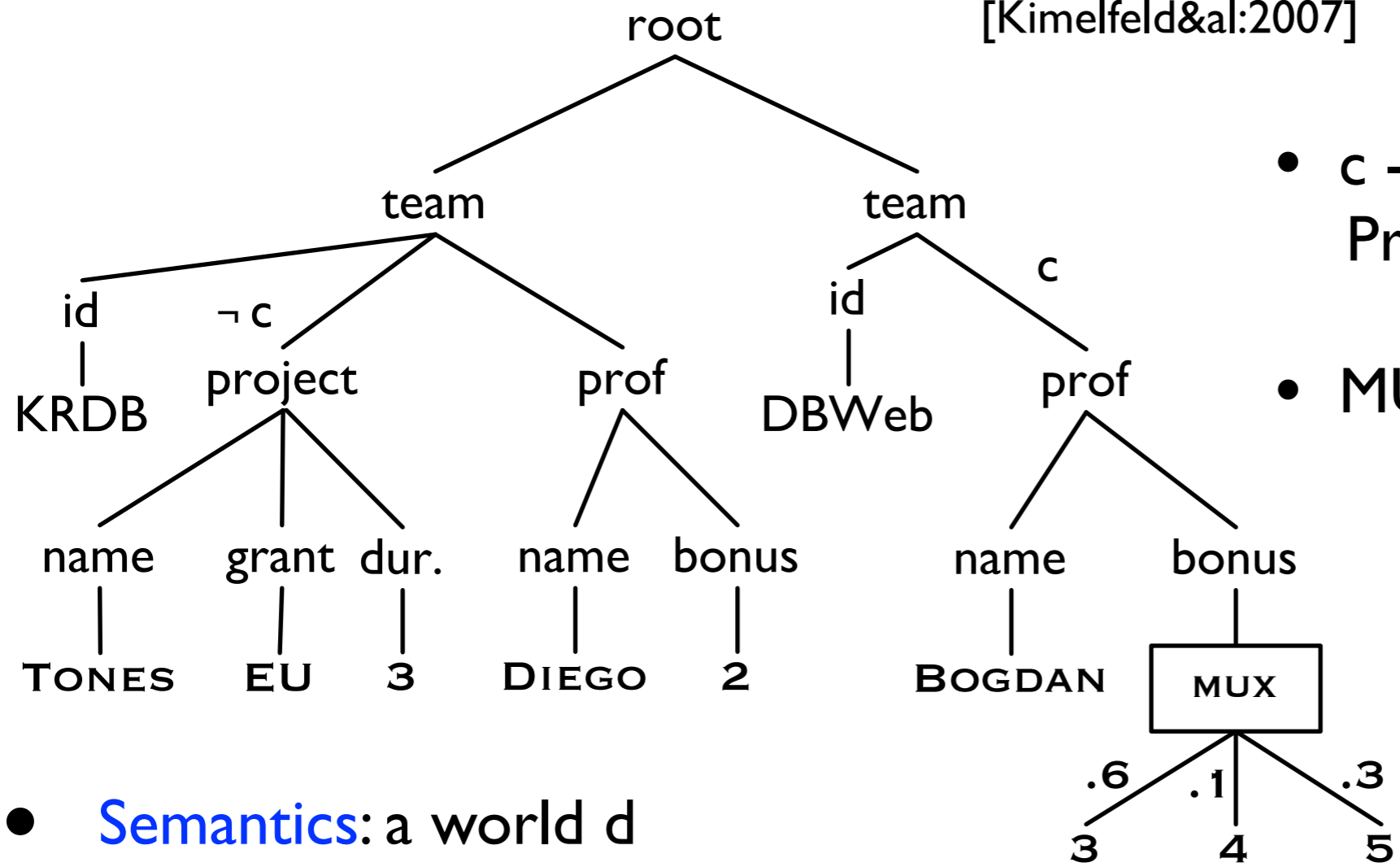


- c - event: “current”
Pr(c) = 0.4
- MUX - mutually exclusive options

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



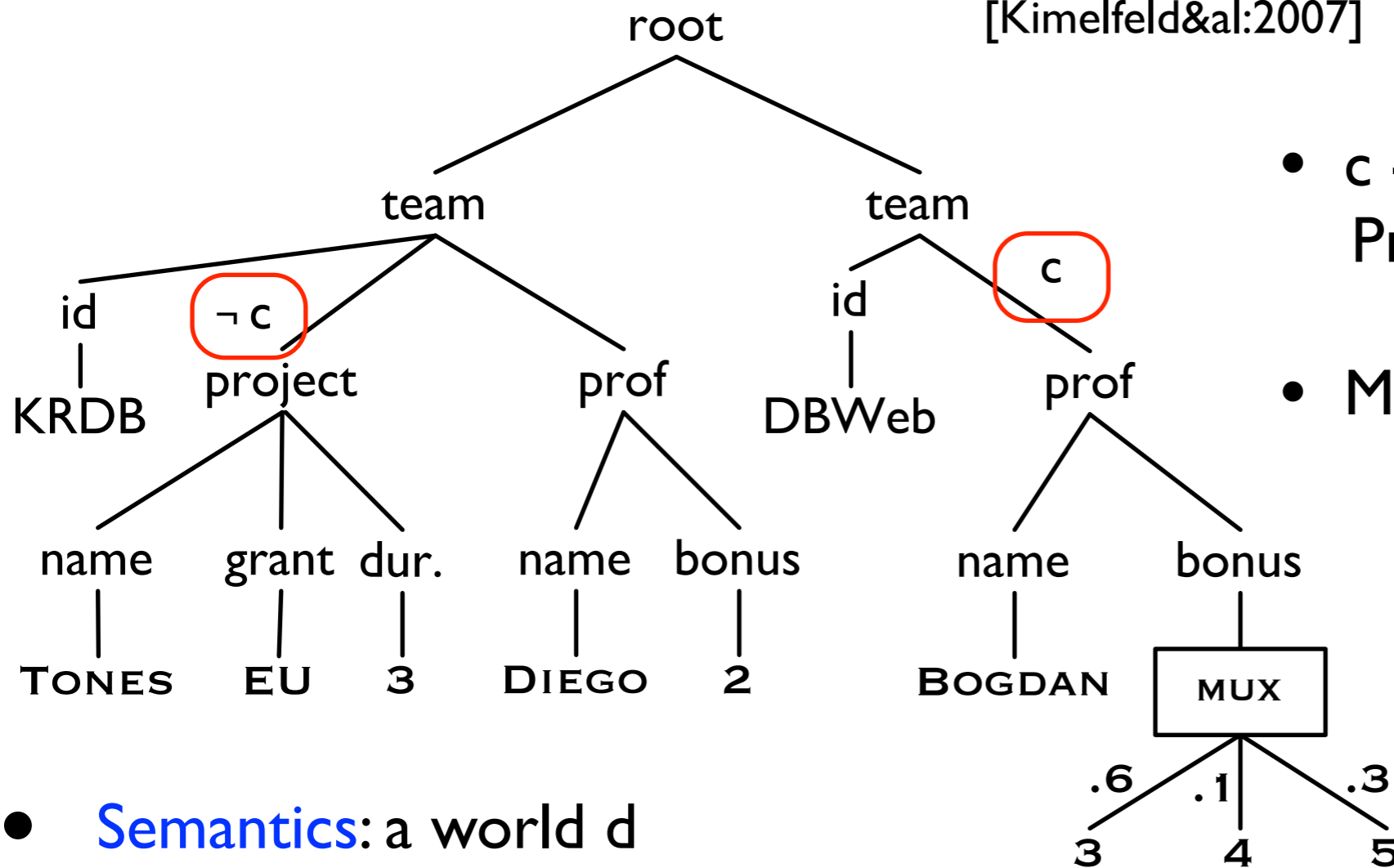
- c - event: “current”
Pr(c) = 0.4
- MUX - mutually exclusive options

- **Semantics:** a world d
 - c = true (current data)
 - MUX:4
 - Pr(d) = 0.4 x 0.1

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



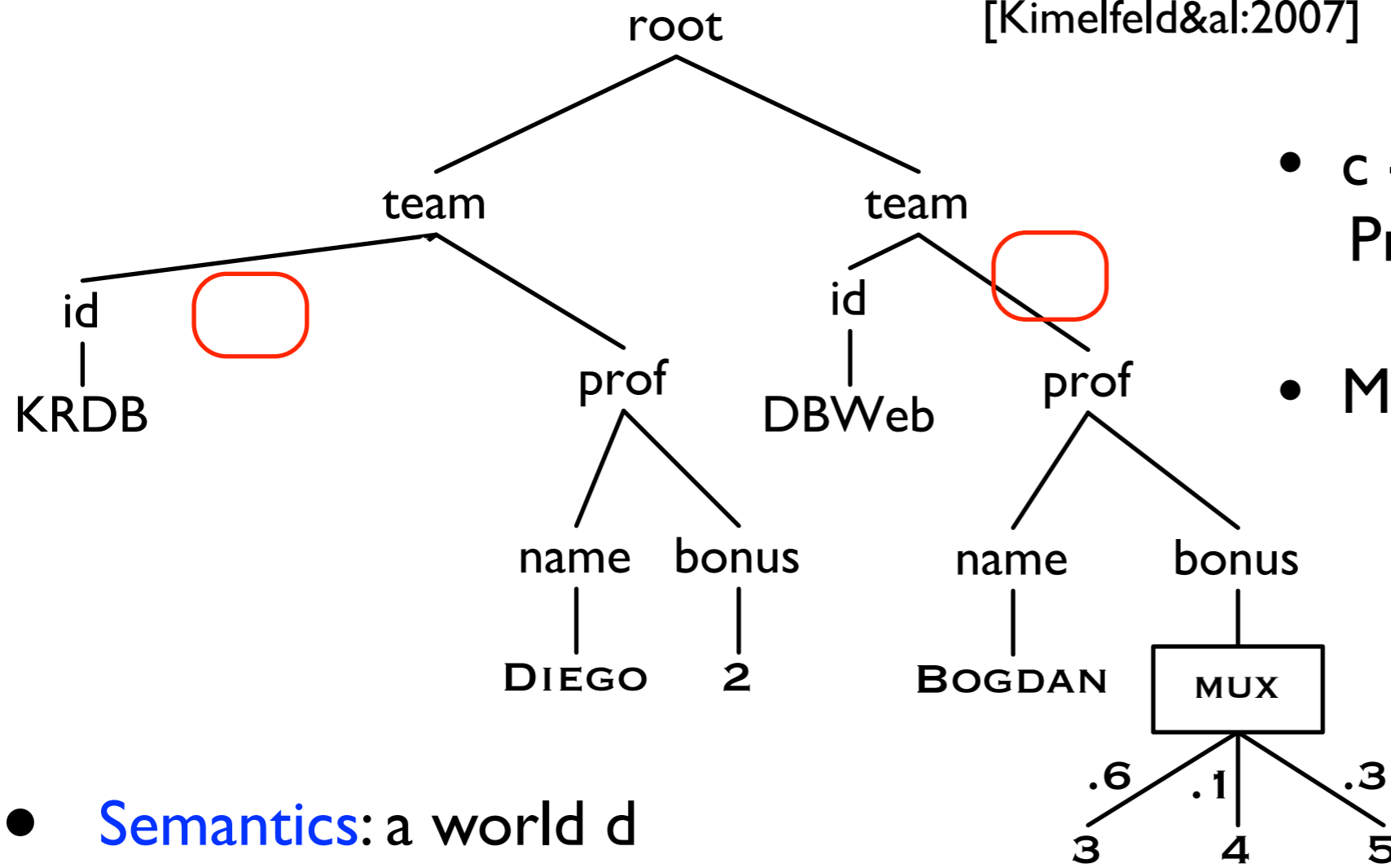
- c - event: “current”
Pr(c) = 0.4
- MUX - mutually exclusive options

- **Semantics:** a world d
 - **c = true** (current data)
 - MUX:4
 - Pr(d) = 0.4 x 0.1

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



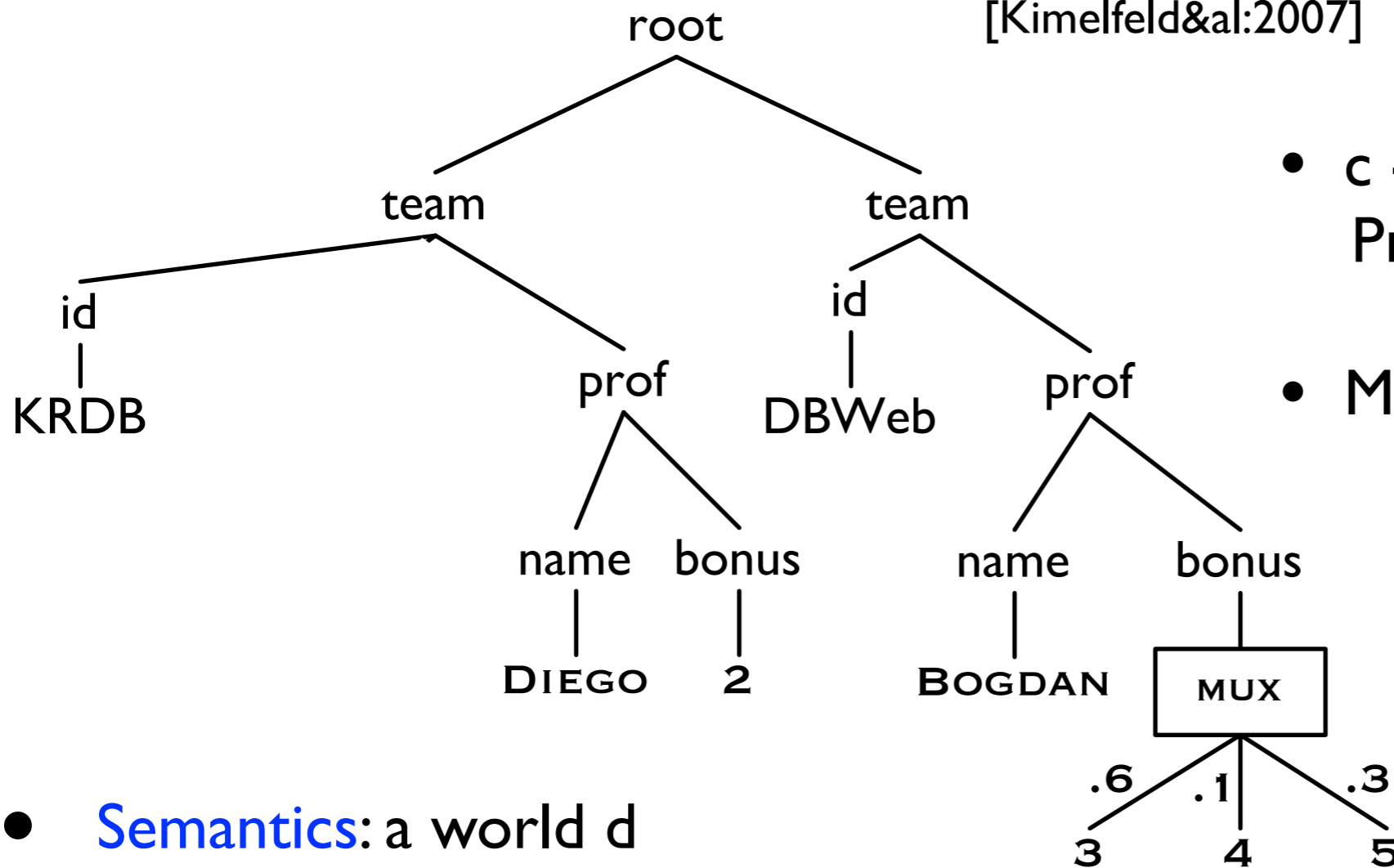
- c - event: “current”
Pr(c) = 0.4
- MUX - mutually exclusive options

- **Semantics:** a world d
- **c = true** (current data)
- MUX:4
- Pr(d) = 0.4 x 0.1

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



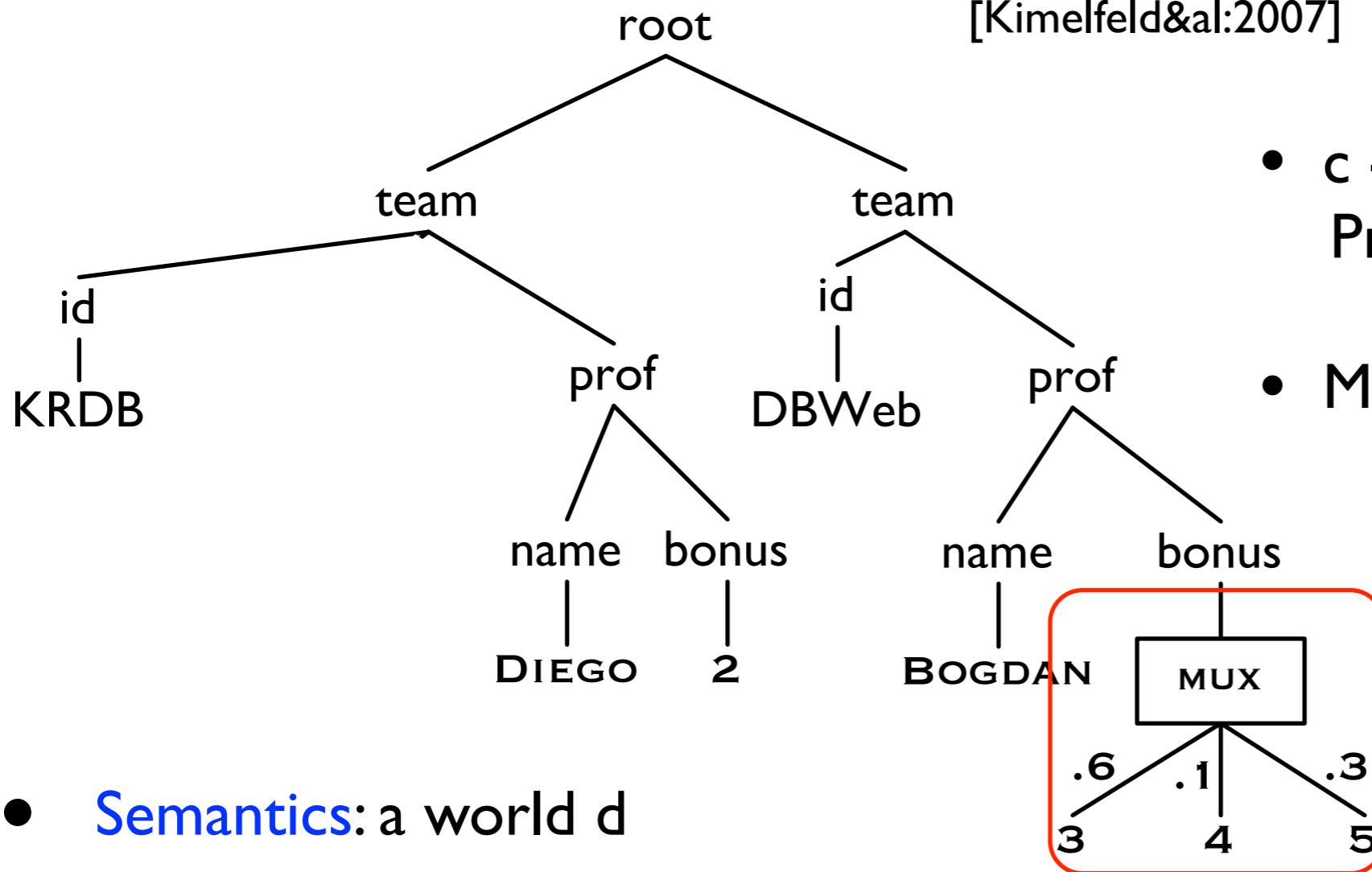
- c - event: “current”
 $\Pr(c) = 0.4$
- MUX - mutually exclusive options

- **Semantics:** a world d
 - c = true (current data)
 - MUX:4
 - $\Pr(d) = 0.4 \times 0.1$

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



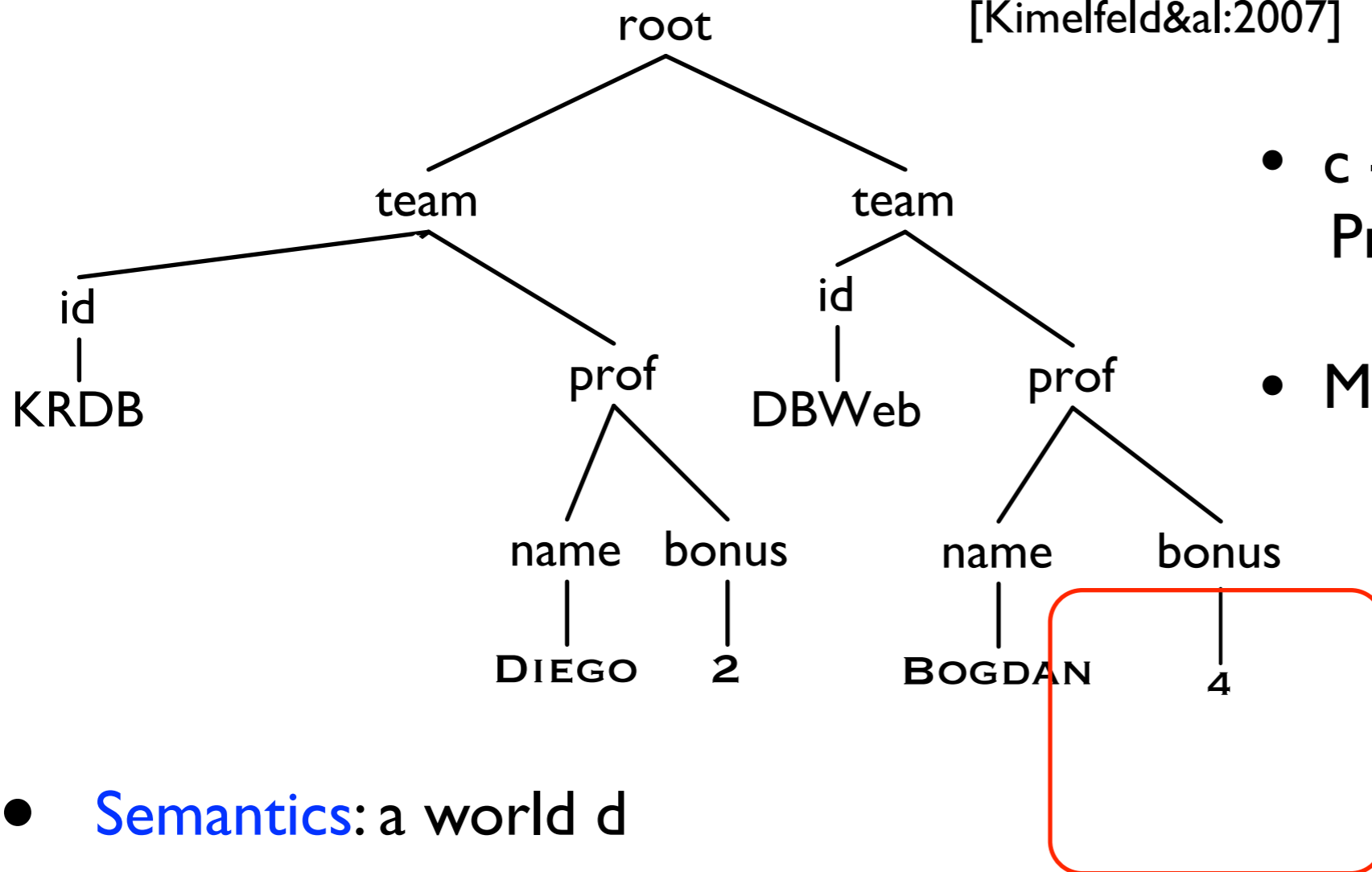
- c - event: “current”
 $\Pr(c) = 0.4$
- MUX - mutually exclusive options

- **Semantics:** a world d
 - c = true (current data)
 - **MUX:4**
 - $\Pr(d) = 0.4 \times 0.1$

PrXML Data Model

[Kimelfeld&al:2007]

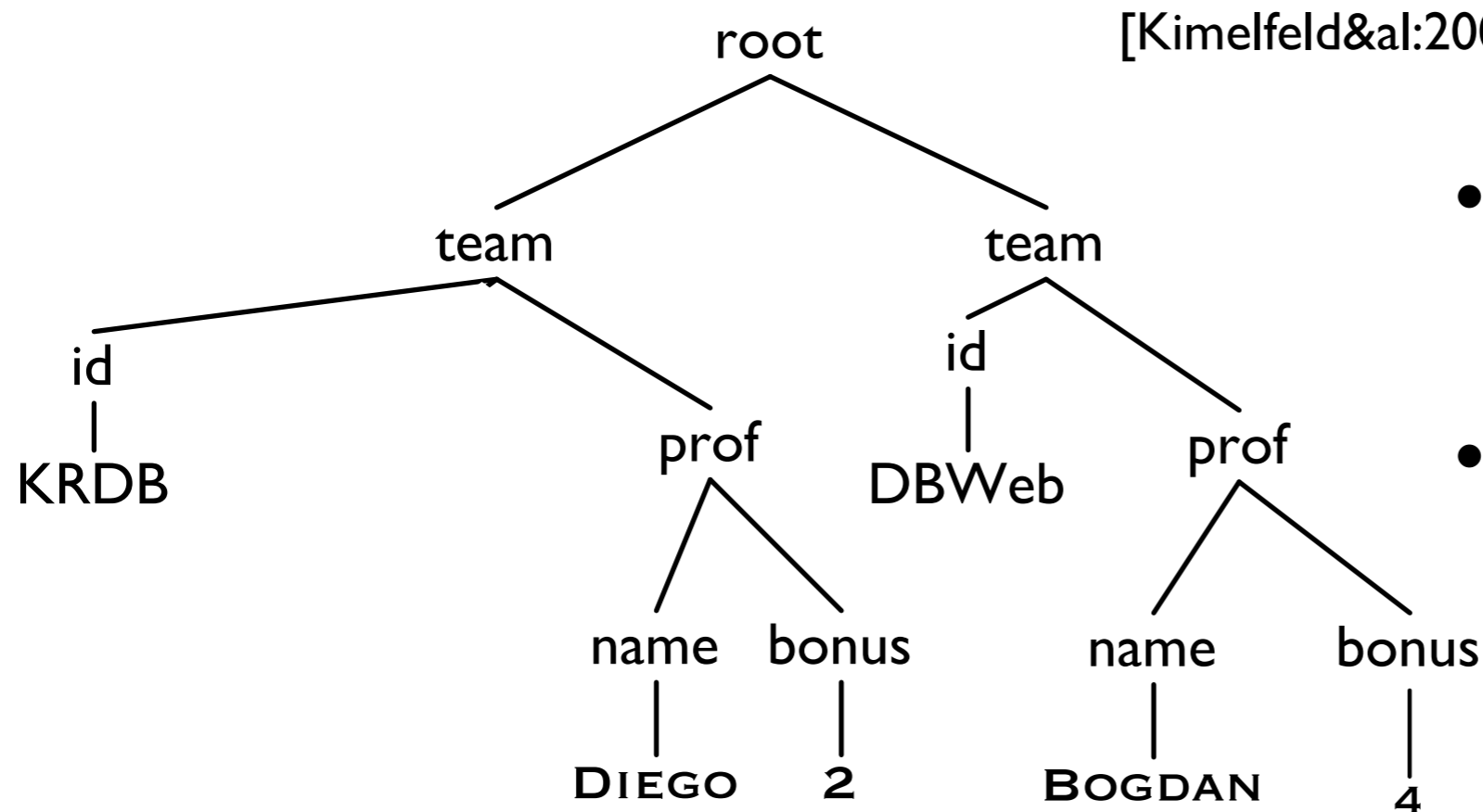
[Senellart&al:2007]



- c - event: “current”
 $\Pr(c) = 0.4$
- MUX - mutually exclusive options

- **Semantics:** a world d
 - $c = \text{true}$ (current data)
 - **MUX:4**
 - $\Pr(d) = 0.4 \times 0.1$

PrXML Data Model



[Kimelfeld&al:2007]

[Senellart&al:2007]

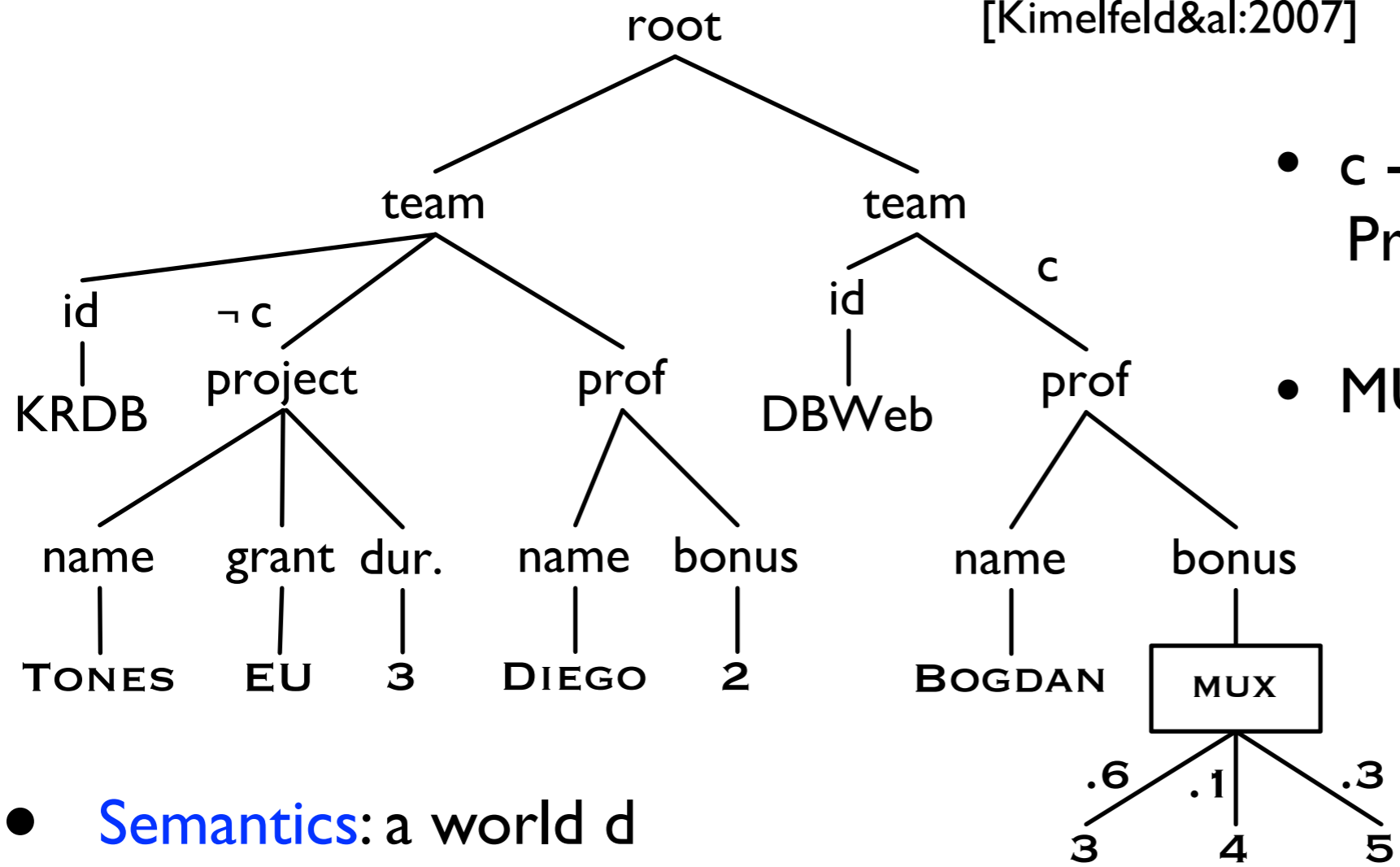
- c - event: “current”
 $\Pr(c) = 0.4$
- MUX - mutually exclusive options

- **Semantics:** a world d
 - c = true (current data)
 - MUX:4
 - $\Pr(d) = 0.4 \times 0.1$

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



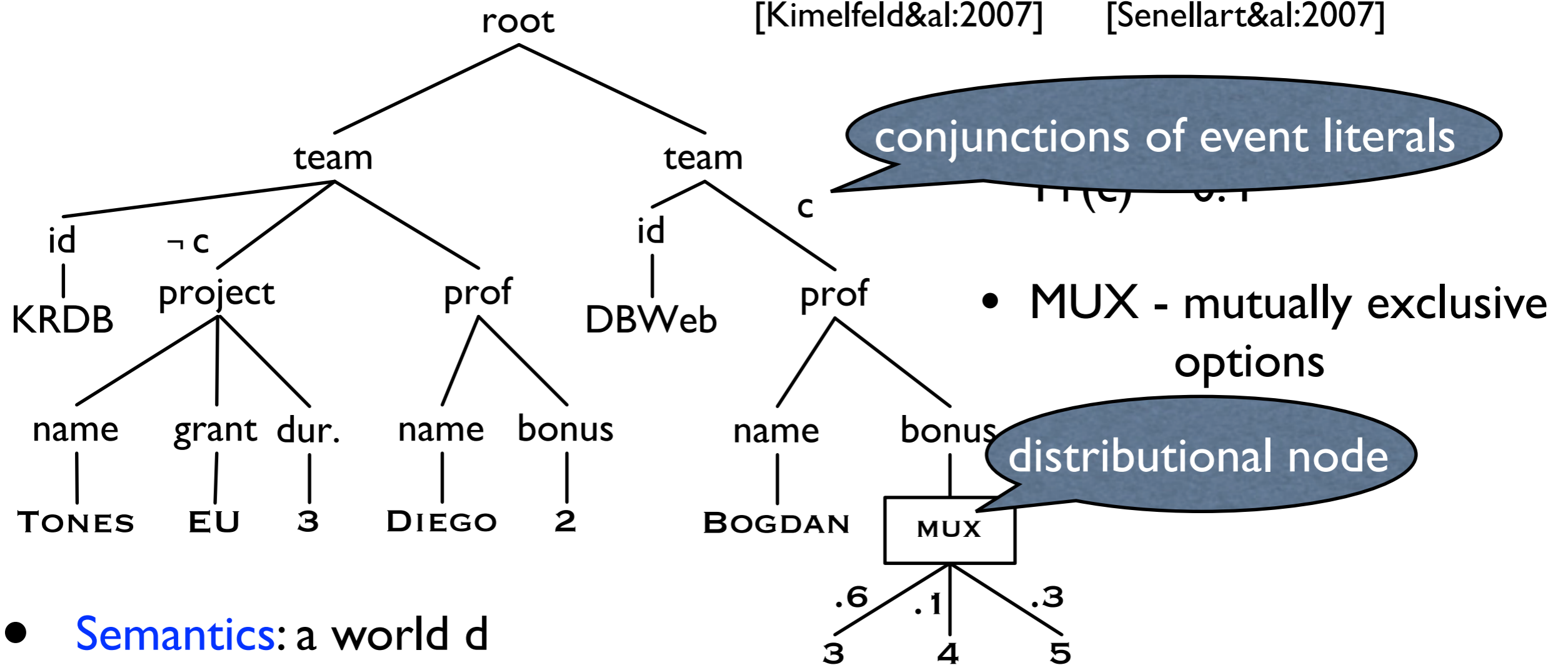
- c - event: “current”
Pr(c) = 0.4
- MUX - mutually exclusive options

- **Semantics:** a world d
 - c = true (current data)
 - MUX:4
 - Pr(d) = 0.4 x 0.1

PrXML Data Model

[Kimelfeld&al:2007]

[Senellart&al:2007]



- **Semantics:** a world d

- $c = \text{true}$ (current data)

- MUX:4

Global-PrXML (**G-PrXML**): conjunctions of event literals

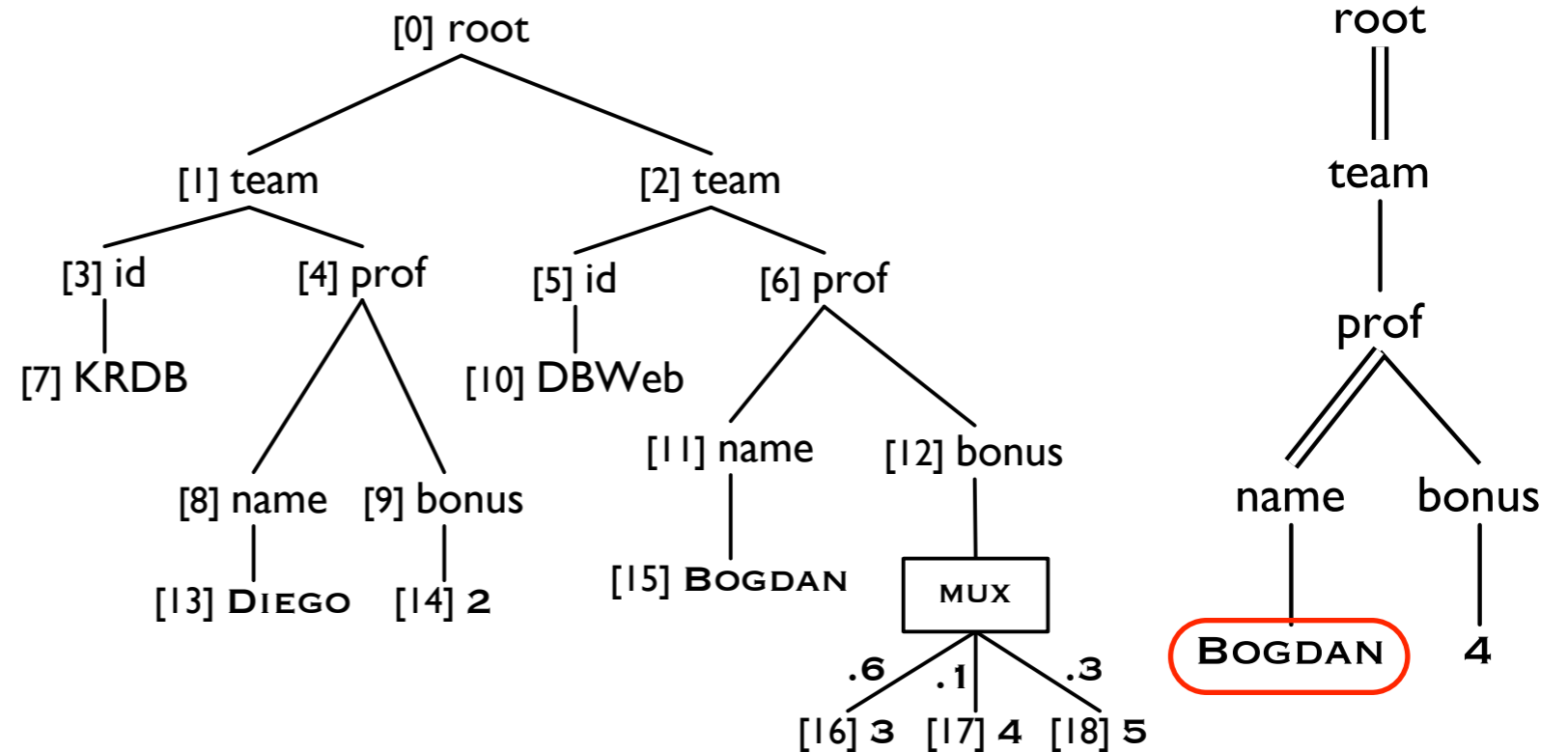
- $\Pr(d) = 0.4 \times$ Local-PrXML (**L-PrXML**): distributional nodes

Tree-Pattern Queries over PrXML

[Kimelfeld&al:2007]

[Senellart&al:2007]

Query Q: Return Bogdans who received a bonus of 4?



Tree-Pattern Queries over PrXML

[Kimelfeld&al:2007]

[Senellart&al:2007]

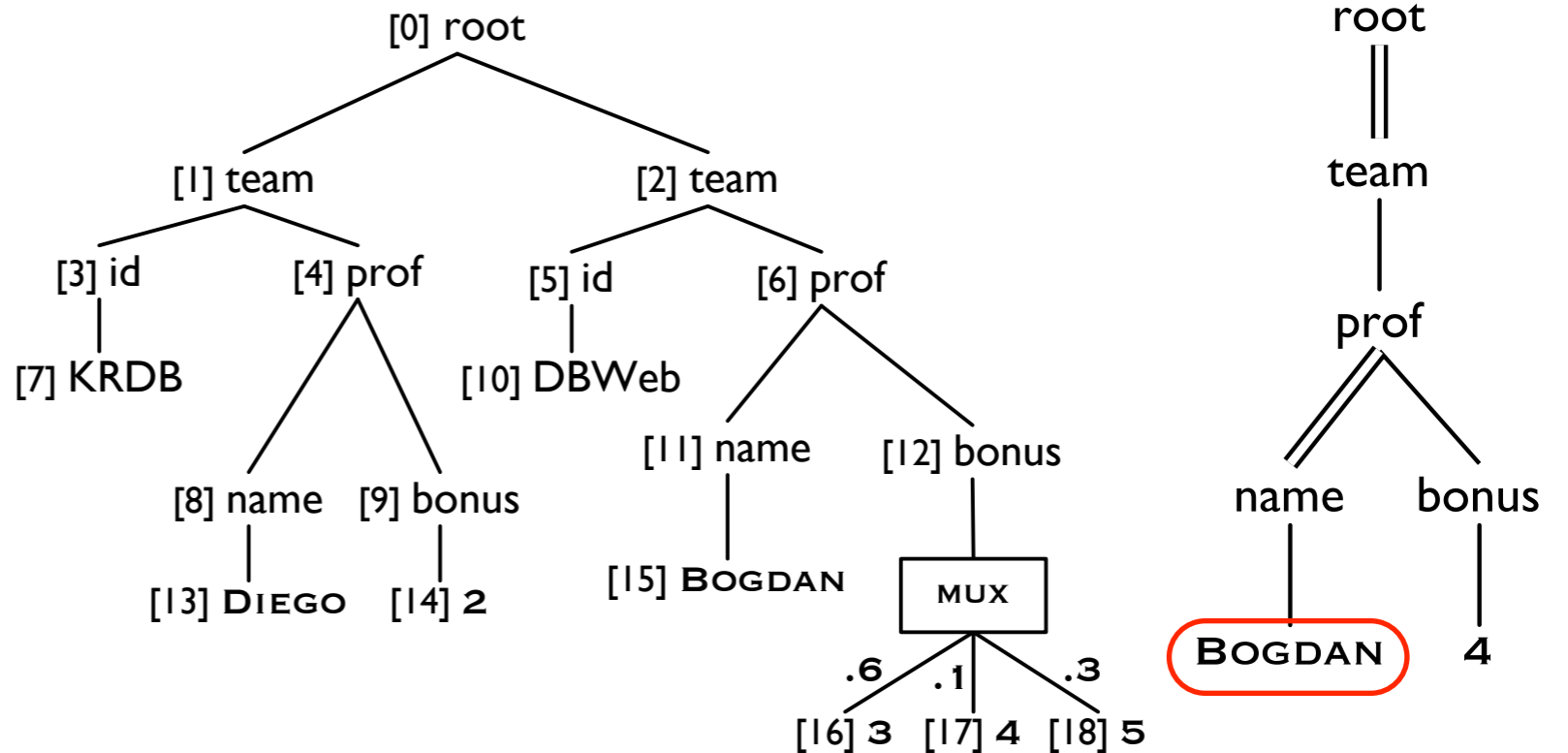
Query Q: Return Bogdans who received a bonus of 4?

Answers over worlds:

$Q(w3) = \text{no}, \quad \text{Pr}(w3)=0.6$

$Q(w4) = [15], \quad \text{Pr}(w4)=0.1$

$Q(w5) = \text{no}, \quad \text{Pr}(w5)=0.3$



Tree-Pattern Queries over PrXML

[Kimelfeld&al:2007]

[Senellart&al:2007]

Query Q: Return Bogdans who received a bonus of 4?

Answers over worlds:

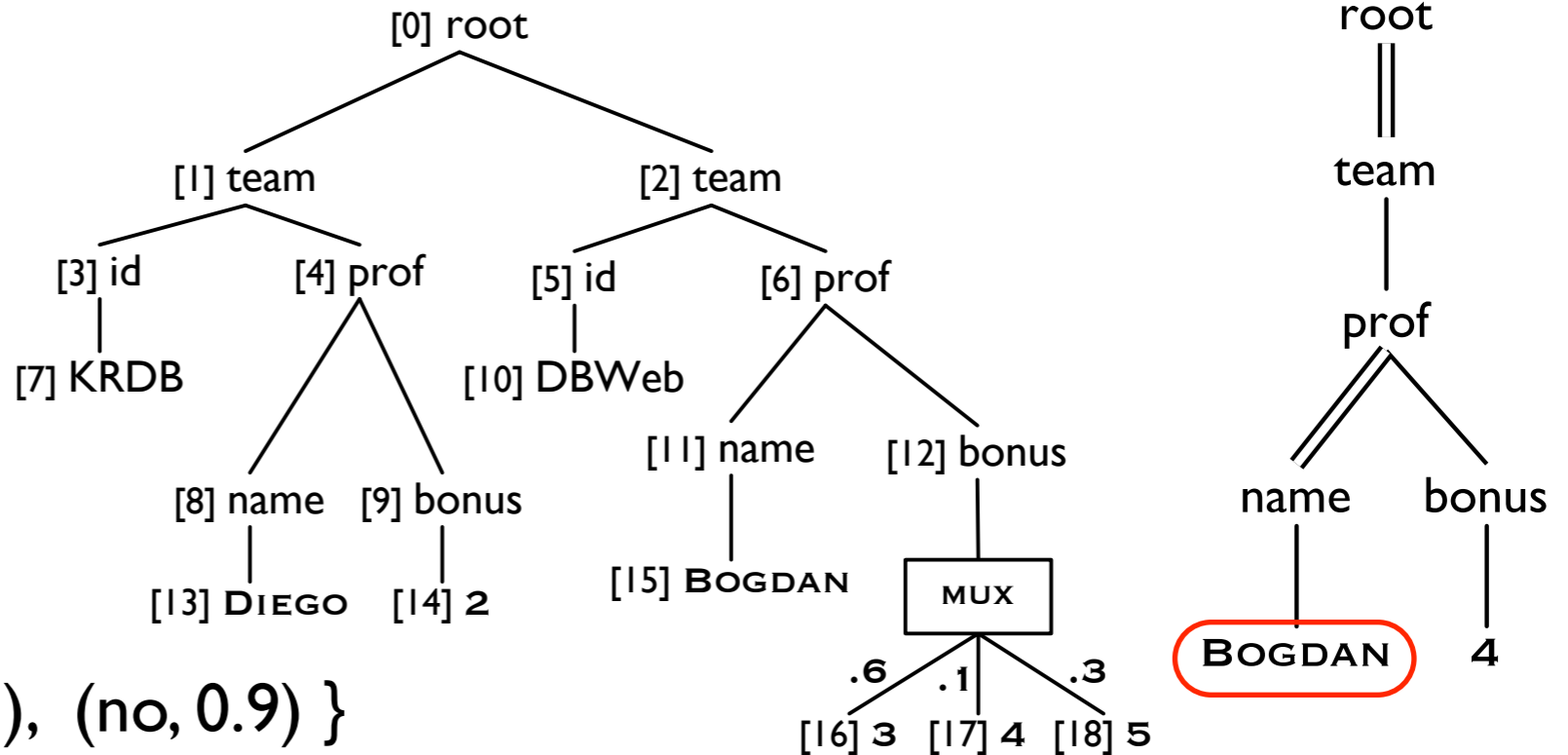
$Q(w3) = \text{no}, \quad \text{Pr}(w3)=0.6$

$Q(w4) = [15], \quad \text{Pr}(w4)=0.1$

$Q(w5) = \text{no}, \quad \text{Pr}(w5)=0.3$

Answers over PrXML with

original Ids: $\{ (([15], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$



Tree-Pattern Queries over PrXML

[Kimelfeld&al:2007]

[Senellart&al:2007]

Query Q: Return Bogdans who received a bonus of 4?

Answers over worlds:

$Q(w3) = \text{no}, \quad \text{Pr}(w3)=0.6$

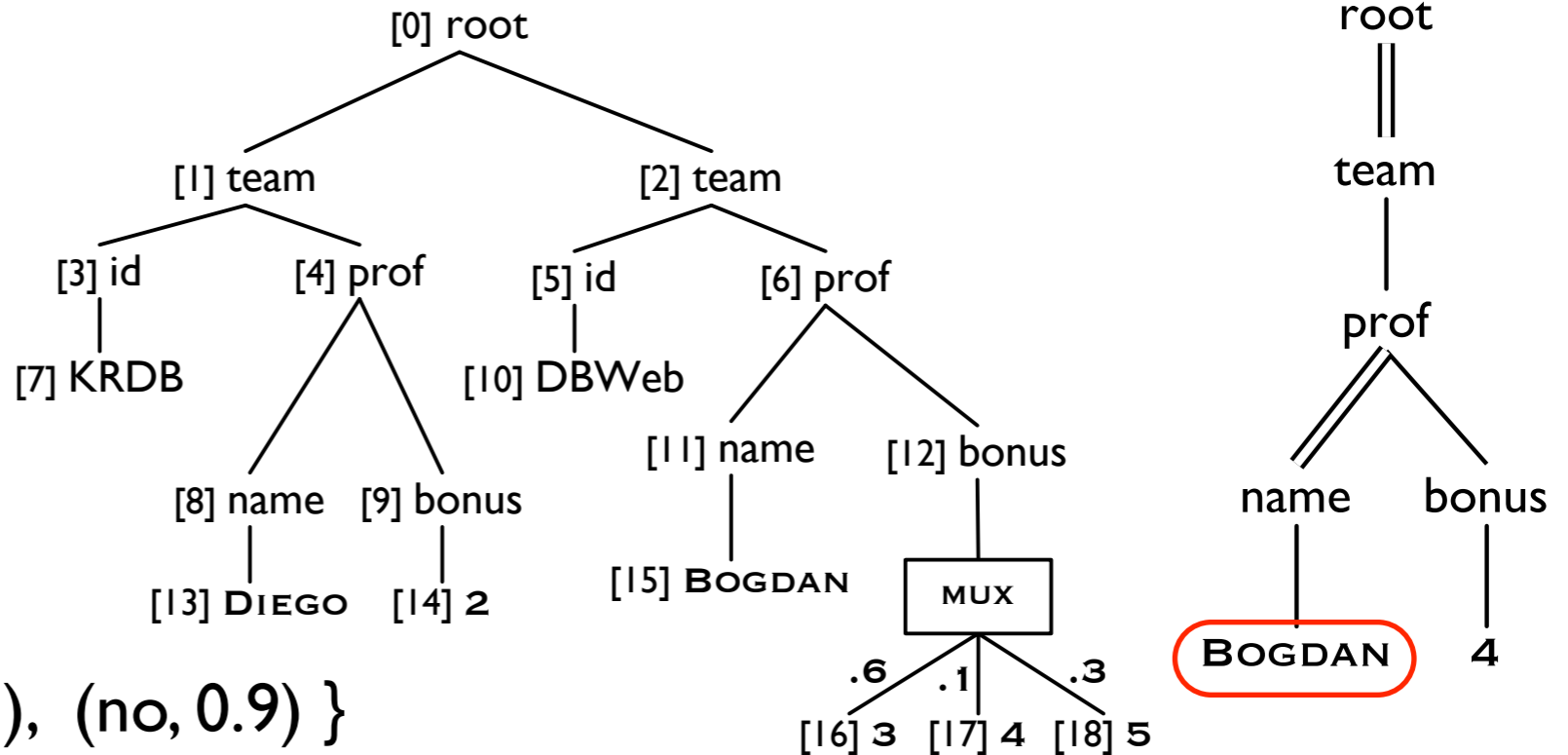
$Q(w4) = [15], \quad \text{Pr}(w4)=0.1$

$Q(w5) = \text{no}, \quad \text{Pr}(w5)=0.3$

Answers over PrXML with

original Ids: $\{ (([15], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$

fresh Ids: $\{ ([1], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$



Tree-Pattern Queries over PrXML

[Kimelfeld&al:2007]

[Senellart&al:2007]

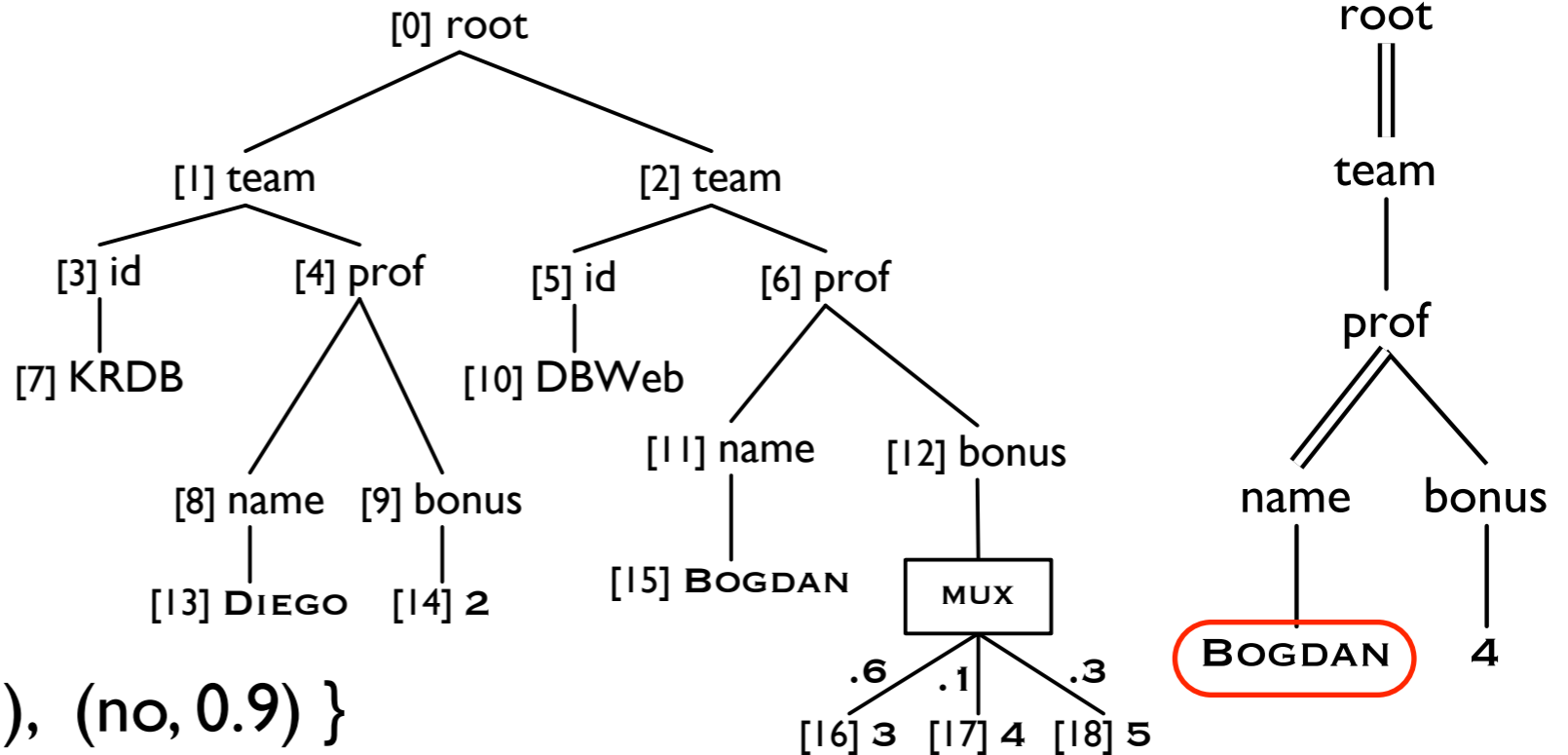
Query Q: Return Bogdans who received a bonus of 4?

Answers over worlds:

$Q(w3) = \text{no}, \quad \text{Pr}(w3)=0.6$

$Q(w4) = [15], \quad \text{Pr}(w4)=0.1$

$Q(w5) = \text{no}, \quad \text{Pr}(w5)=0.3$



Answers over PrXML with

original Ids: $\{ (([15], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$

fresh Ids: $\{ ([1], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$

Query answering for PrXML:

- **probability** computation:
 $\text{Pr}([i] \in Q(\text{random doc})) = ?$

Tree-Pattern Queries over PrXML

[Kimelfeld&al:2007]

[Senellart&al:2007]

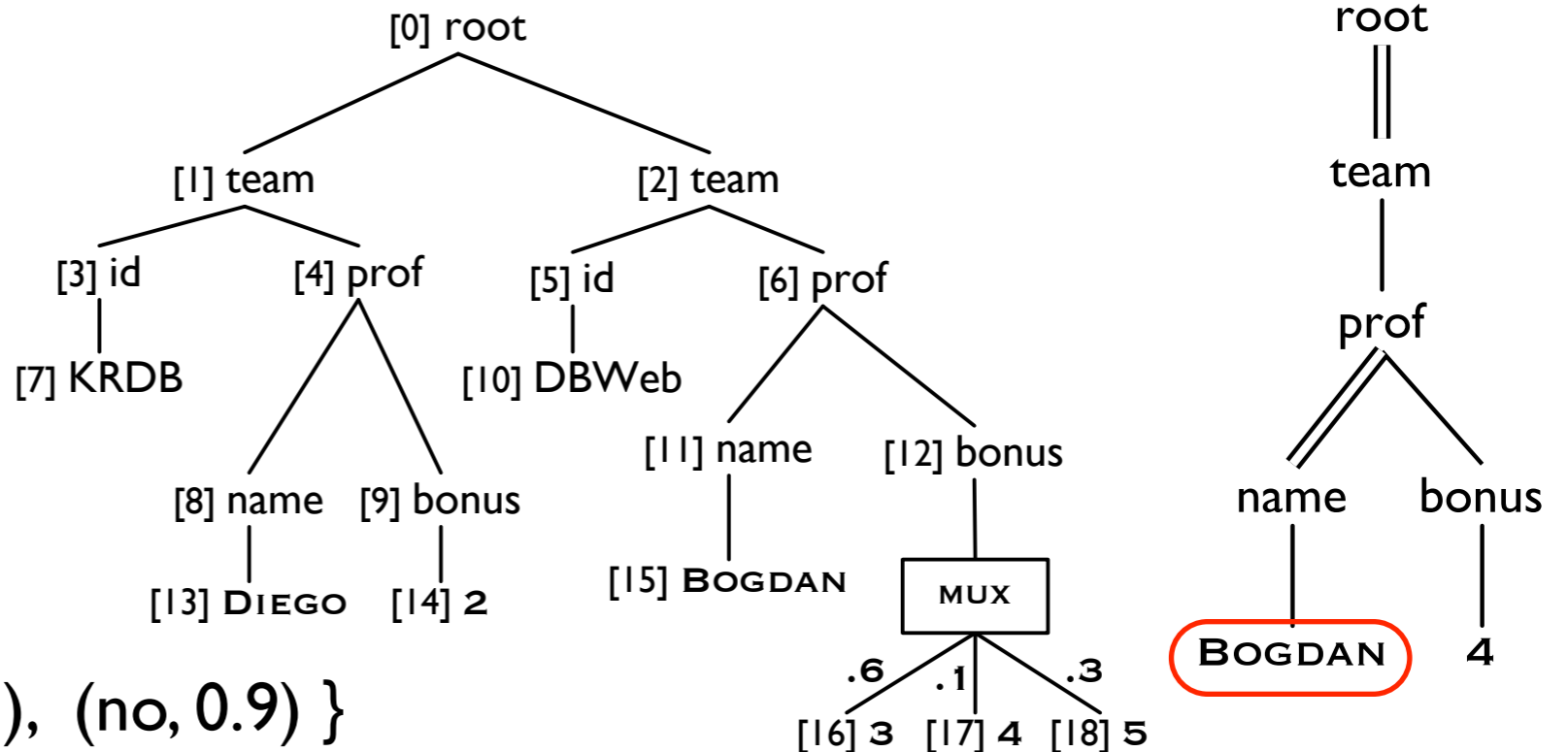
Query Q: Return Bogdans who received a bonus of 4?

Answers over worlds:

$Q(w3) = \text{no}, \quad \text{Pr}(w3)=0.6$

$Q(w4) = [15], \quad \text{Pr}(w4)=0.1$

$Q(w5) = \text{no}, \quad \text{Pr}(w5)=0.3$



Answers over PrXML with

original Ids: $\{ (([15], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$

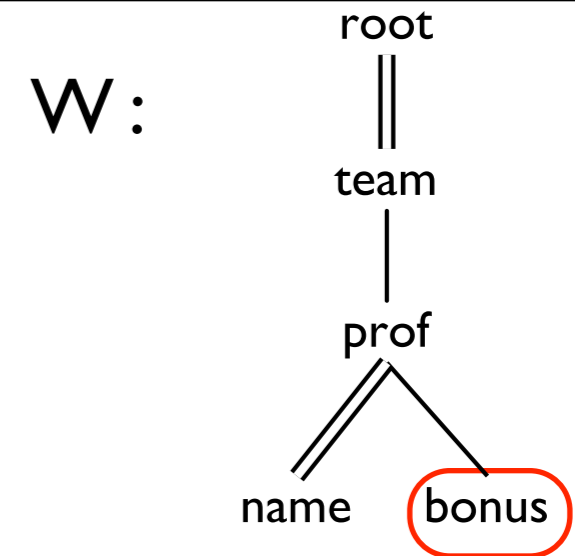
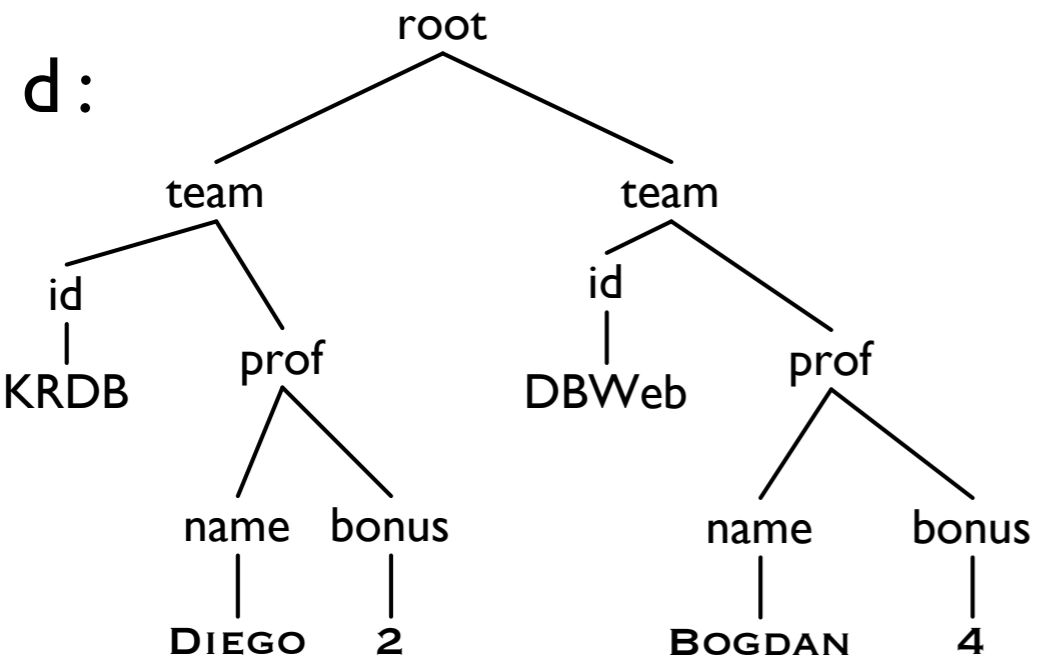
fresh Ids: $\{ ([1], \text{Bogdan}), 0.1), (\text{no}, 0.9) \}$

Query answering for PrXML:

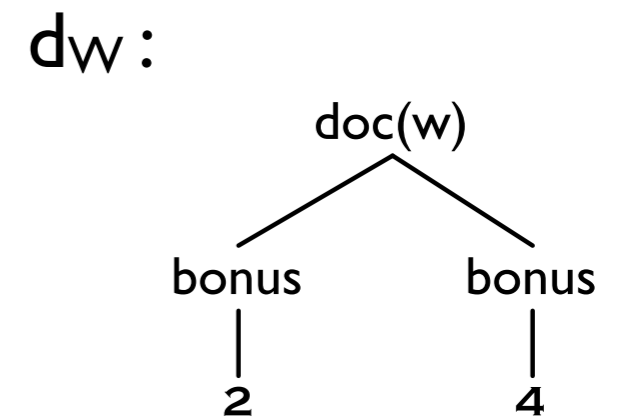
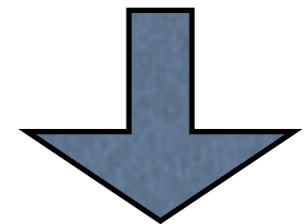
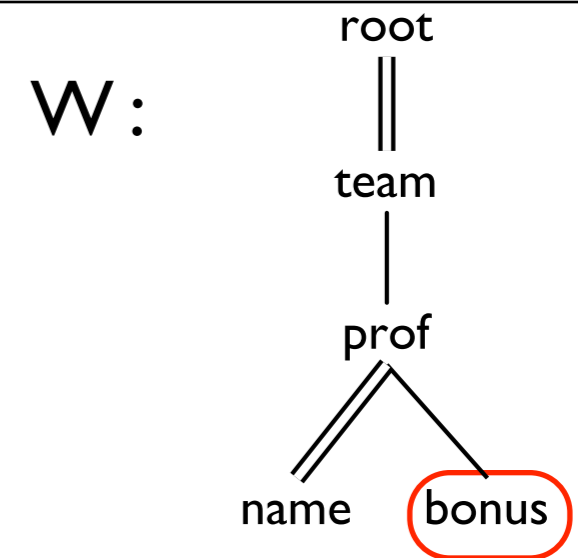
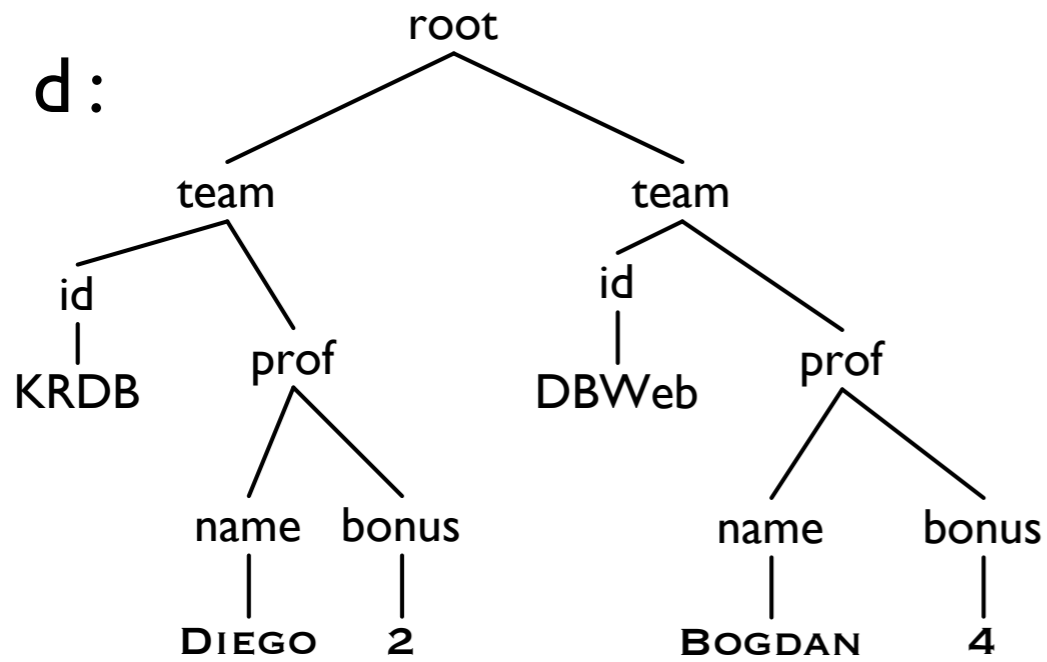
- **probability** computation:
 $\text{Pr}([i] \in Q(\text{random doc})) = ?$

- **PTIME** for local PrXML
- dynamic programming
- **#P-hard** for global PrXML

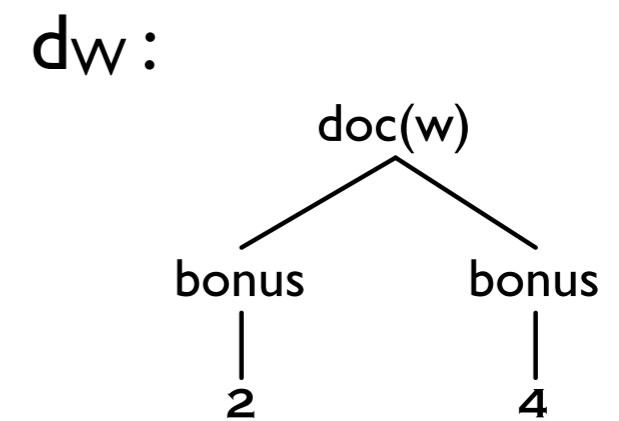
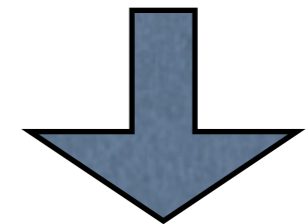
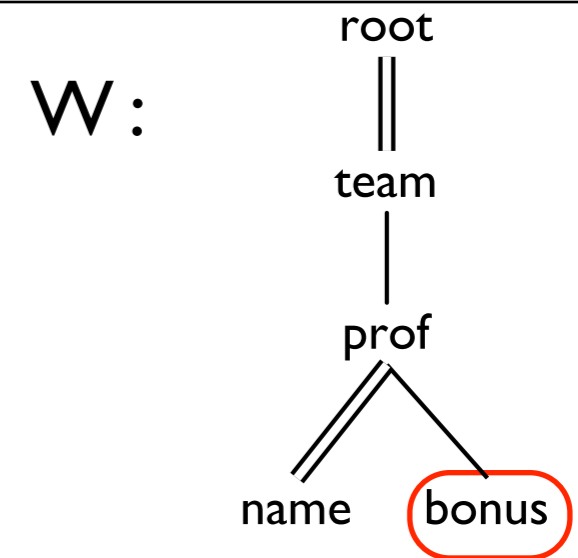
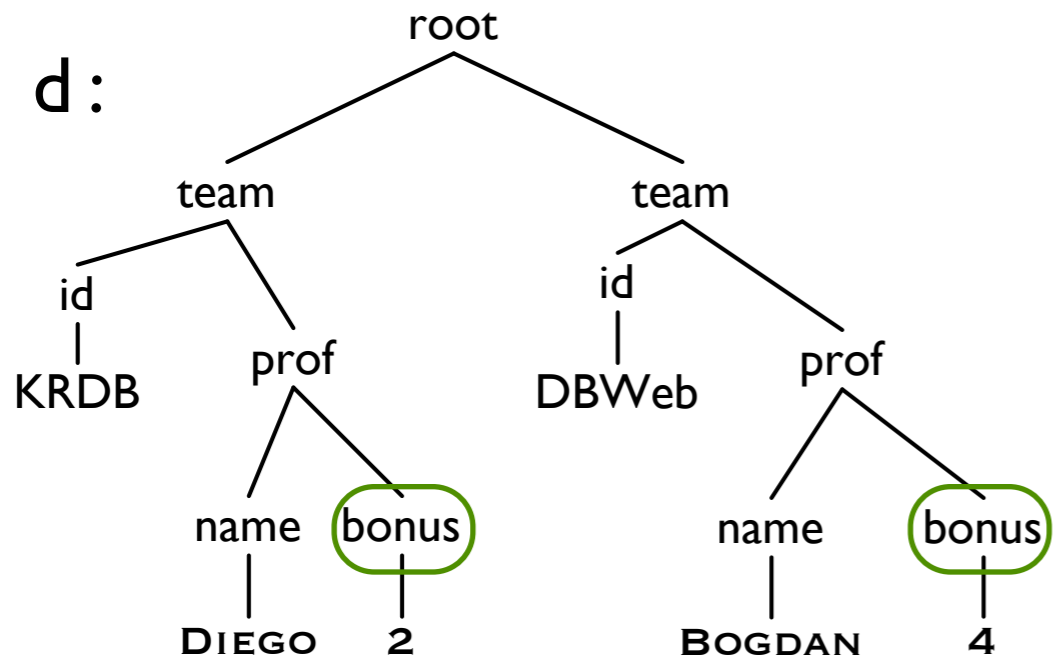
Views over Documents



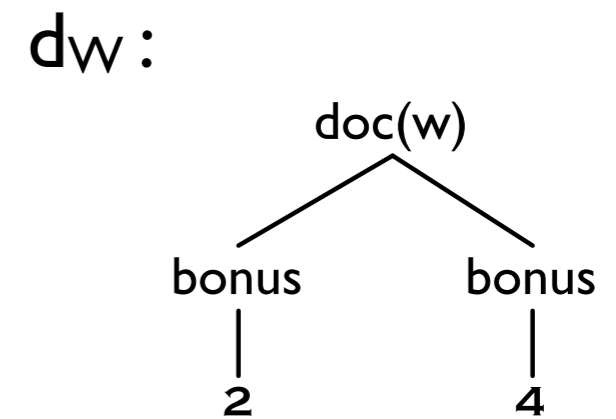
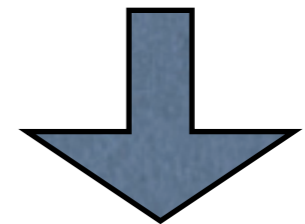
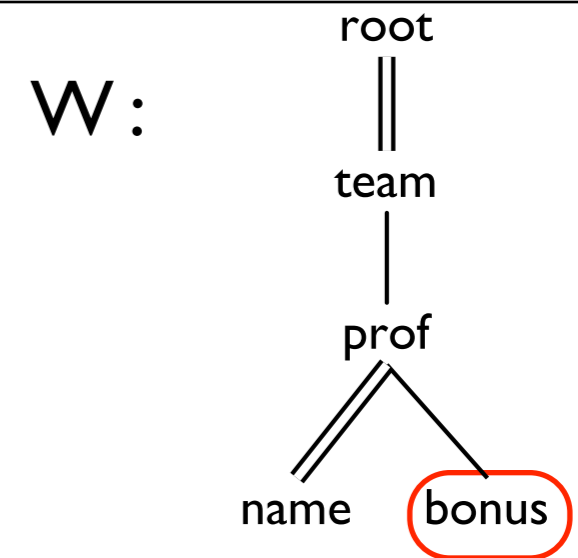
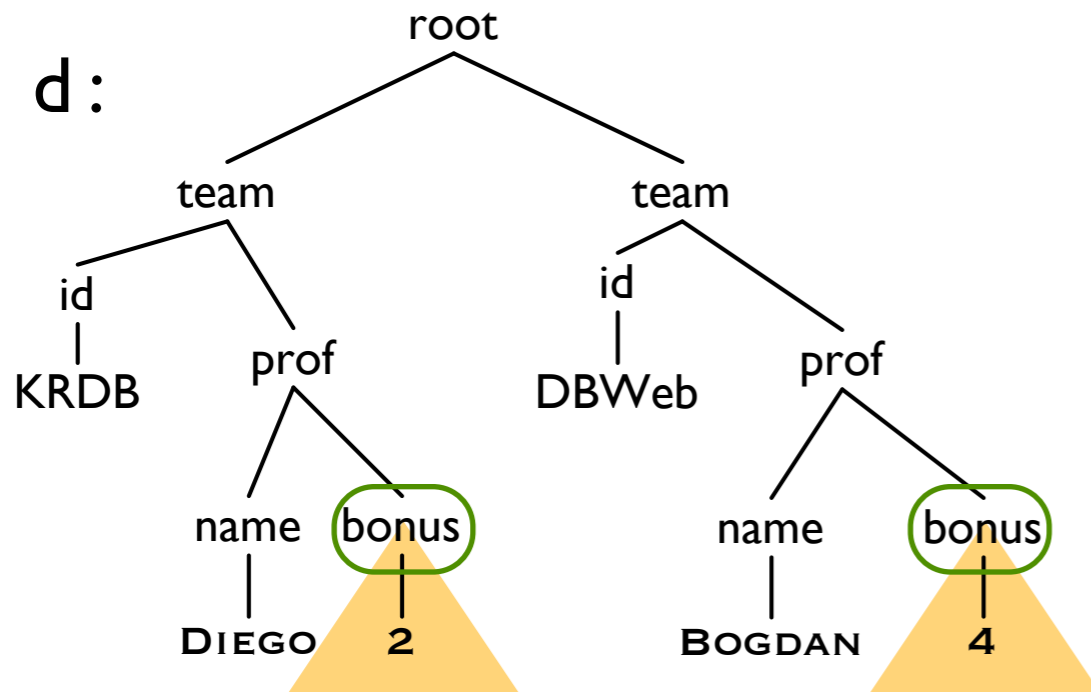
Views over Documents



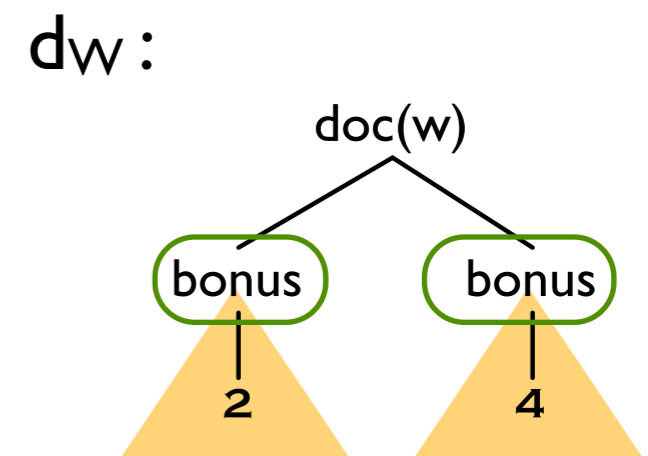
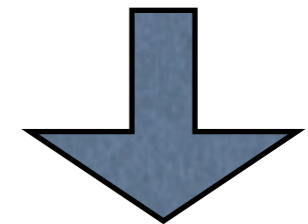
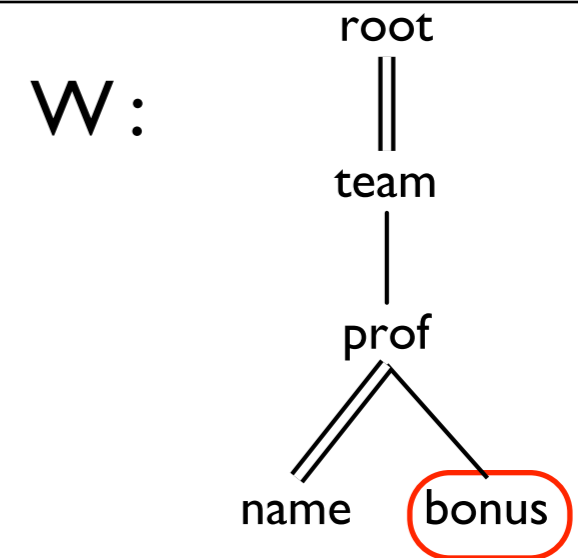
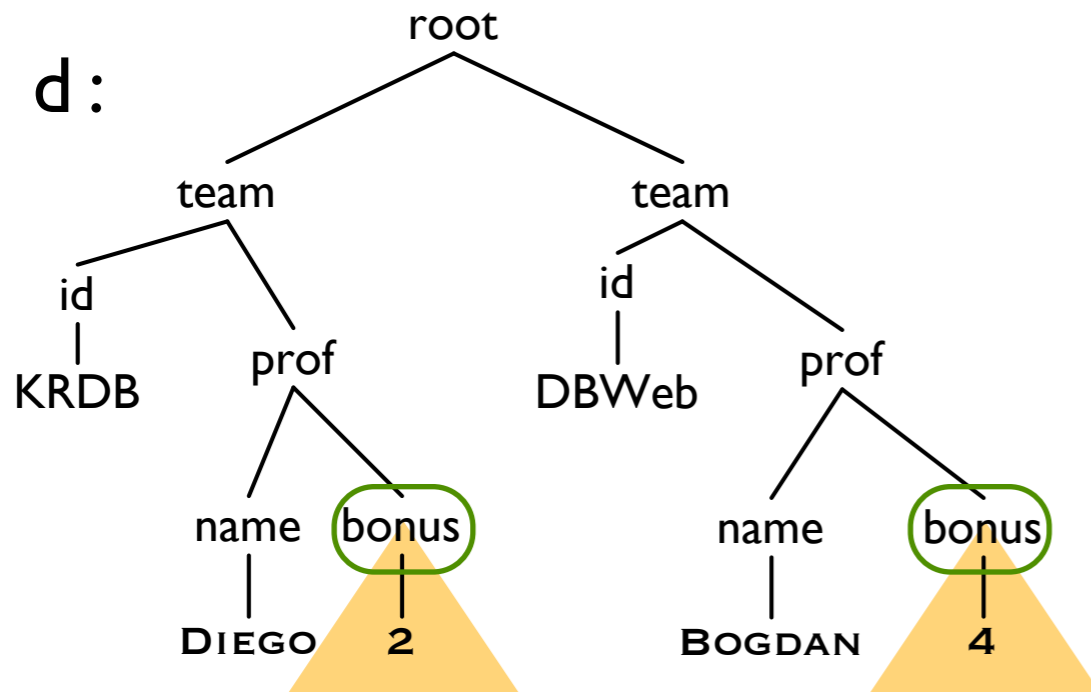
Views over Documents



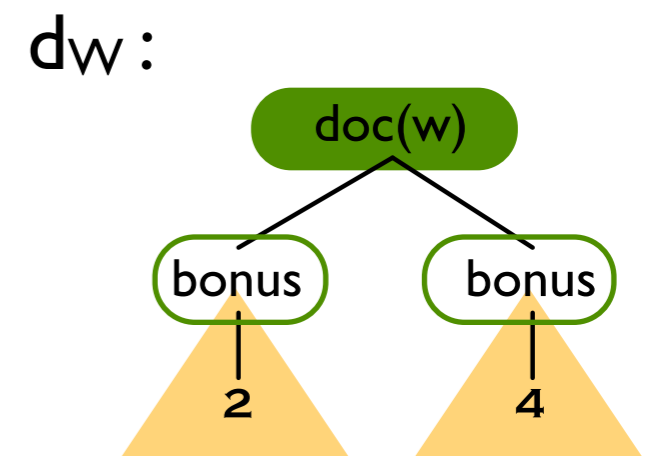
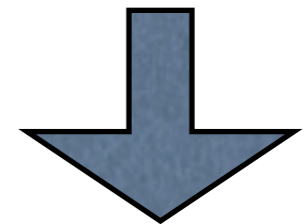
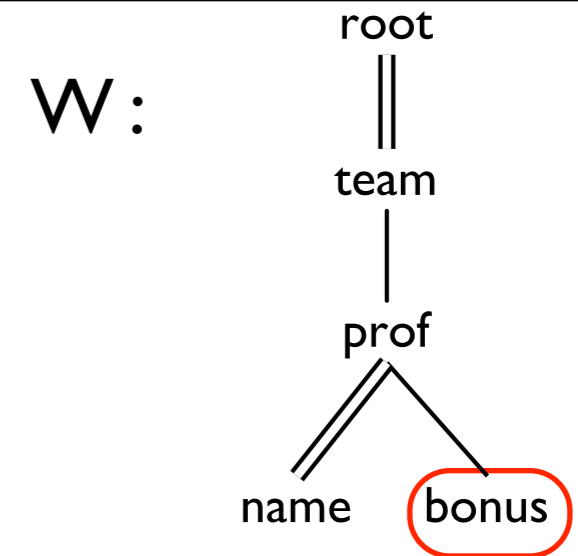
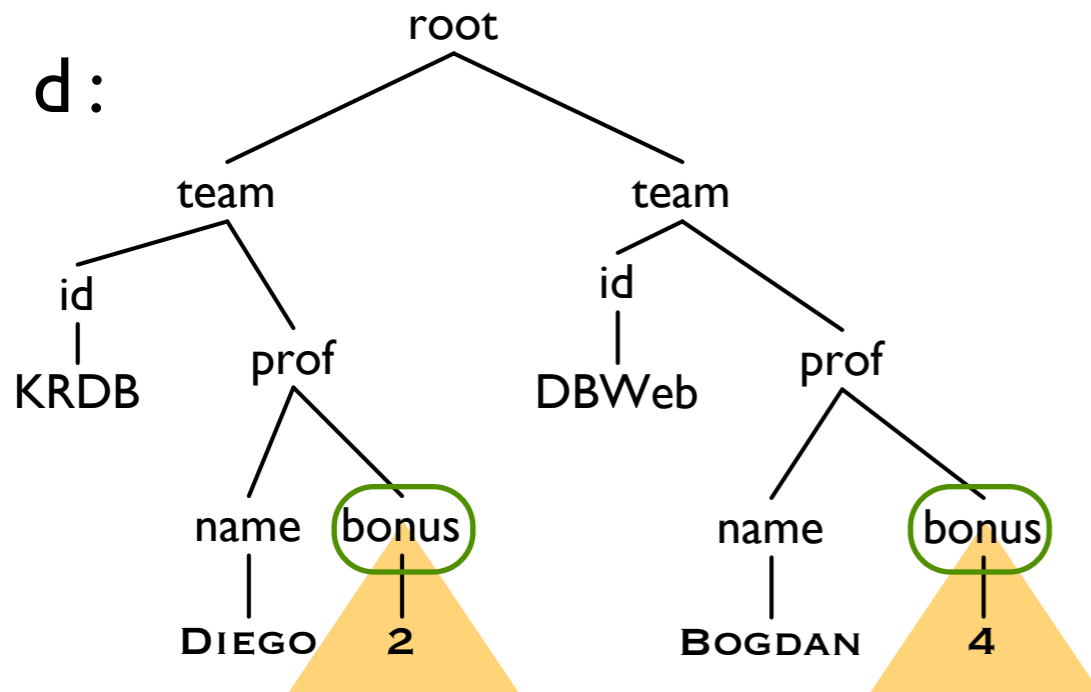
Views over Documents



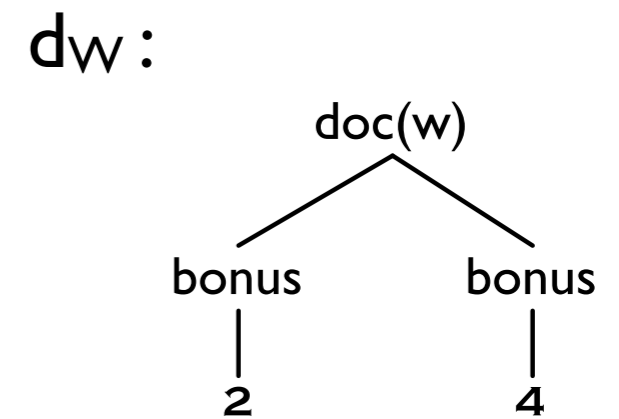
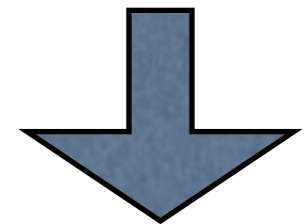
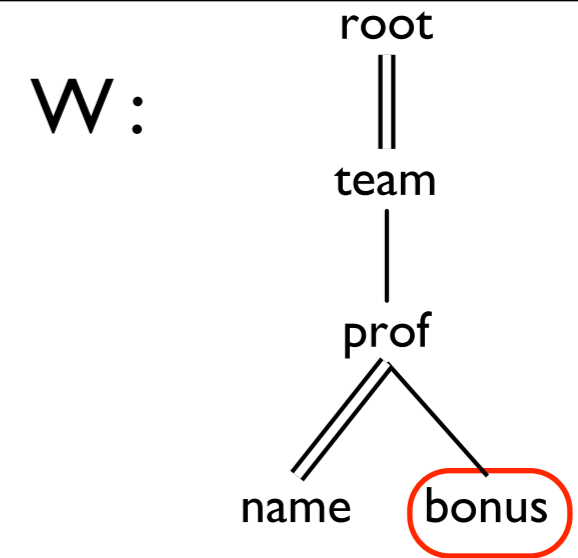
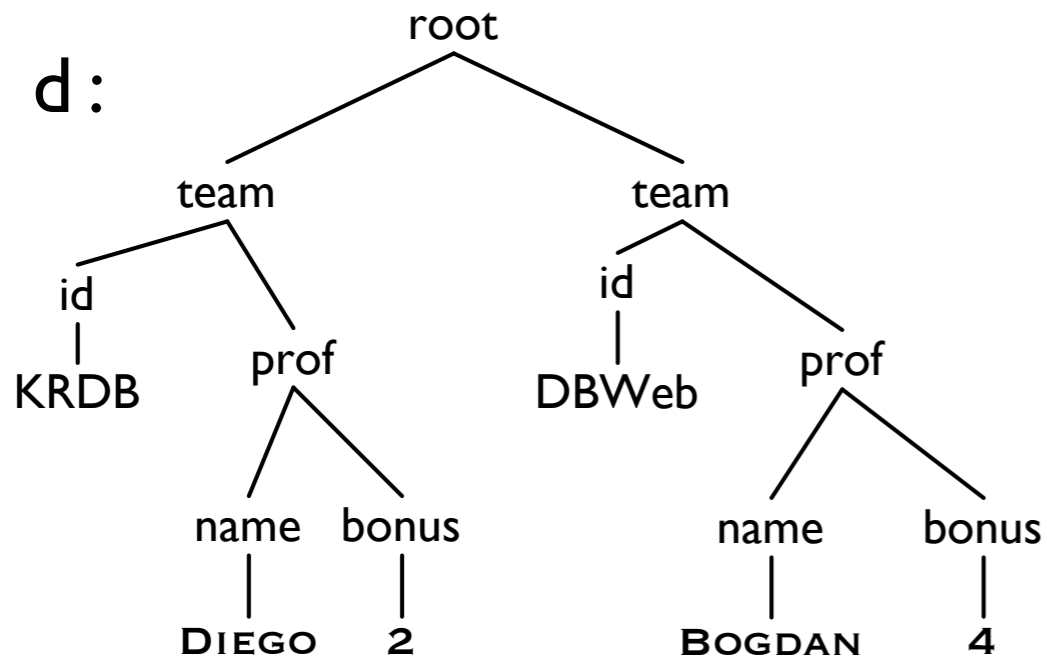
Views over Documents



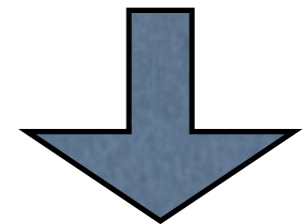
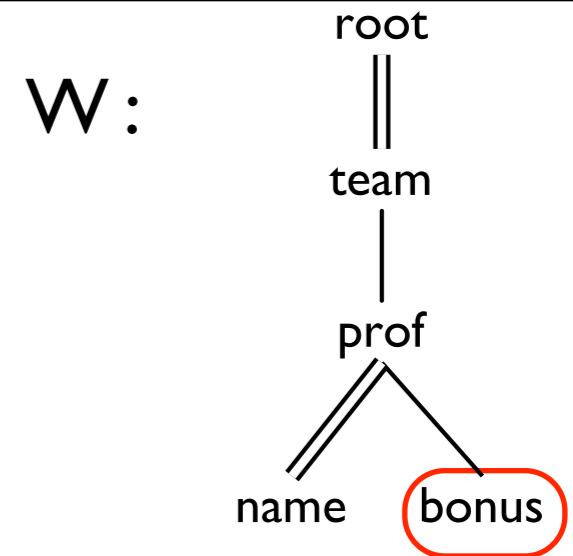
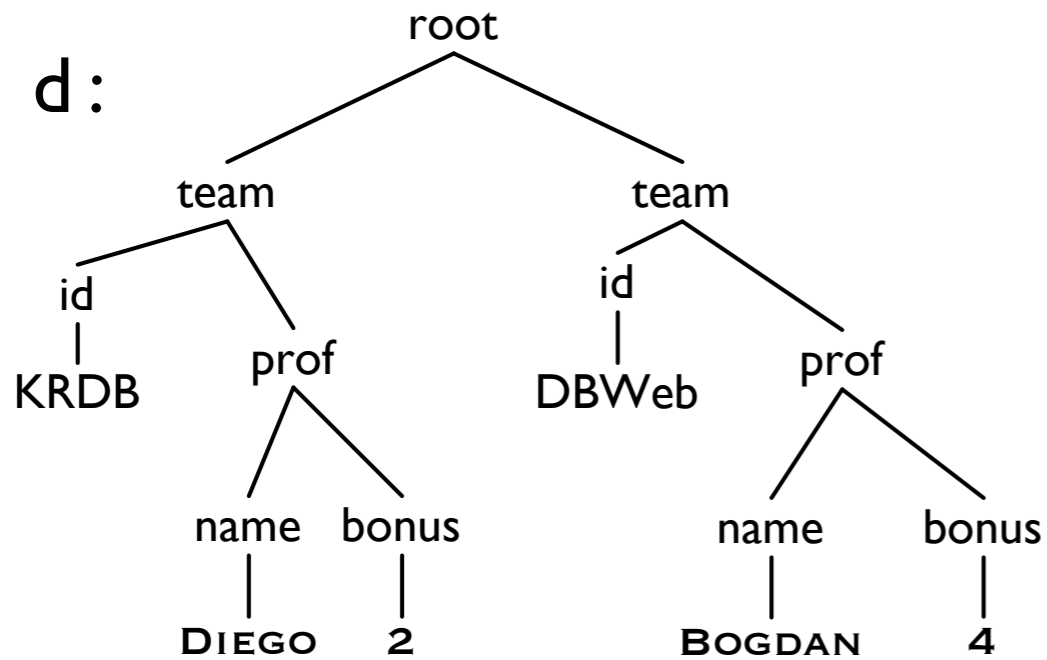
Views over Documents



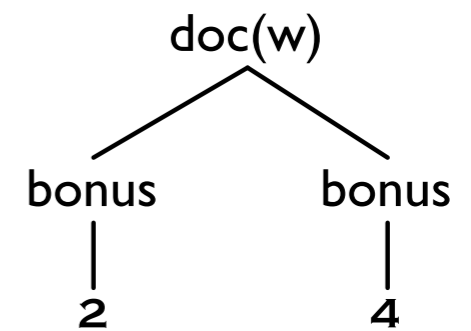
Views over Documents



Views over Documents

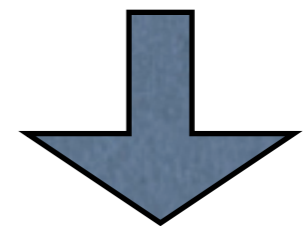
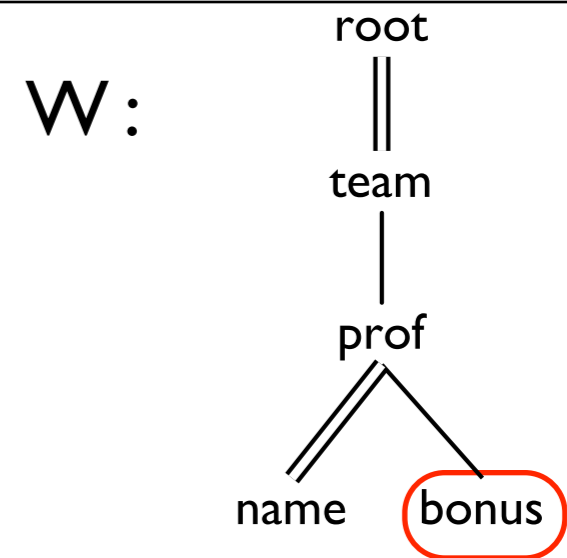
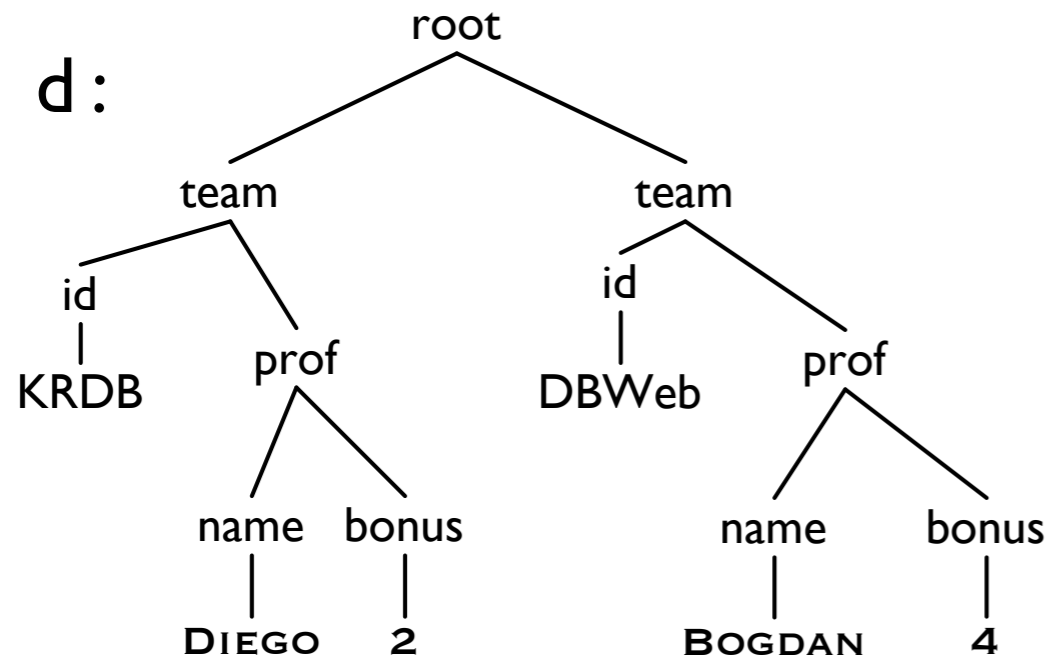


dw:

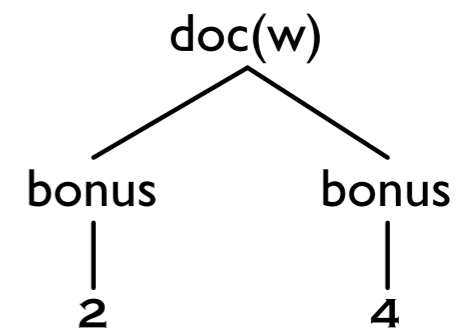


A **view** v over a doc d is a **document** d_v composed from subdocuments of d

Views over Documents



dw:

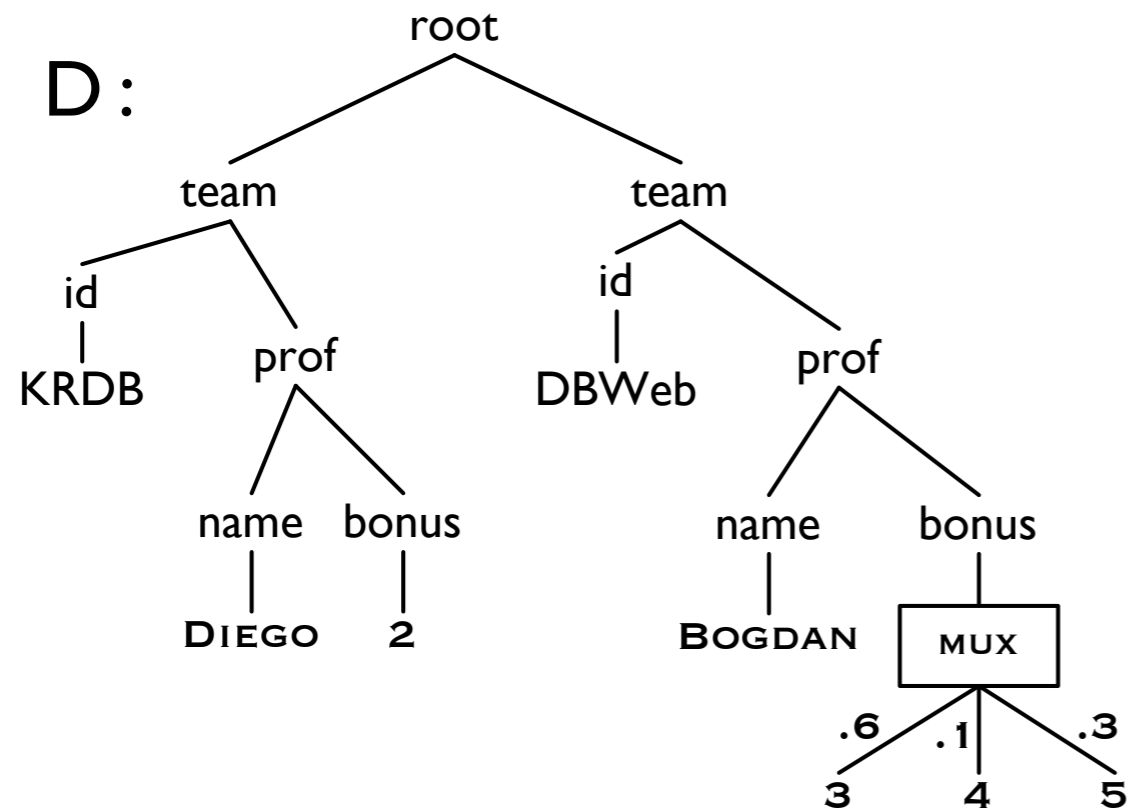


A **view** v over a doc d is a **document** d_v composed from subdocuments of d

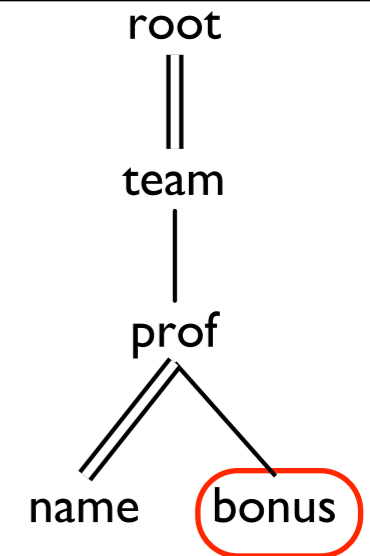
Views can either

- export **original** doc **Ids**
- introduce **fresh** **Ids**

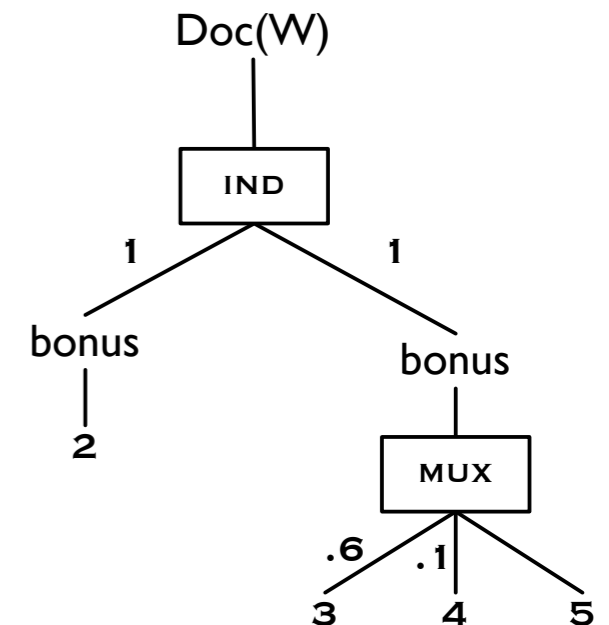
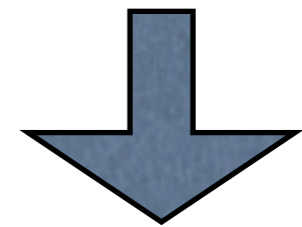
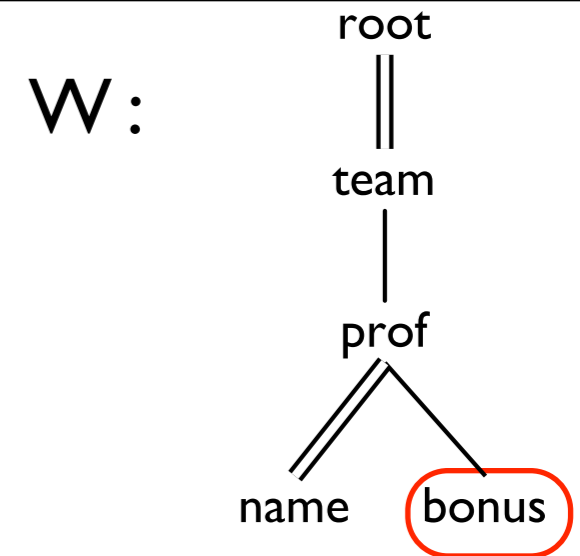
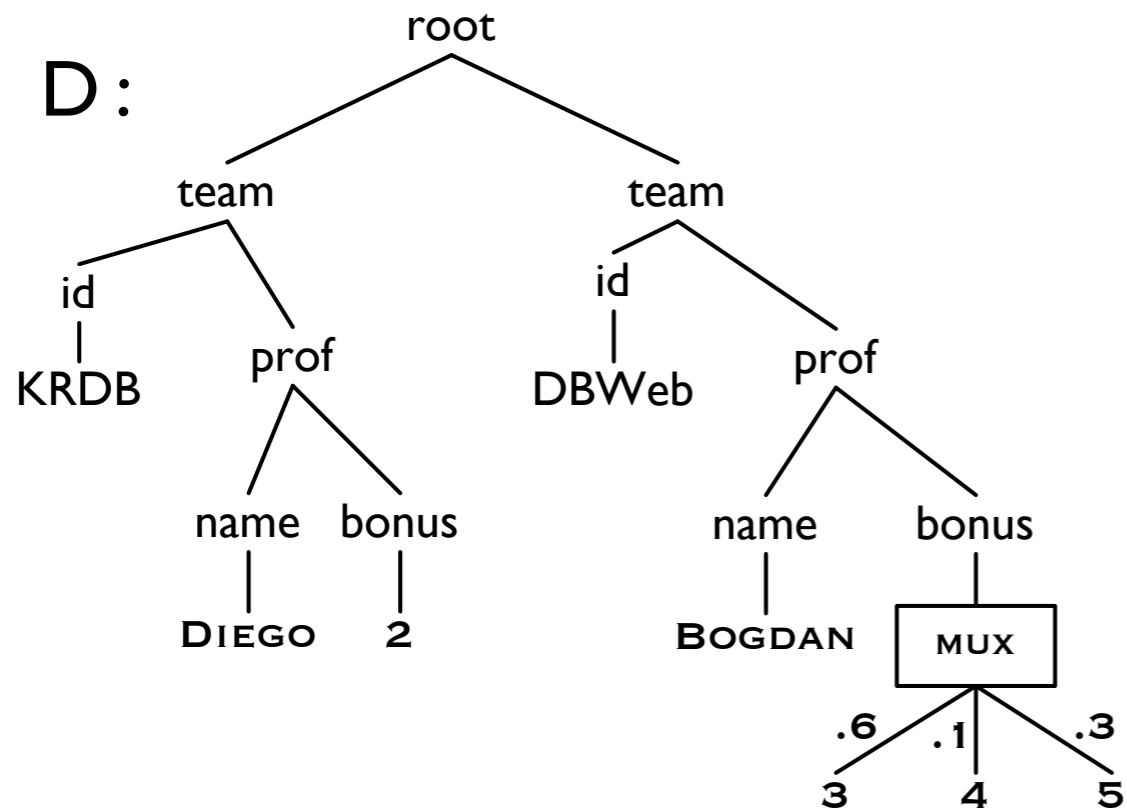
Views over Probabilistic Documents



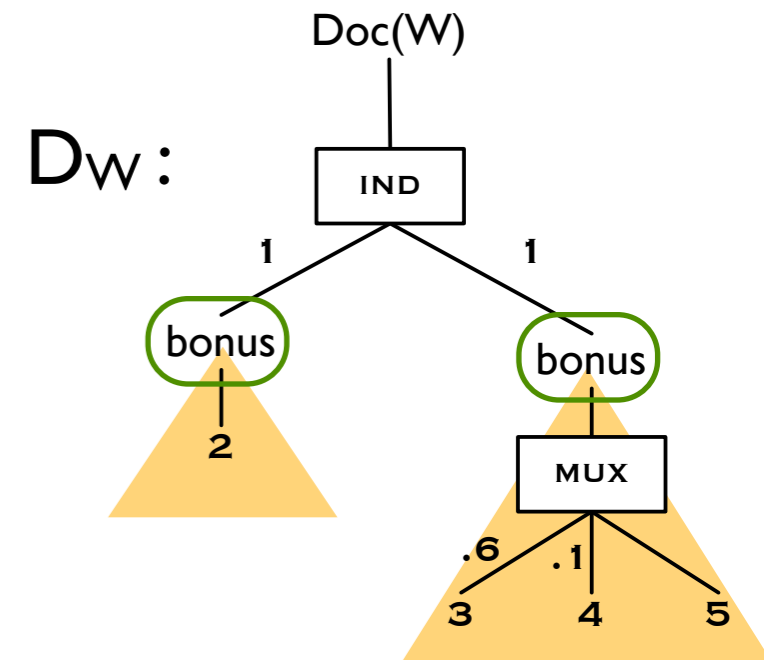
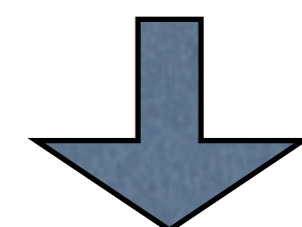
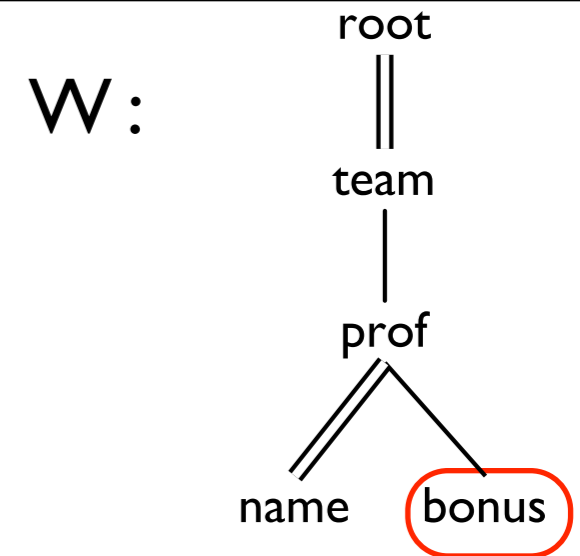
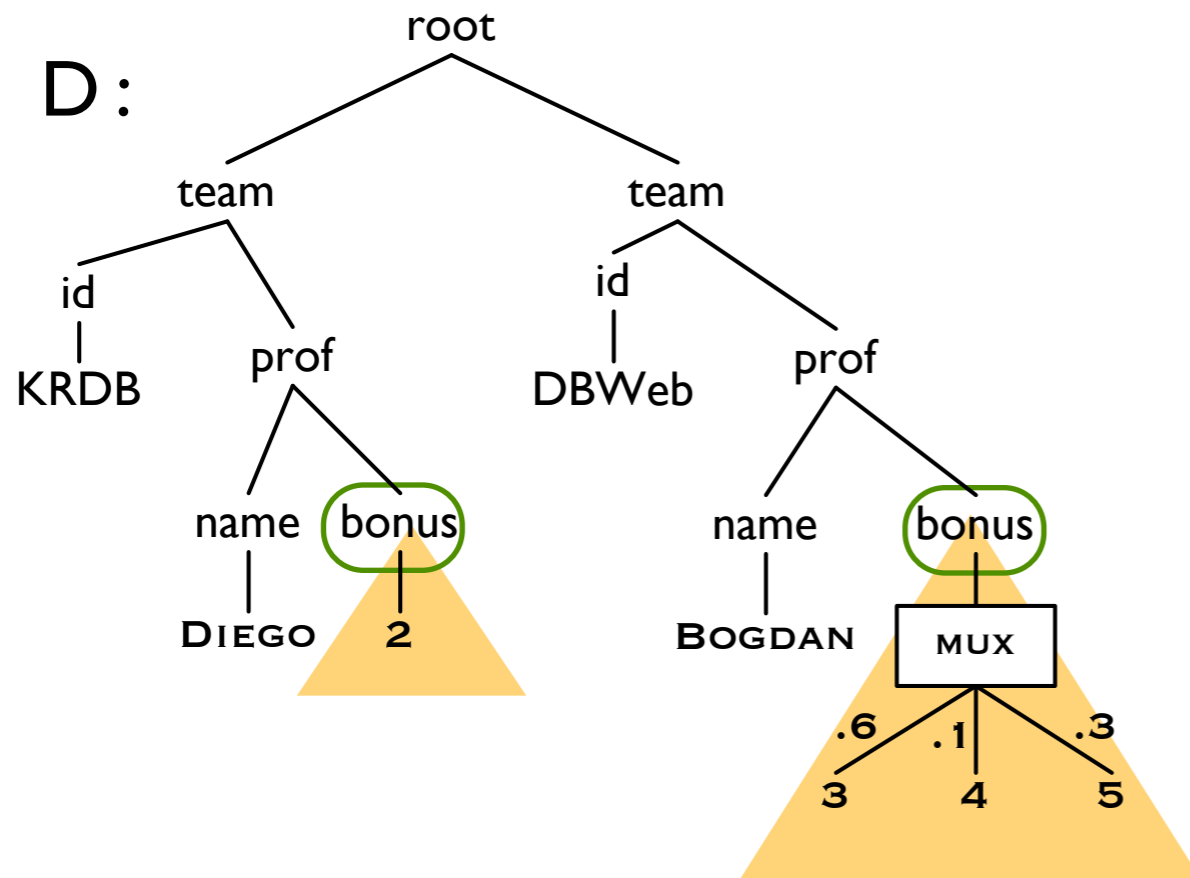
W:



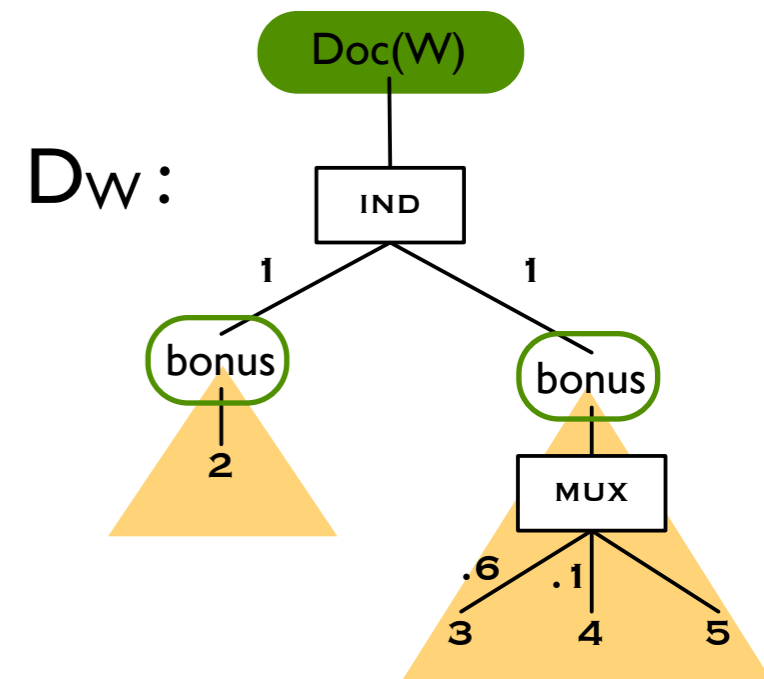
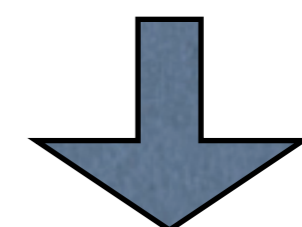
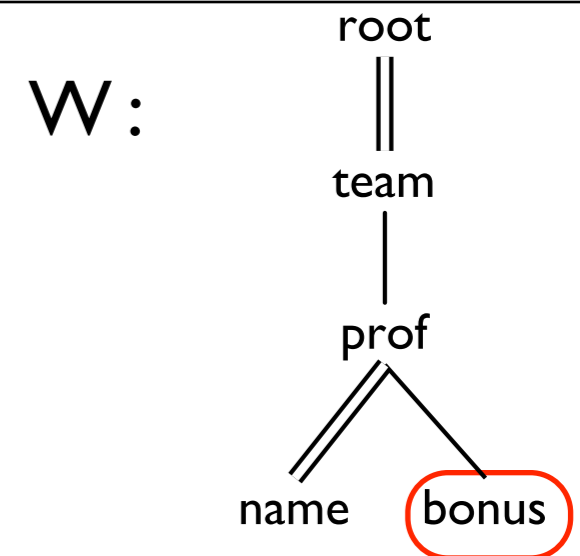
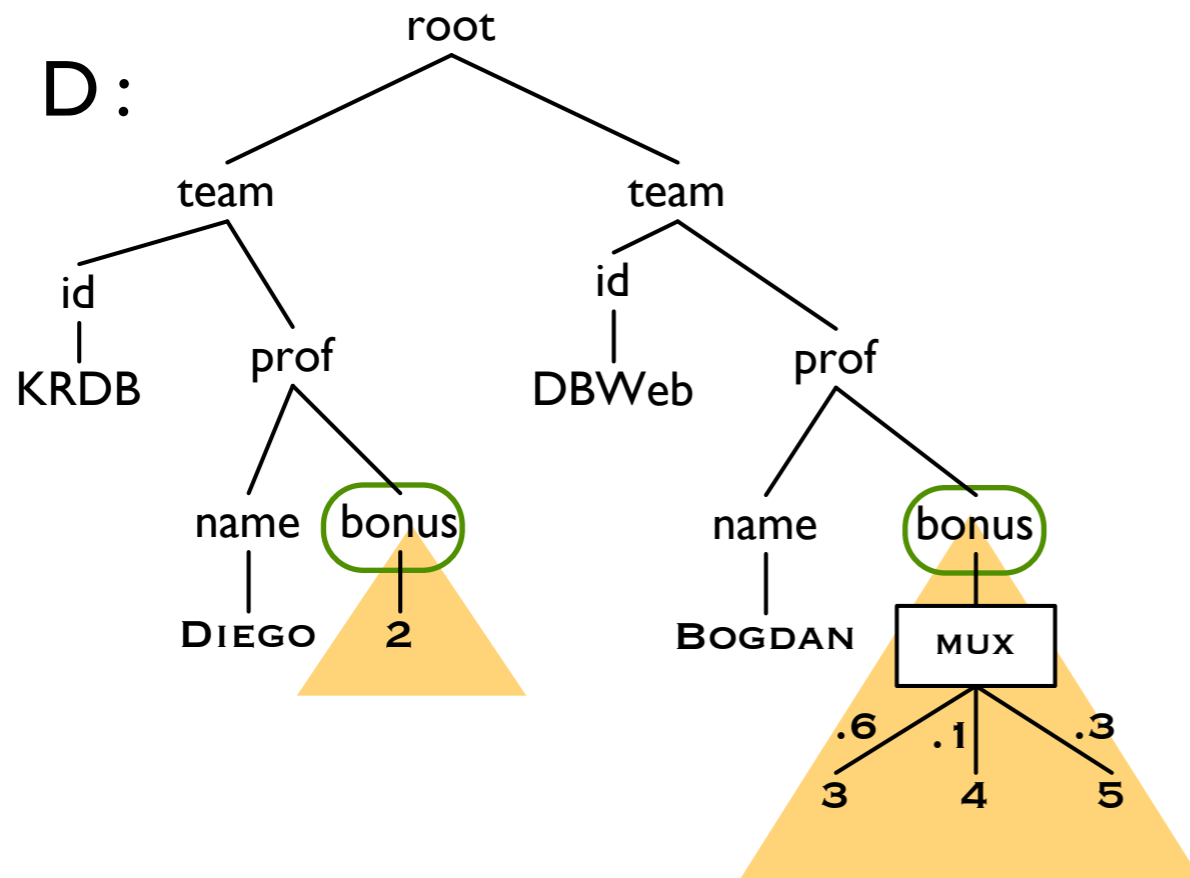
Views over Probabilistic Documents



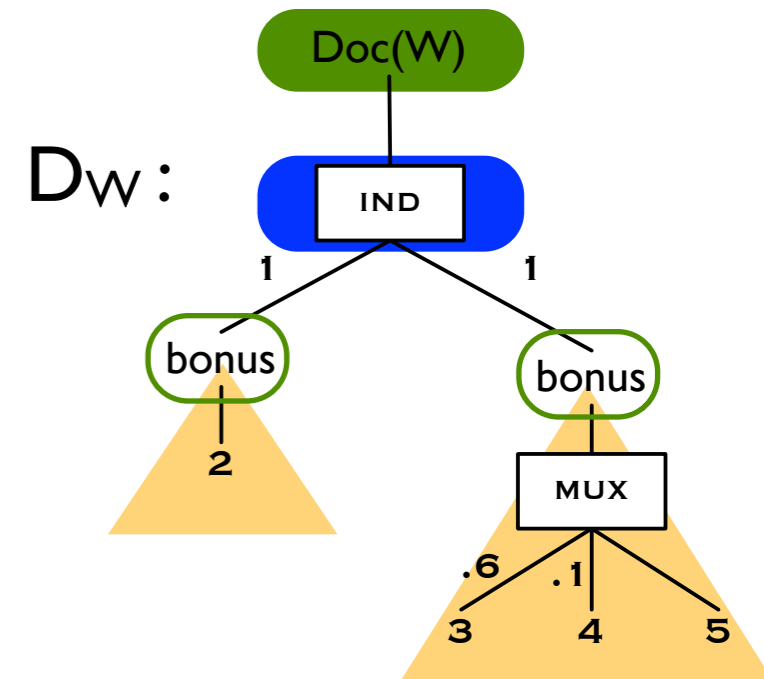
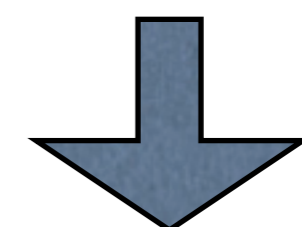
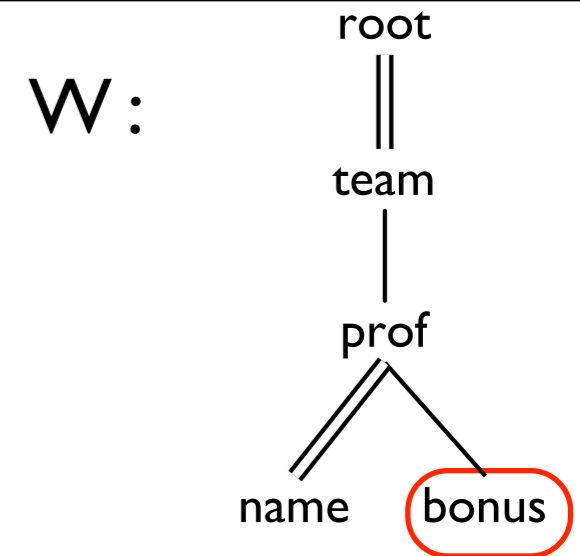
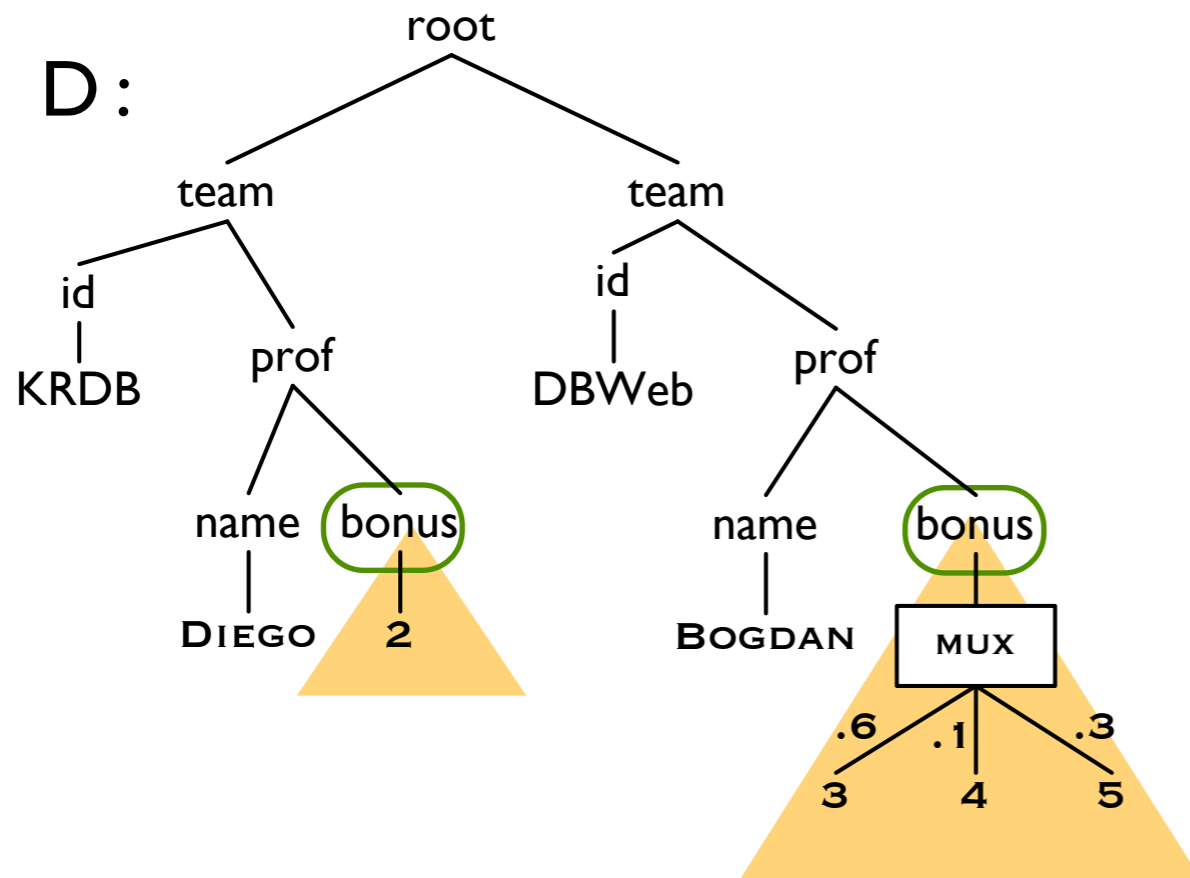
Views over Probabilistic Documents



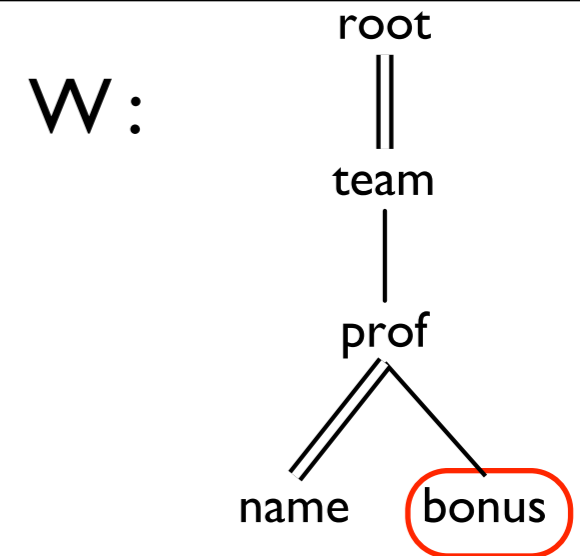
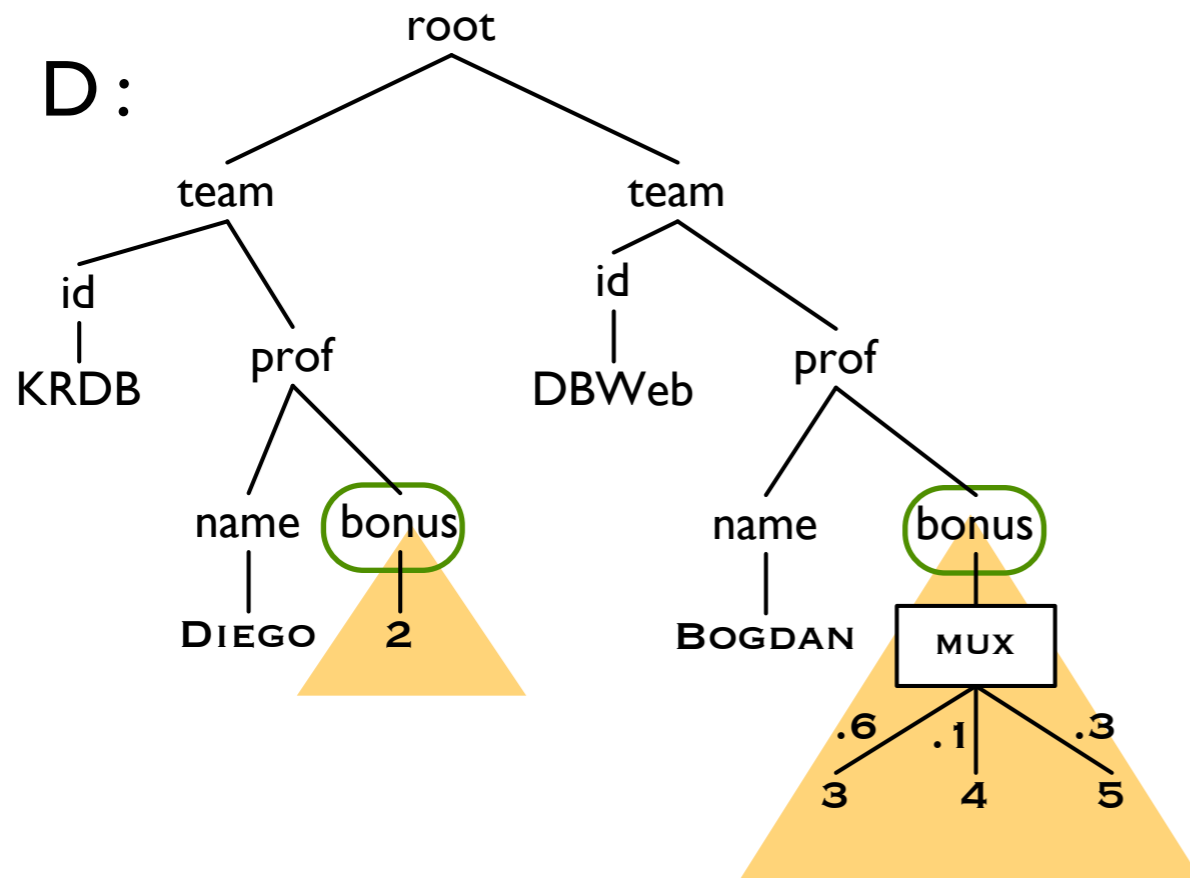
Views over Probabilistic Documents



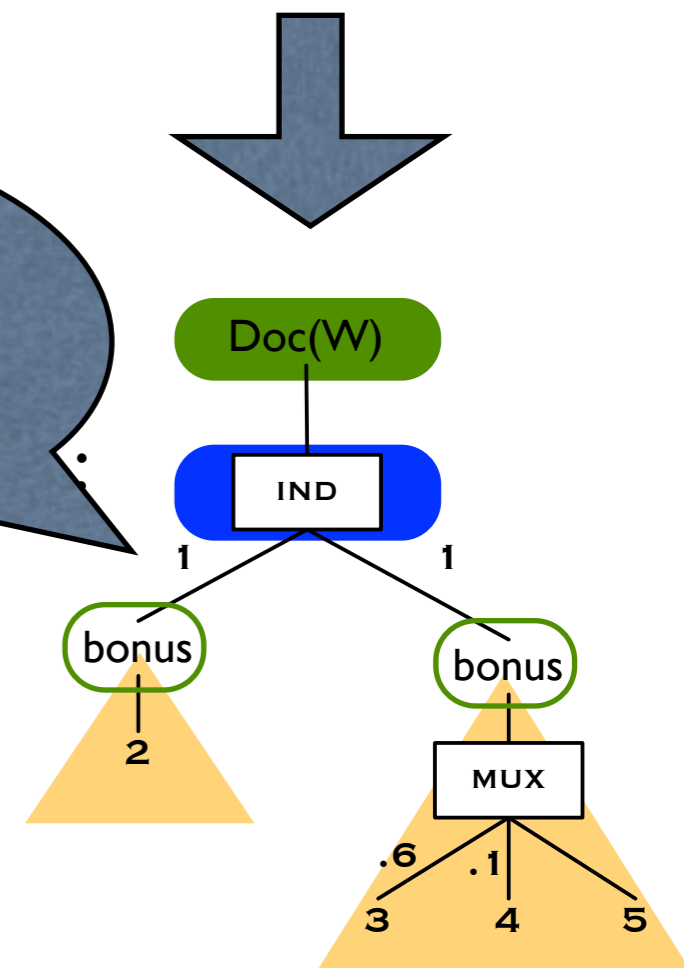
Views over Probabilistic Documents



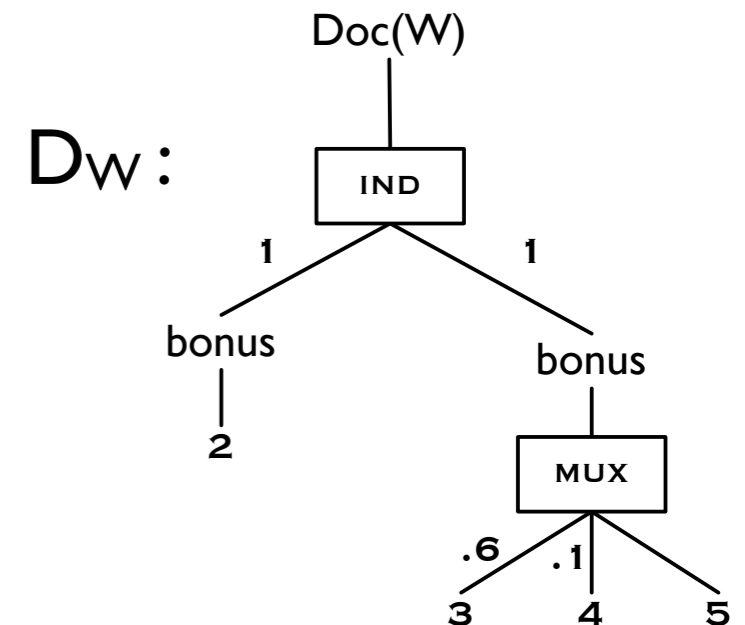
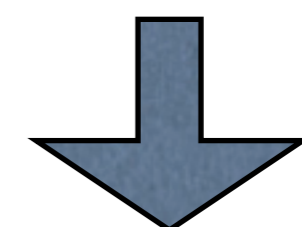
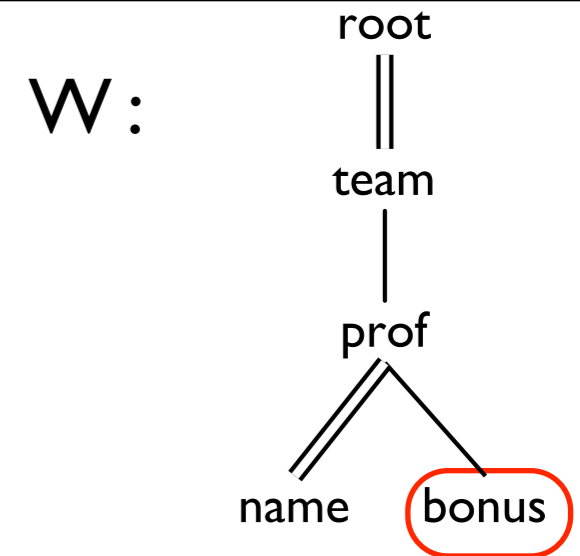
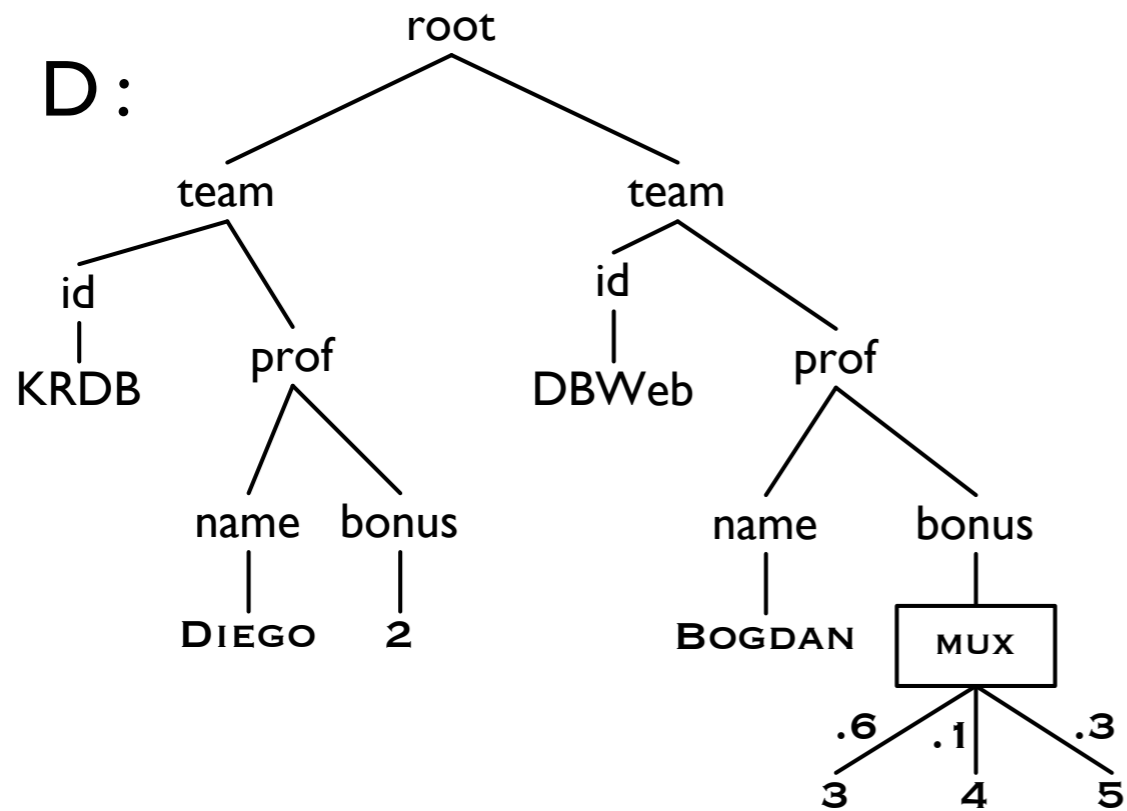
Views over Probabilistic Documents



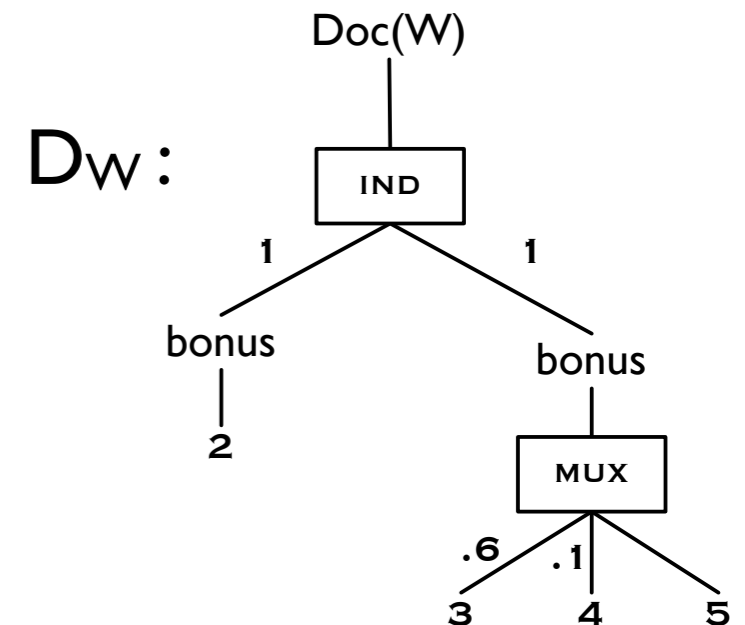
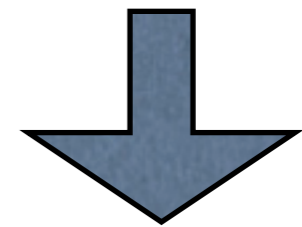
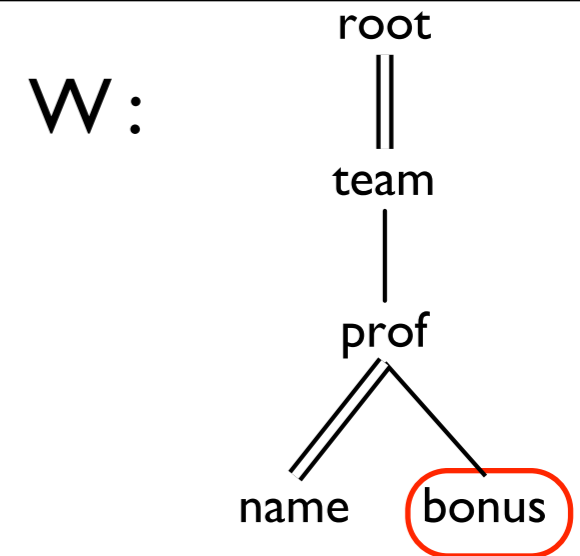
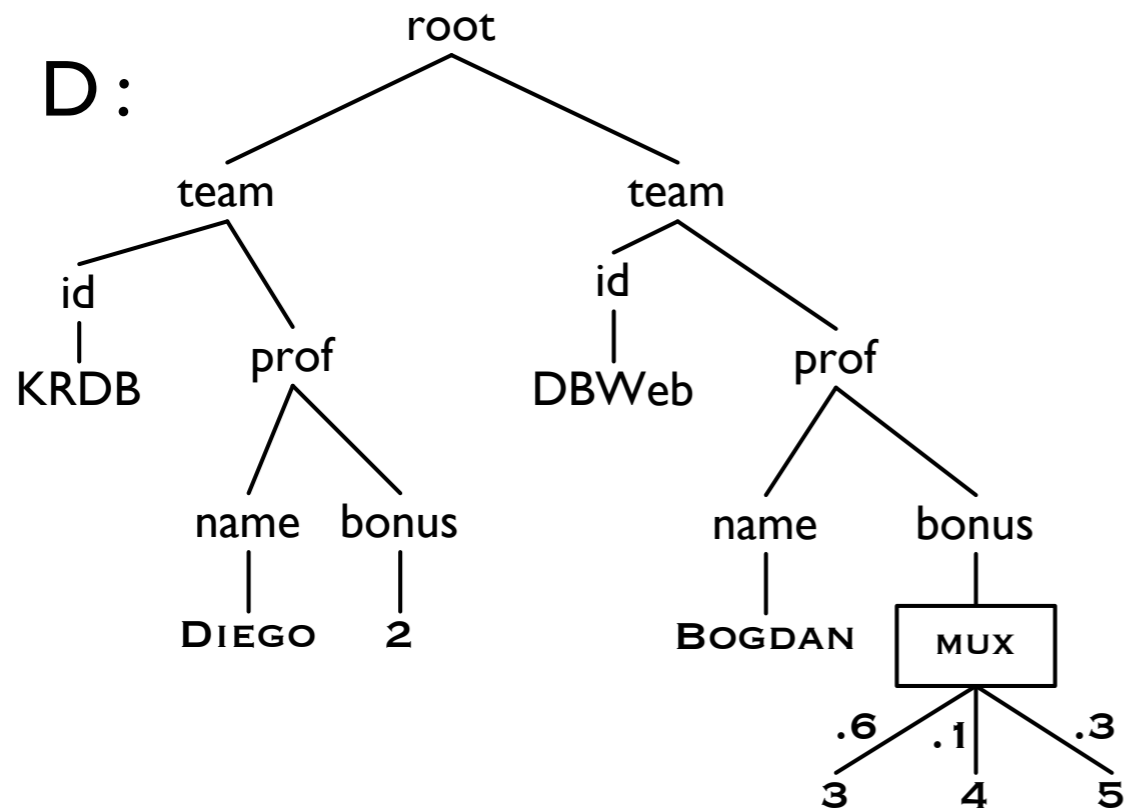
$P(\text{bonus} \in W(D))$
It is < 1 in general



Views over Probabilistic Documents

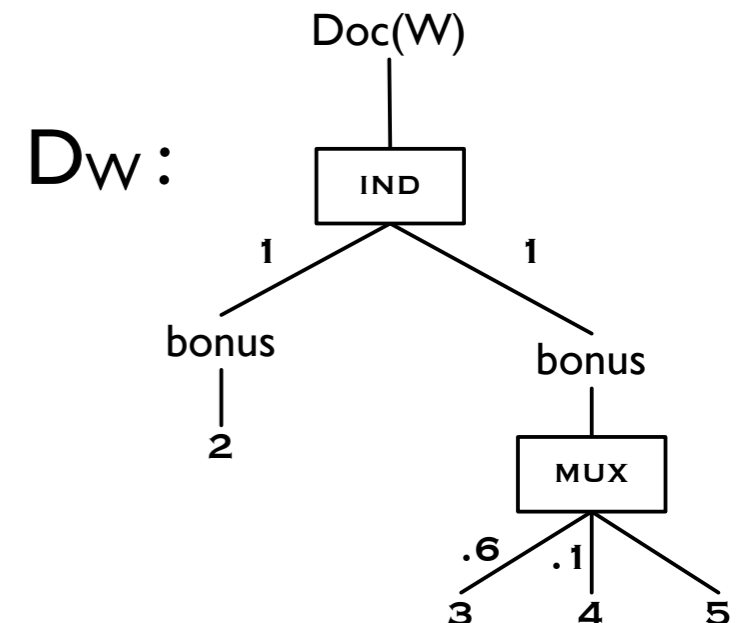
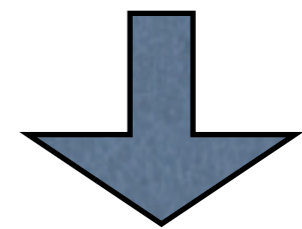
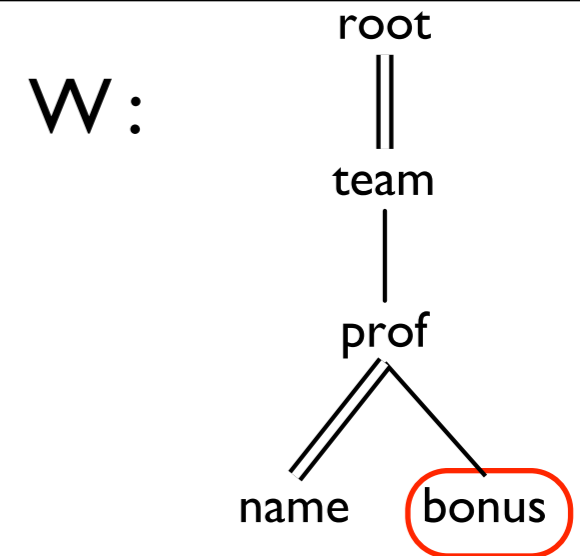
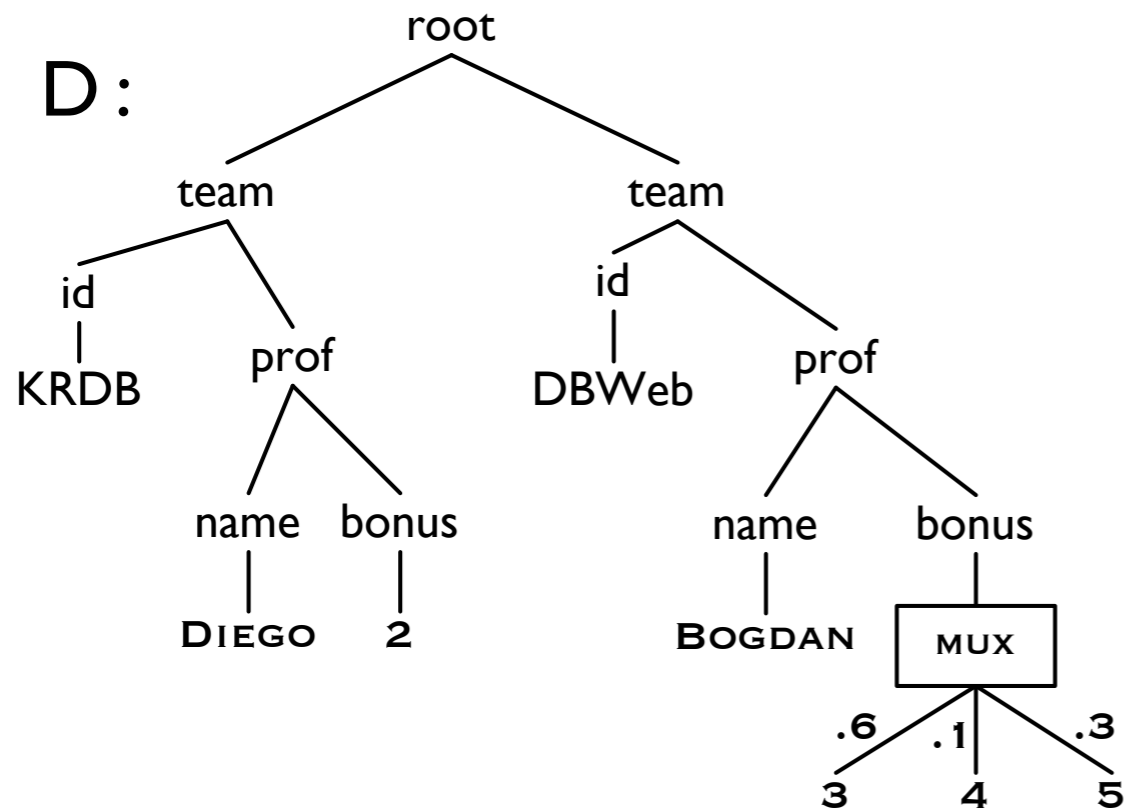


Views over Probabilistic Documents



A **view** v over a p-doc D :
is a **p-doc** d_v
composed
from p-subdocuments of D

Views over Probabilistic Documents



A **view** v over a p-doc D :
is a **p-doc** d_v
composed
from p-subdocuments of D

Views can either

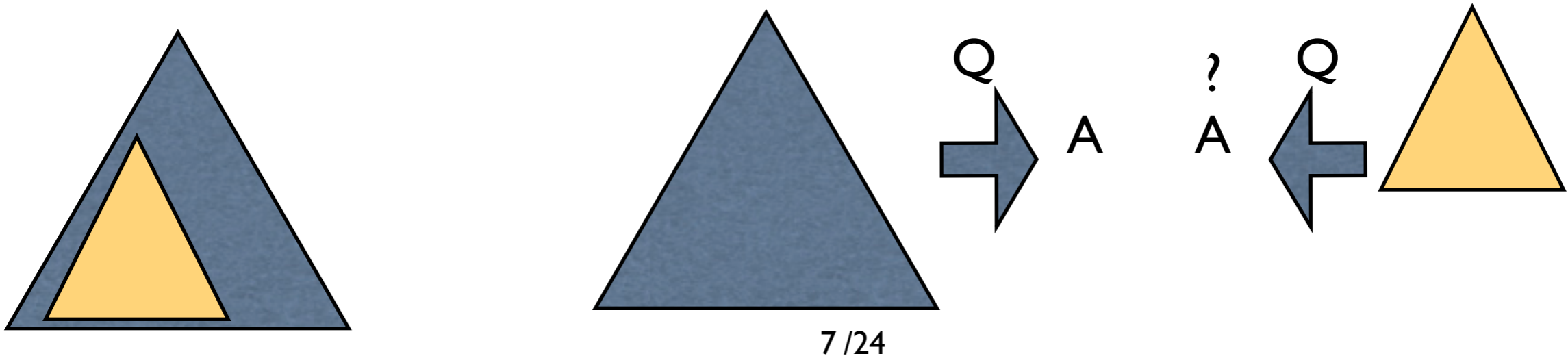
- export **original** doc **Ids**
- introduce **fresh** **Ids**

Why views are important?

- To improve/facilitate query answering:
 - Can we reuse previously computed answers to improve/facilitate query answering?
- To deal with access limitation:
 - Is there a chance to compute correct answers and probabilities if one sees only a fragment of a probabilistic document?

Why views are important?

- To improve/facilitate query answering:
 - Can we reuse previously computed answers to improve/facilitate query answering?
- To deal with access limitation:
 - Is there a chance to compute correct answers and probabilities if one sees only a fragment of a probabilistic document?



View-based Query Answering

[Cautis&al'11]

- Given a query Q and views V, W, \dots
- **Deterministic rewriting** problem:
find a query $R \sim$ **rewriting** of Q with V, W, \dots s.t.

$$Q(d) = R(d_v, d_w, \dots)$$

View-based Query Answering

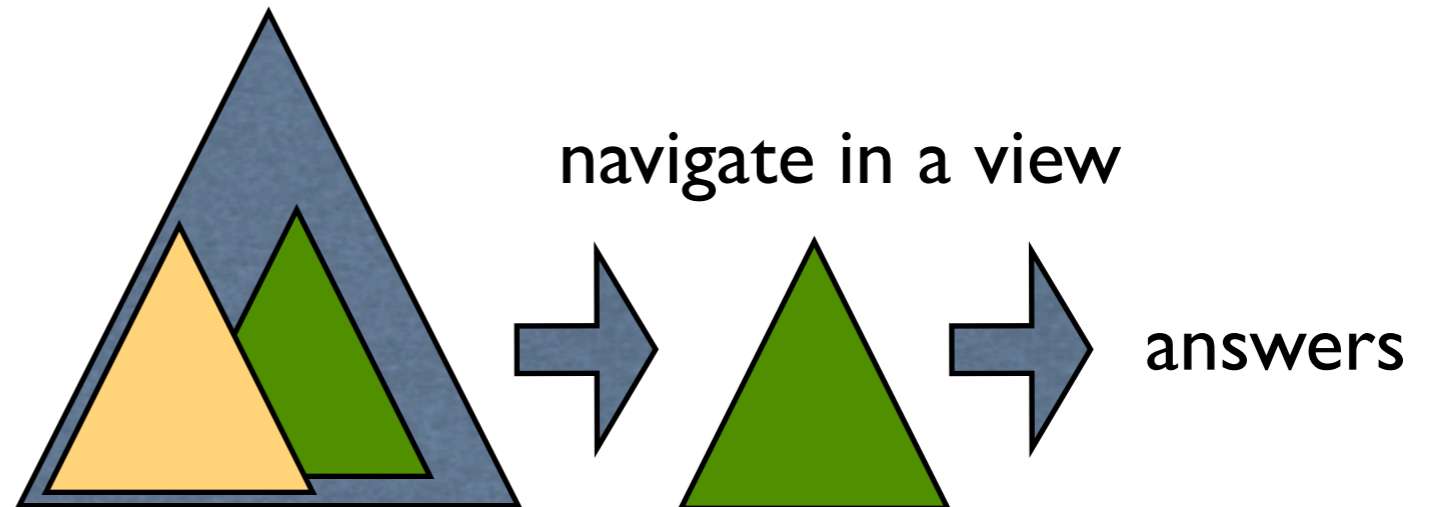
[Cautis&al'11]

- Given a query Q and views V, W, \dots
- **Deterministic rewriting** problem:
find a query $R \sim$ **rewriting** of Q with V, W, \dots s.t.

$$Q(d) = R(d_v, d_w, \dots)$$

- Two approaches:

- I. Views have fresh Ids:
rewriting by **compensation**



View-based Query Answering

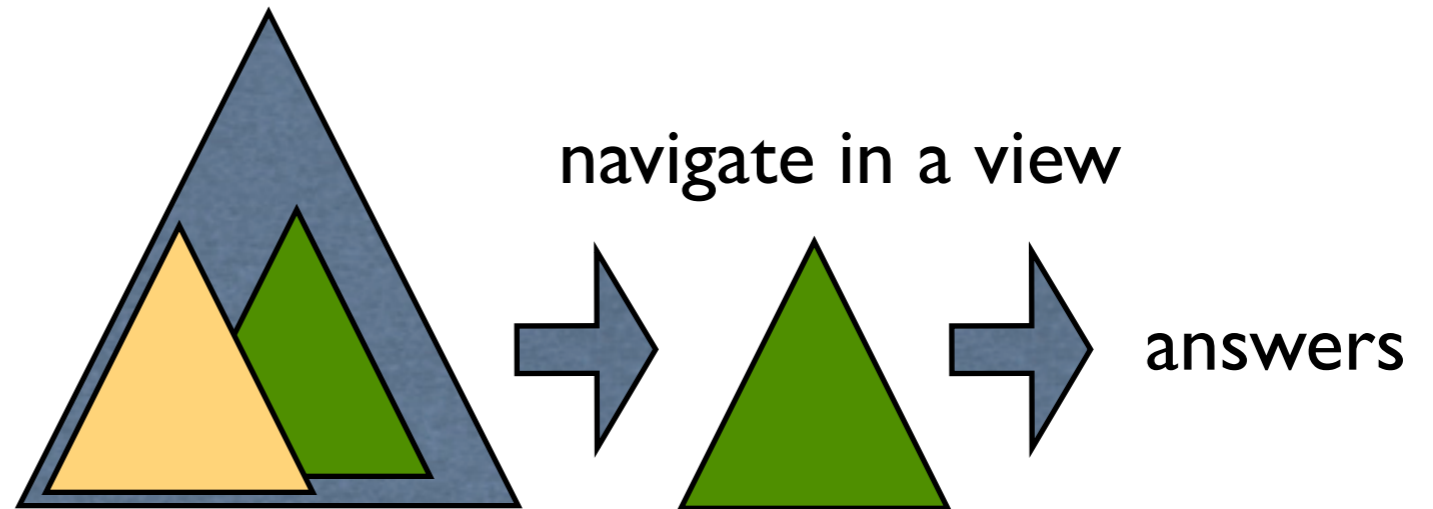
[Cautis&al'11]

- Given a query Q and views V, W, \dots
- Deterministic rewriting** problem:
find a query $R \sim$ **rewriting** of Q with V, W, \dots s.t.

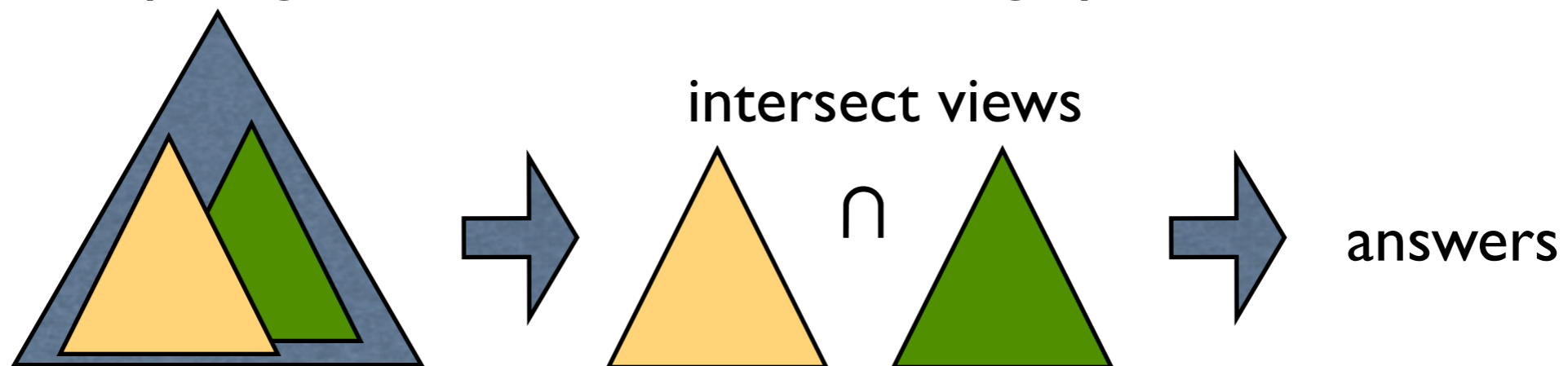
$$Q(d) = R(d_v, d_w, \dots)$$

- Two approaches:

- Views have fresh Ids:
rewriting by **compensation**



- Views keep original document Ids: rewriting by **intersection**



Probabilistic View-based Query Answering

- Over probabilistic XML: (R, F)
 - I. find a **rewriting** R :
retrieves answers without probabilities

$$\begin{aligned} & \Pr(a \in R(D_v, D_w, \dots)) > 0 \\ \text{iff } & \Pr(a \in Q(D)) > 0 \end{aligned}$$

Probabilistic View-based Query Answering

- Over probabilistic XML: (R, F)
 1. find a **rewriting R**:
retrieves answers without probabilities
 2. find a **probability function F**:
computes probabilities of answers

$$\Pr(a \in R(D_v, D_w, \dots)) > 0$$
$$\text{iff } \Pr(a \in Q(D)) > 0$$

$$F(a, D_v, D_w, \dots) = \Pr(a \in Q(D))$$

Probabilistic View-based Query Answering

- Over probabilistic XML: (R, F)

1. find a **rewriting R**:
retrieves answers without probabilities

$$\Pr(a \in R(D_v, D_w, \dots)) > 0 \\ \text{iff } \Pr(a \in Q(D)) > 0$$

2. find a **probability function F**:
computes probabilities of answers

$$F(a, D_v, D_w, \dots) = \Pr(a \in Q(D))$$

- **Proposition**

If R is a deterministic rewriting \Rightarrow it retrieves required answers

Probabilistic View-based Query Answering

- Over probabilistic XML: (R, F)

1. find a **rewriting** R :
retrieves answers without probabilities

$$\Pr(a \in R(D_v, D_w, \dots)) > 0 \\ \text{iff } \Pr(a \in Q(D)) > 0$$

2. find a **probability function** F :
computes probabilities of answers

$$F(a, D_v, D_w, \dots) = \Pr(a \in Q(D))$$

- **Proposition**

If R is a deterministic rewriting \Rightarrow it retrieves required answers

- What to study then?

Properties of F

- Does F **always exist** whenever R does?
- Are there settings where computing answers with R and F over D_v, D_w, \dots is **easier than** with Q over D ?

Outline

- Rewriting over probabilistic XML
- Rewriting by compensation
- Rewriting by intersection

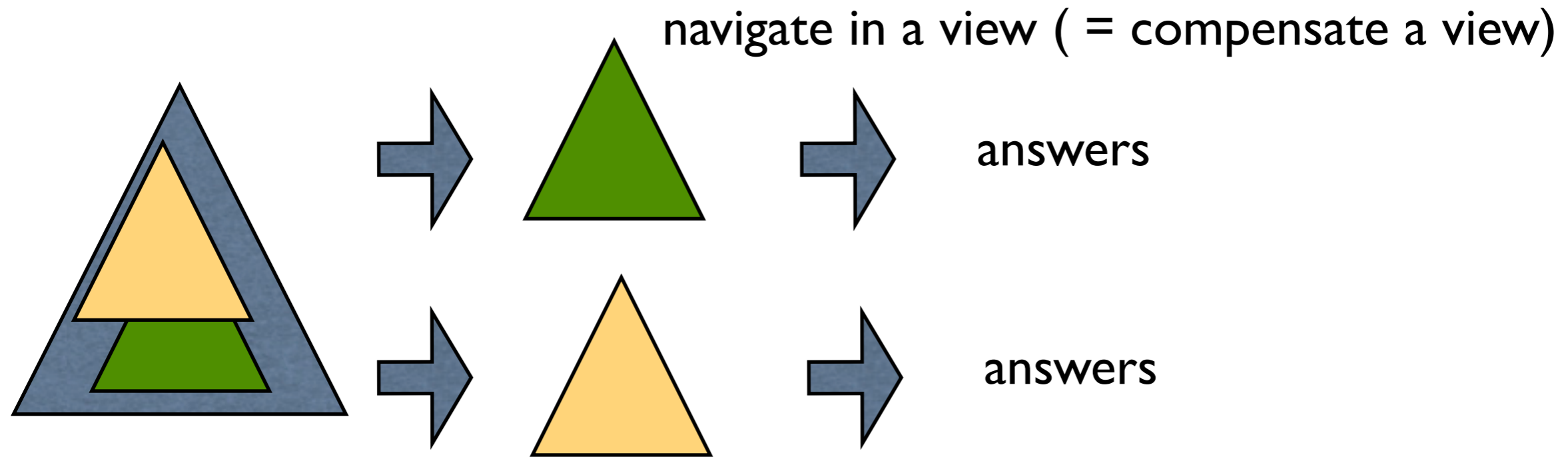
Outline

- Rewriting over probabilistic XML
- Rewriting by compensation
- Rewriting by intersection

Outline

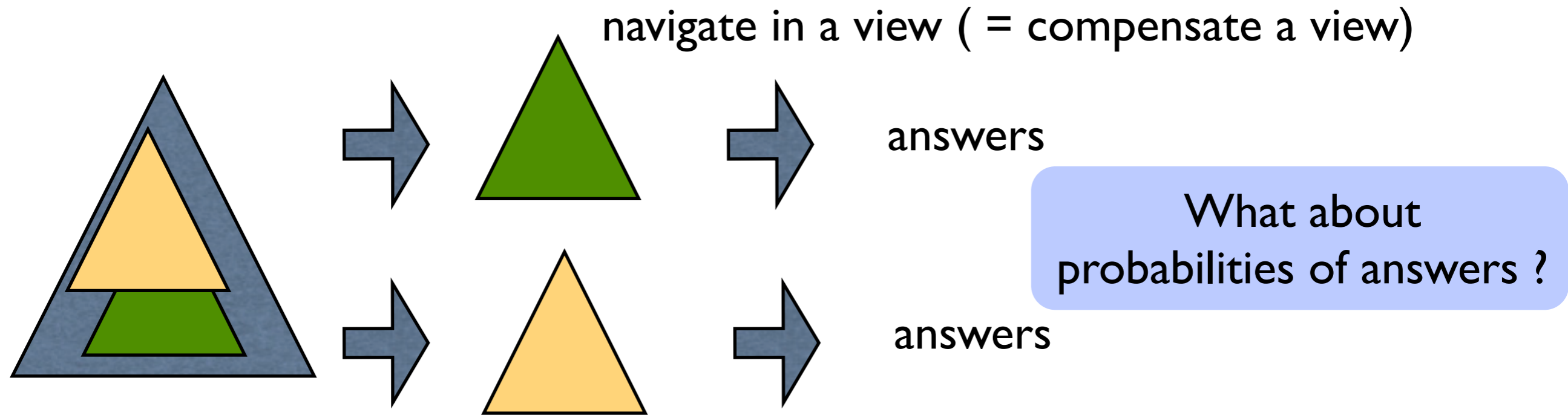
- Rewriting over probabilistic XML
- Rewriting by compensation
- Rewriting by intersection

Rewriting by Compensation: General Idea



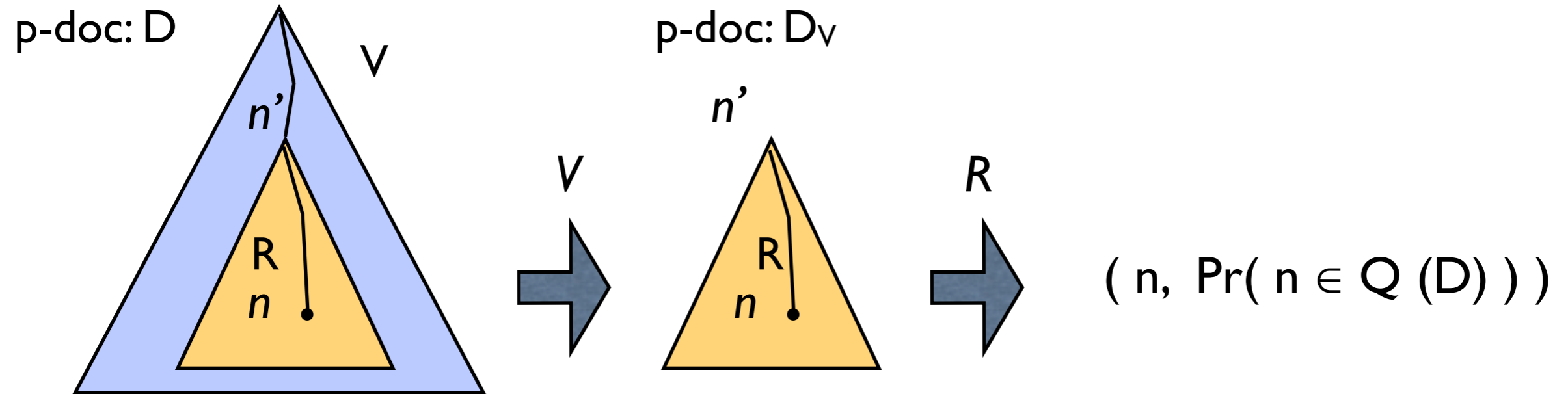
- In the deterministic case:
 - all views that give us answers are good
 - finding a rewriting for views can be done in PTIME [Xu,Ozsoyoglu'05]
- What about probabilistic case?
 - If a view gives answers (by a compensation), when will it give their probabilities?

Rewriting by Compensation: General Idea

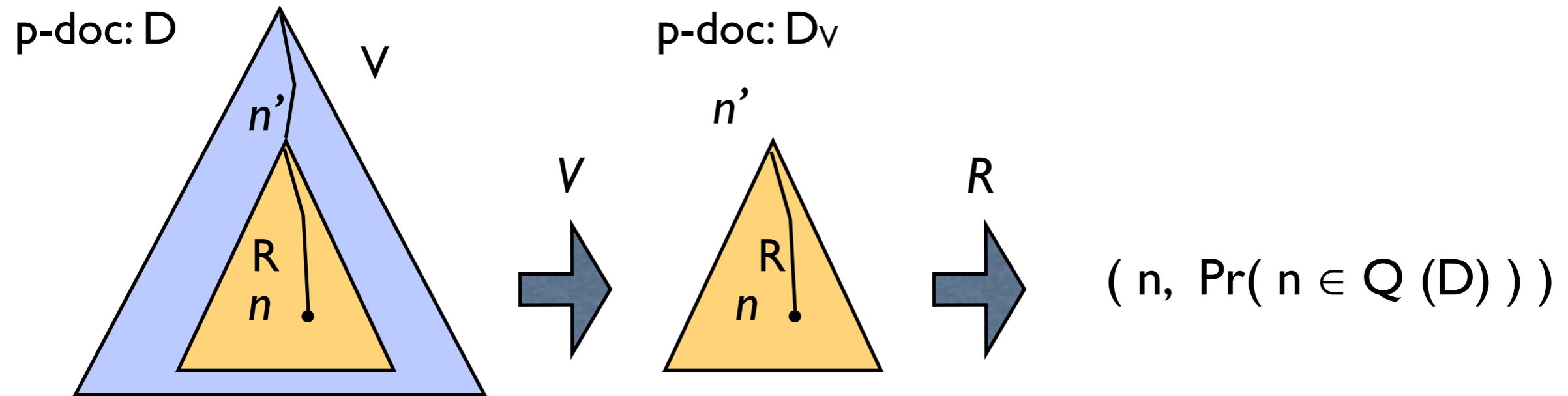


- In the deterministic case:
 - all views that give us answers are good
 - finding a rewriting for views can be done in PTIME [Xu,Ozsoyoglu'05]
- What about probabilistic case?
 - If a view gives answers (by a compensation), when will it give their probabilities?

A Natural Approach



A Natural Approach

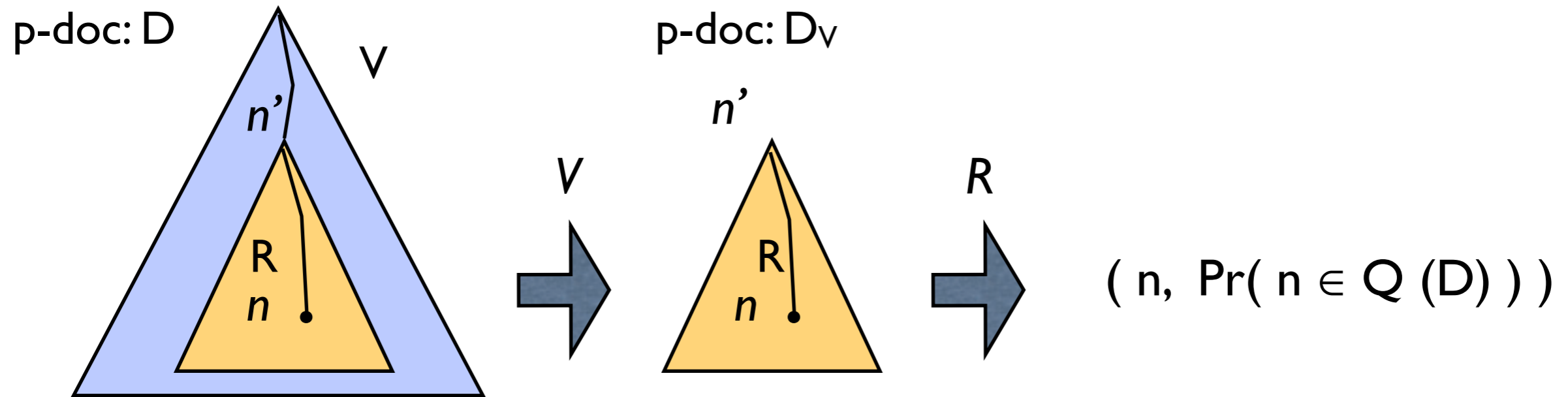


$\Pr(n \in Q (D)) =$ Prob. that a node n is returned by the query

$\Pr(n' \in V(D)) \times$ Prob. to return a node n' by the view V

$\Pr(n \in R (D_V))$ Prob. to find n in D_V by the compensation R

A Natural Approach



$\Pr(n \in Q (D)) =$ Prob. that a node n is returned by the query

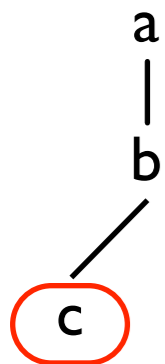
$\Pr(n' \in V(D)) \times$ Prob. to return a node n' by the view V

$\Pr(n \in R (D_V))$ Prob. to find n in D_V by the compensation R

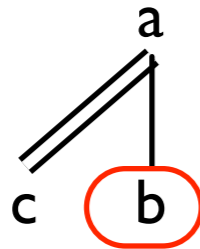
Does this approach work in general?

Views w/ Answers but w/o Probabilities

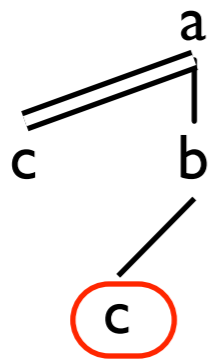
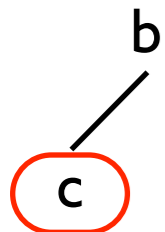
Query Q:



View V:

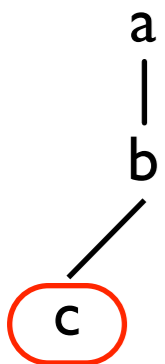


Rewriting R: Compensated query:

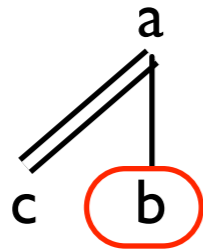


Views w/ Answers but w/o Probabilities

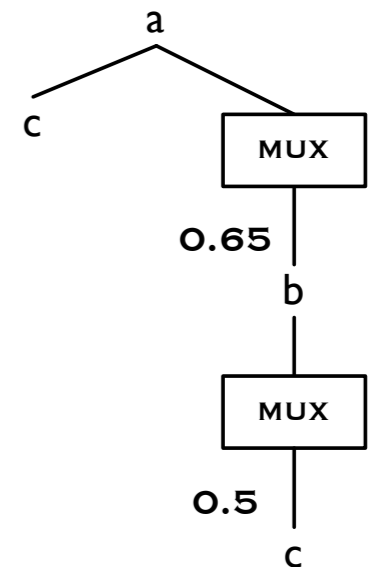
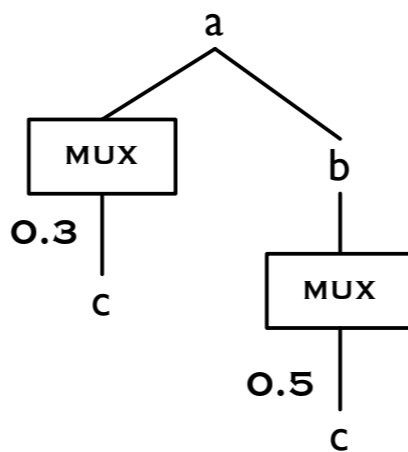
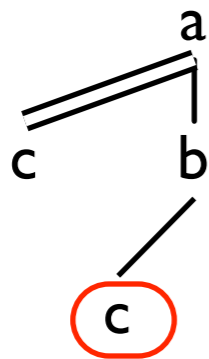
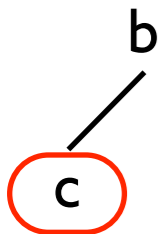
Query Q:



View V:

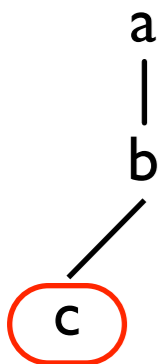


Rewriting R: Compensated query:

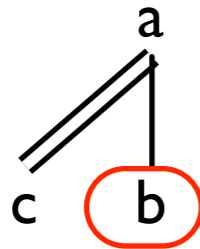


Views w/ Answers but w/o Probabilities

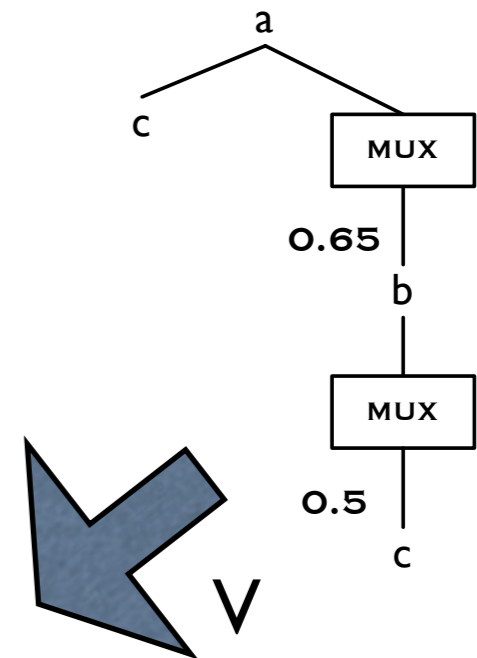
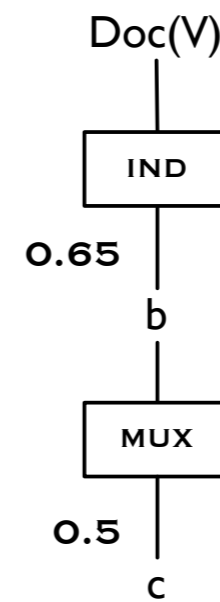
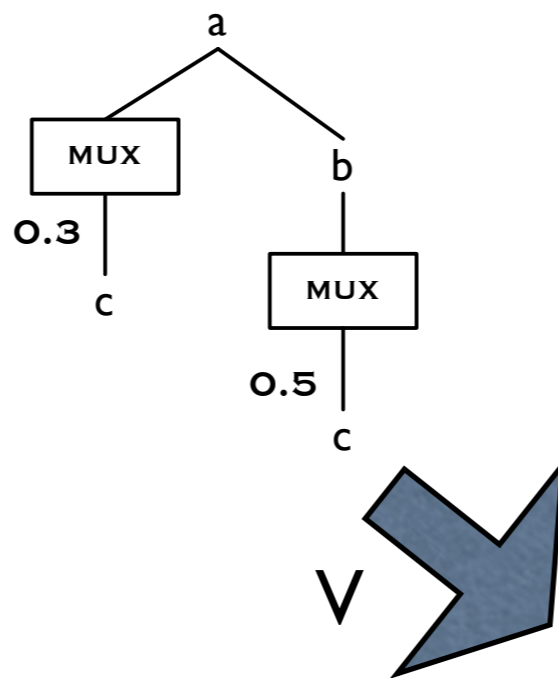
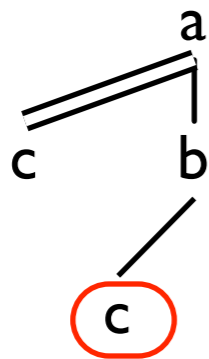
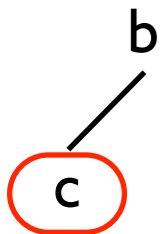
Query Q:



View V:

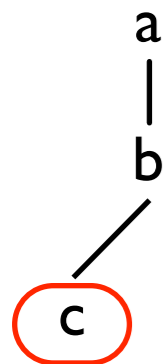


Rewriting R: Compensated query:

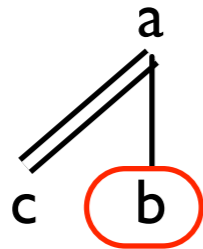


Views w/ Answers but w/o Probabilities

Query Q:



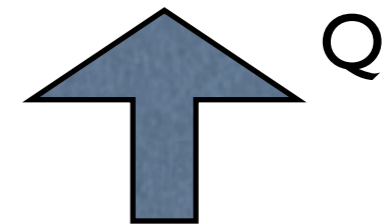
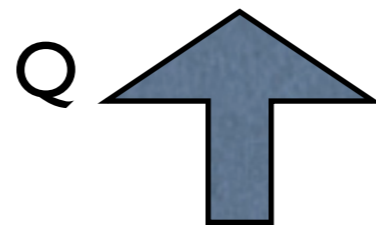
View V:



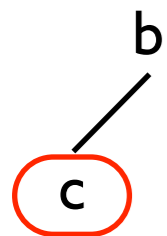
(c, 0.5)

≠

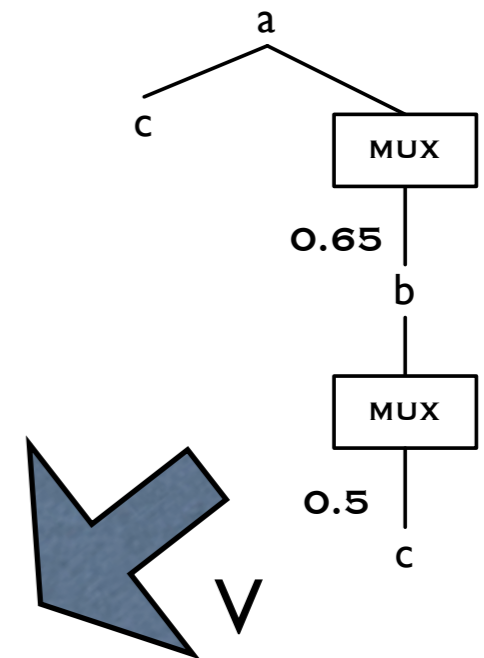
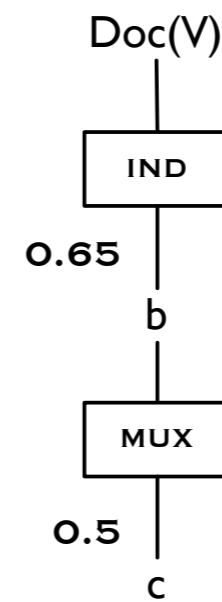
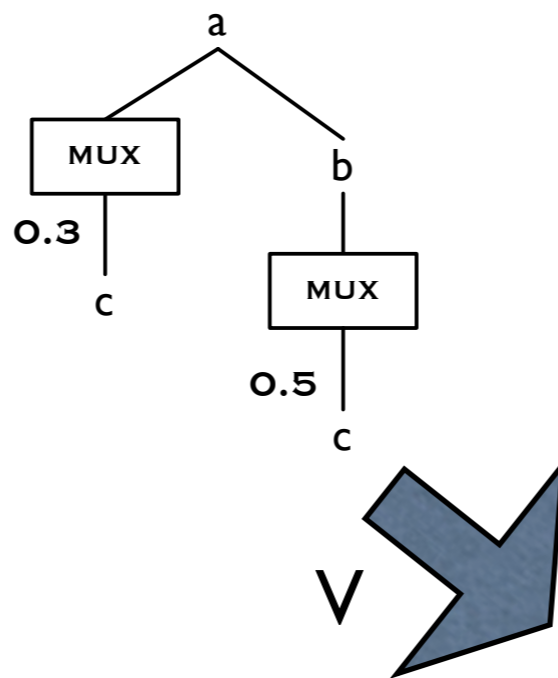
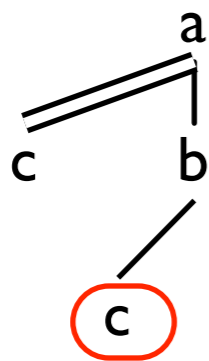
(c, 0.325)



Rewriting R:



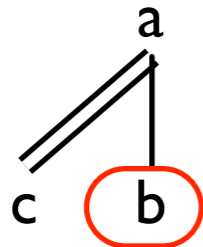
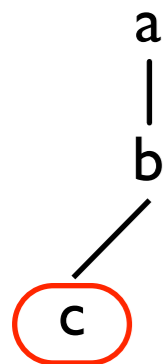
Compensated query:



Views w/ Answers but w/o Probabilities

Query Q:

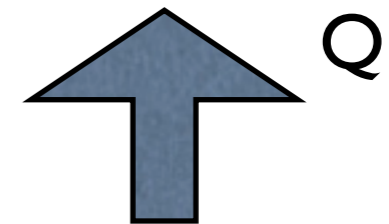
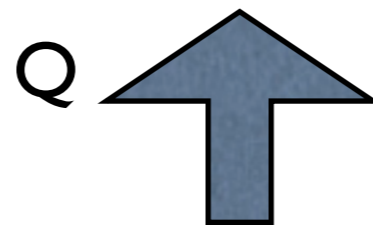
View V:



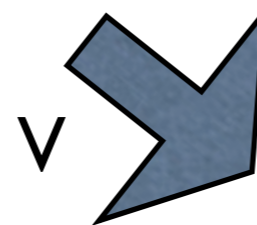
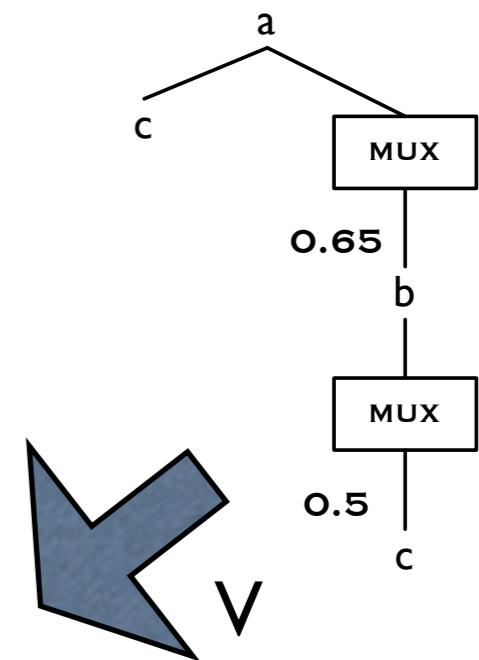
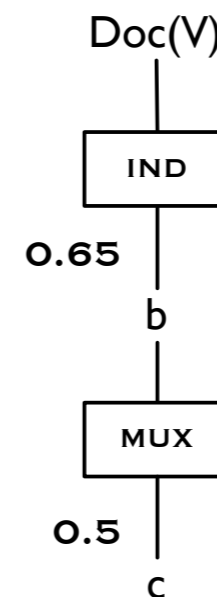
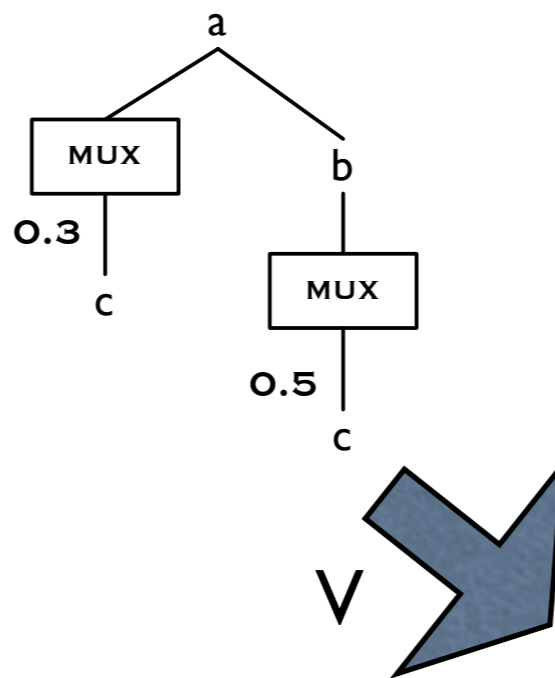
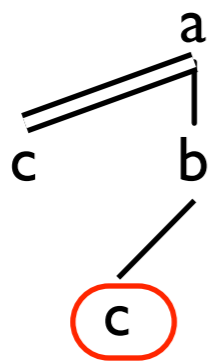
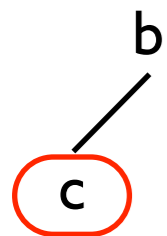
(c, 0.5)

≠

(c, 0.325)



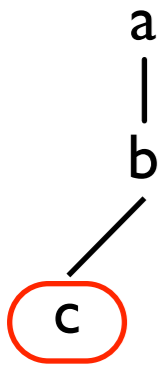
Rewriting R: Compensated query:



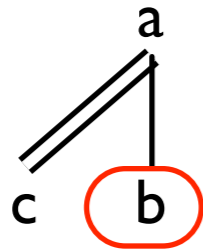
Views **cannot** distinguish these 2 p-docs.
Query **can** distinguish them

Views w/ Answers but w/o Probabilities

Query Q:



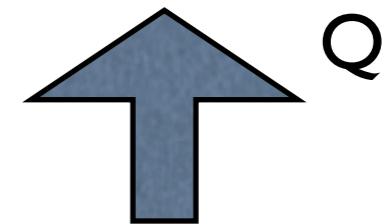
View V:



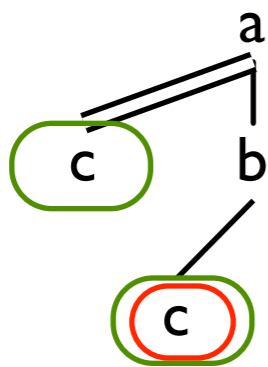
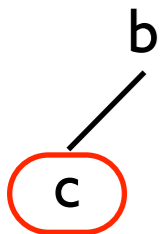
(c, 0.5)

≠

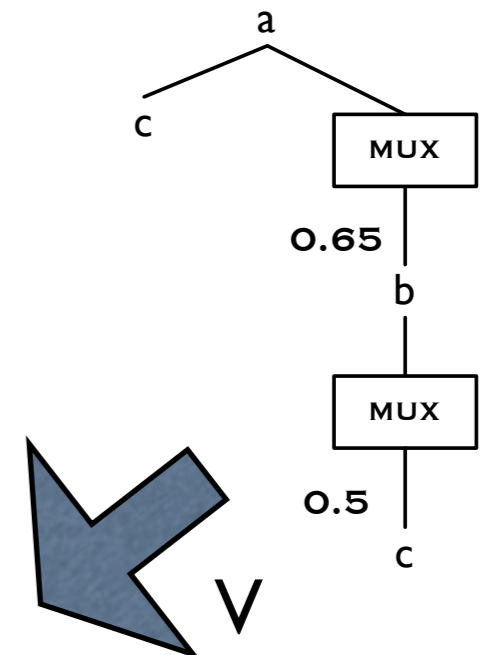
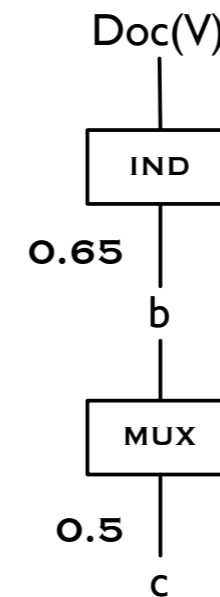
(c, 0.325)



Rewriting R: Compensated query:



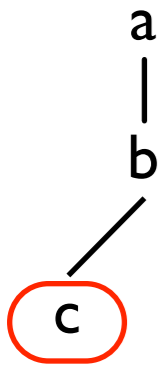
These two c-nodes interact, probabilistically dependent



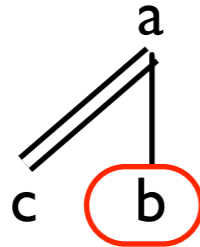
Views **cannot** distinguish these 2 p-docs.
Query **can** distinguish them

Views w/ Answers but w/o Probabilities

Query Q:



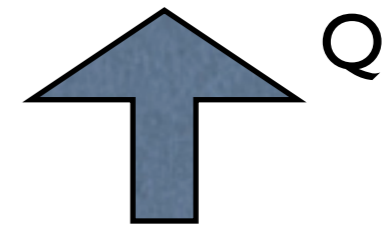
View V:



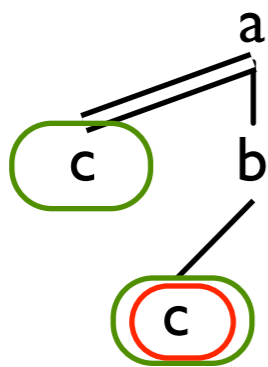
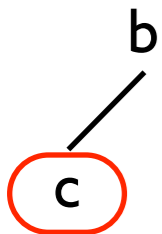
(c, 0.5)

≠

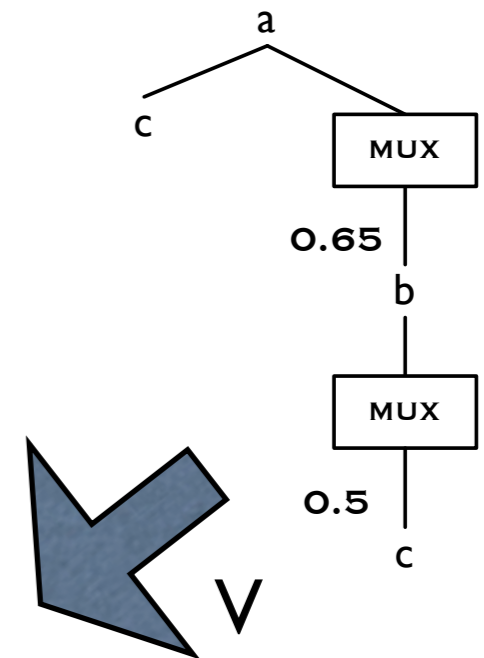
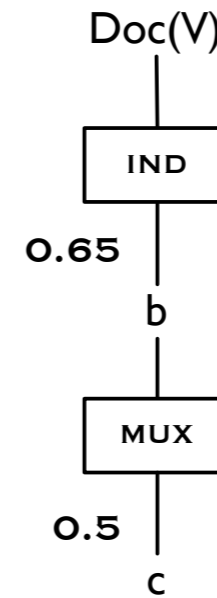
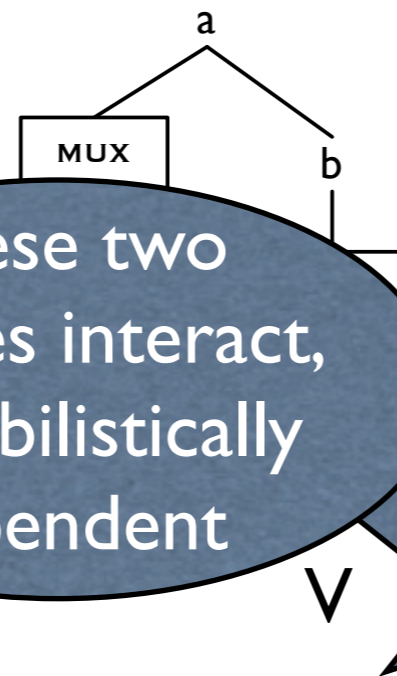
(c, 0.325)



Rewriting R: Compensated query:



These two c-nodes interact, probabilistically dependent



Views **cannot** distinguish these 2 p-docs.
Query **can** distinguish them

Problem: Probabilistic **dependency** between the view and its compensation

Probabilistically Independent Queries

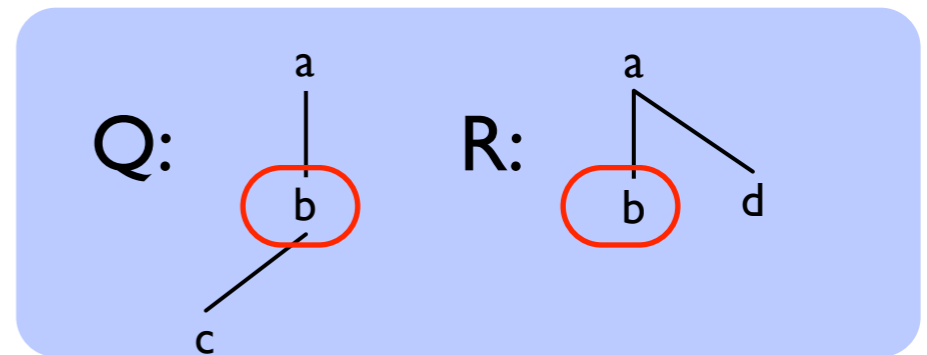
- Reason to introduce:
as a response to non-existence of probabilistic rewritings
- Q and R are probabilistically independent queries iff

$$\Pr(n \in Q \cap R (\triangle)) = \Pr(n \in Q (\triangle)) \times \Pr(n \in R (\triangle)) / \Pr(n \in (\triangle))$$

Probabilistically Independent Queries

- Reason to introduce:
as a response to non-existence of probabilistic rewritings
- Q and R are probabilistically independent queries iff

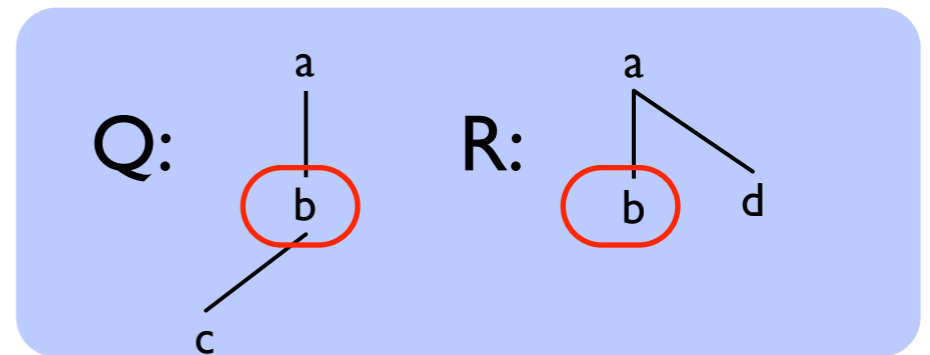
$$\Pr(n \in Q \cap R (\triangle)) = \Pr(n \in Q (\triangle)) \times \Pr(n \in R (\triangle)) / \Pr(n \in (\triangle))$$



Probabilistically Independent Queries

- Reason to introduce:
as a response to non-existence of probabilistic rewritings
- Q and R are probabilistically independent queries iff

$$\Pr(n \in Q \cap R (\triangle)) = \Pr(n \in Q (\triangle)) \times \Pr(n \in R (\triangle)) / \Pr(n \in (\triangle))$$

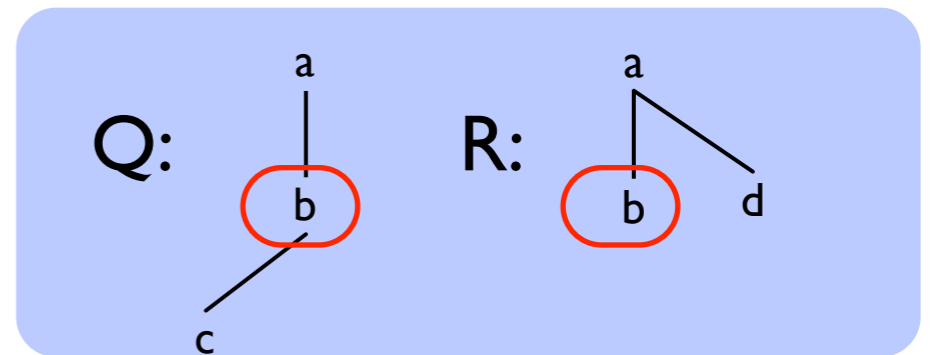


- Proposition:
probabilistic independence of tree-patterns can be decided in PTIME

Probabilistically Independent Queries

- Reason to introduce:
as a response to non-existence of probabilistic rewritings
- Q and R are probabilistically independent queries iff

$$\Pr(n \in Q \cap R (\triangle)) = \Pr(n \in Q (\triangle)) \times \Pr(n \in R (\triangle)) / \Pr(n \in (\triangle))$$



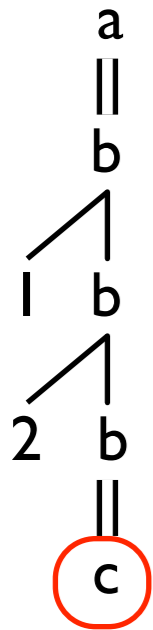
- Proposition:
probabilistic independence of tree-patterns can be decided in PTIME

Question:

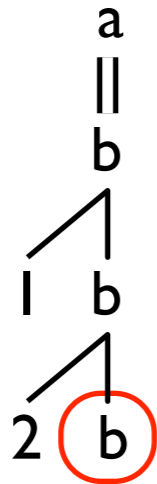
Is probabilistic independence **enough** to guarantee existence of probability functions for compensating rewritings?

Probabilistic Independence is not Sufficient

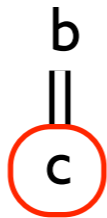
Query Q:



View V:

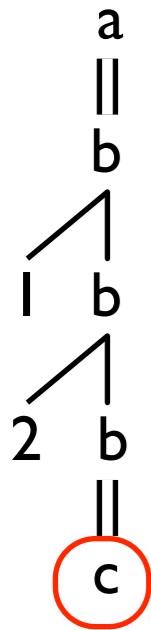


Rewriting R:

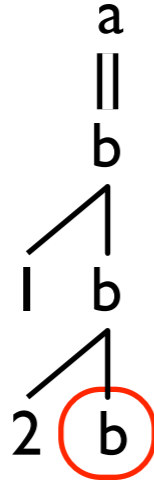


Probabilistic Independence is not Sufficient

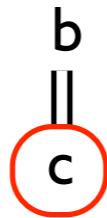
Query Q:



View V:



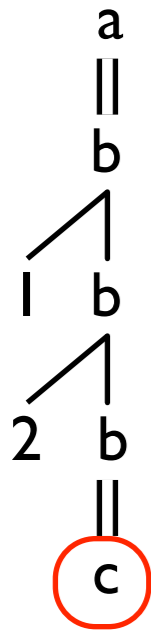
Rewriting R:



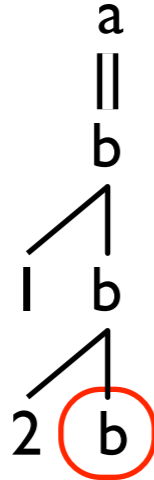
V and R are prob.
independent

Probabilistic Independence is not Sufficient

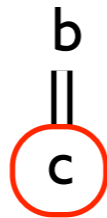
Query Q:



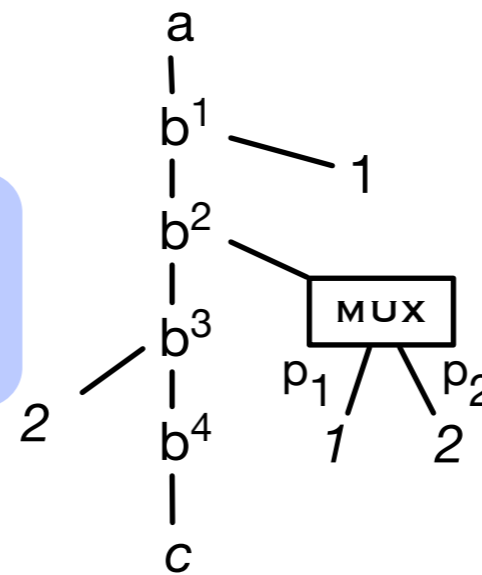
View V:



Rewriting R:

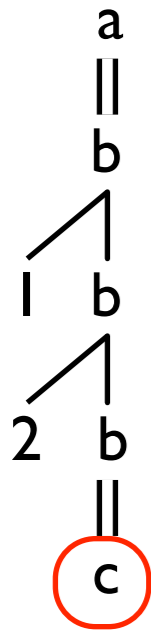


V and R are prob. independent

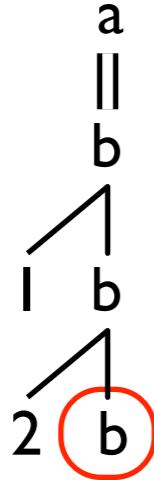


Probabilistic Independence is not Sufficient

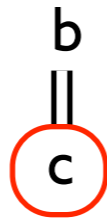
Query Q:



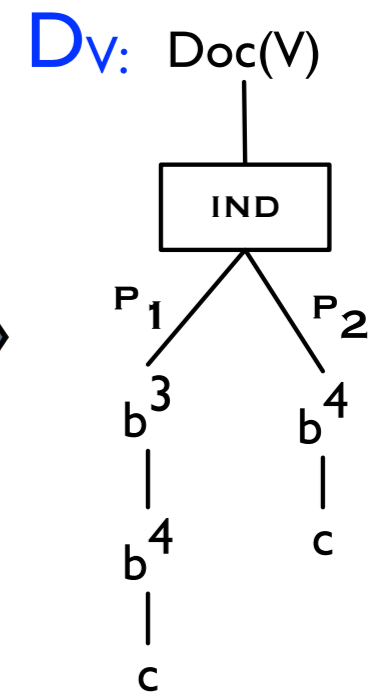
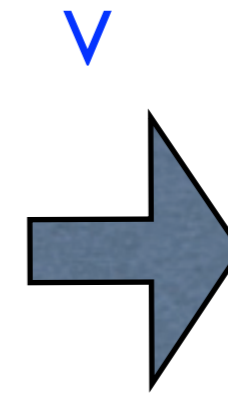
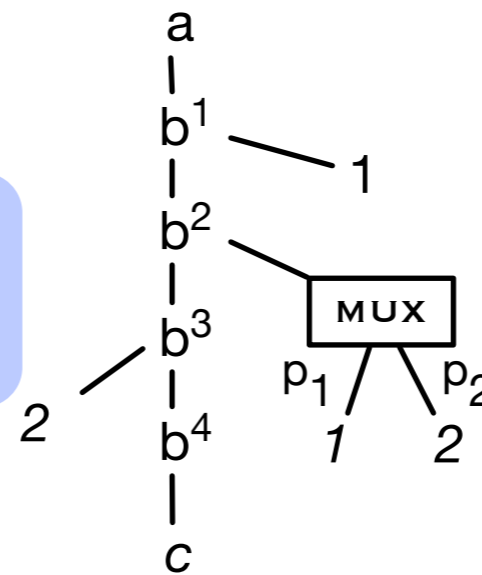
View V:



Rewriting R:

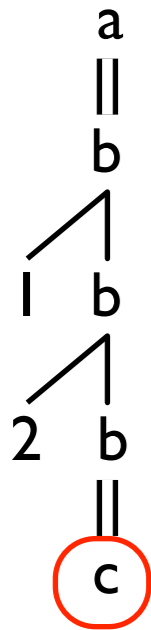


V and R are prob. independent

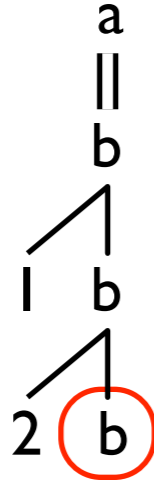


Probabilistic Independence is not Sufficient

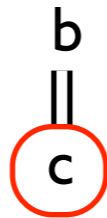
Query Q:



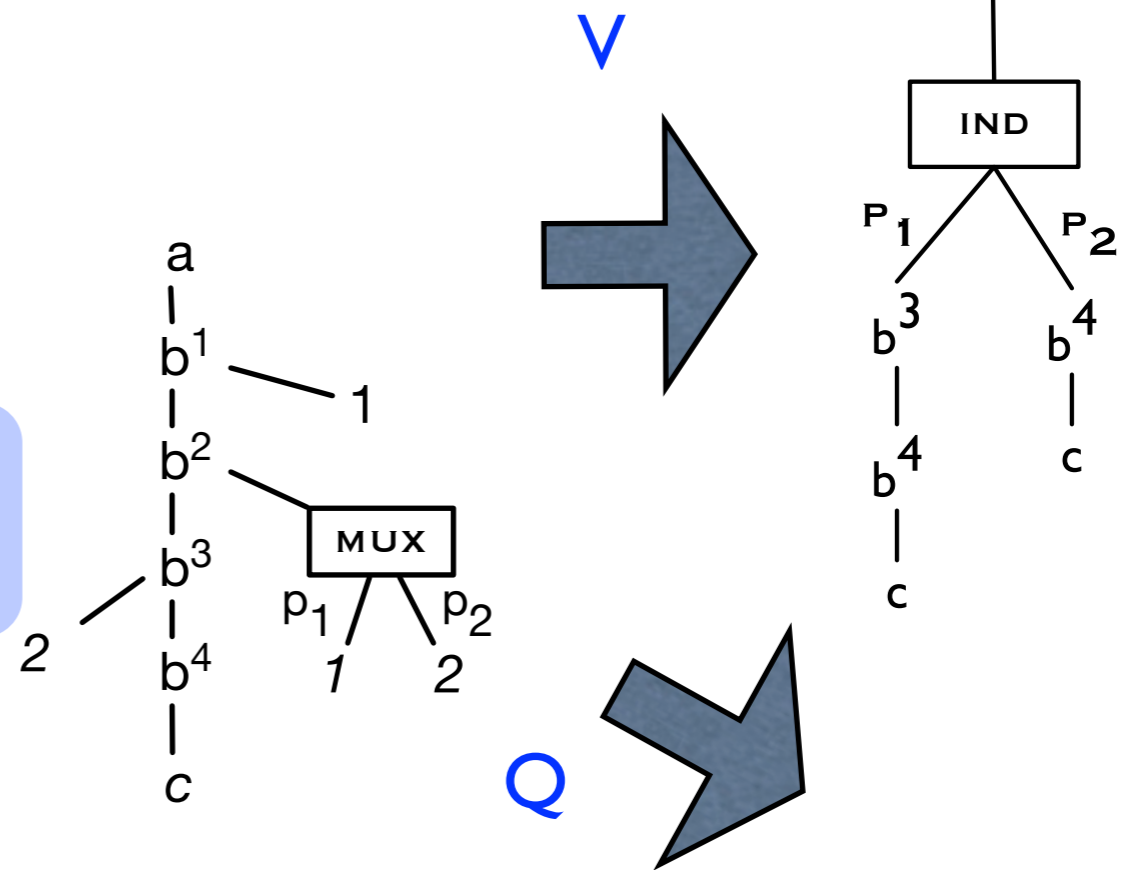
View V:



Rewriting R:



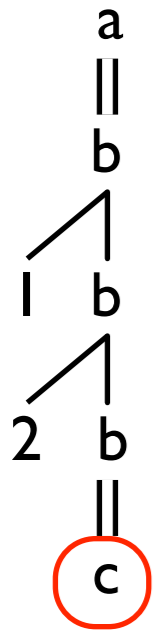
V and R are prob. independent



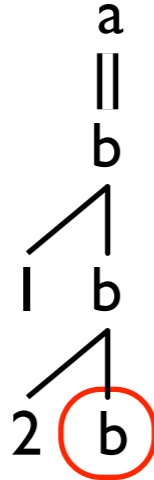
$(c, p_1 + p_2 - p_1 \times p_2)$

Probabilistic Independence is not Sufficient

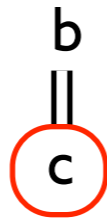
Query Q:



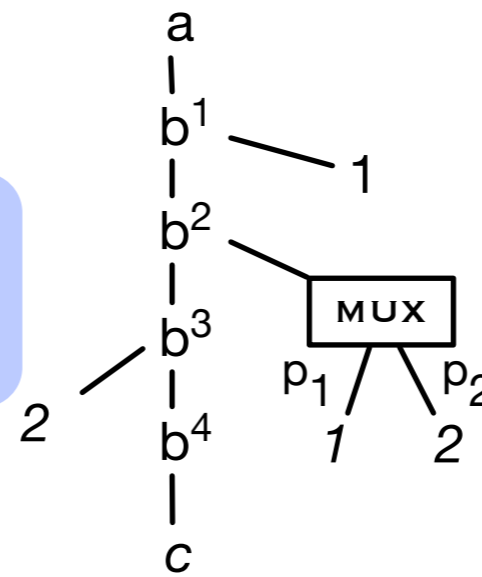
View V:



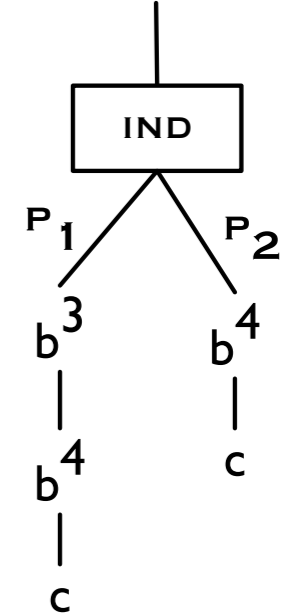
Rewriting R:



V and R are prob. independent



D_V : Doc(V)



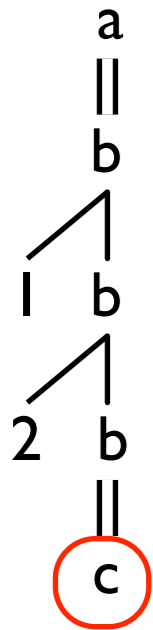
View p-document D_V

- knows that b³ and b⁴ are prob. related,
- but it does not know that relation is via MUX

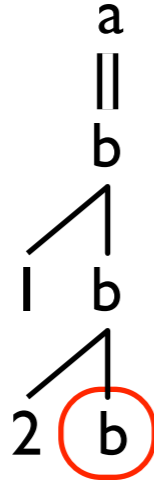
$$(c, p_1 + p_2 - p_1 \times p_2)$$

Probabilistic Independence is not Sufficient

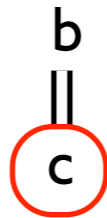
Query Q:



View V:

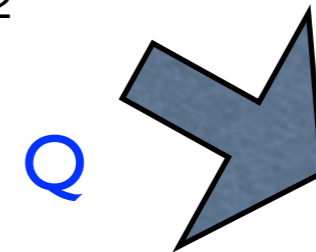
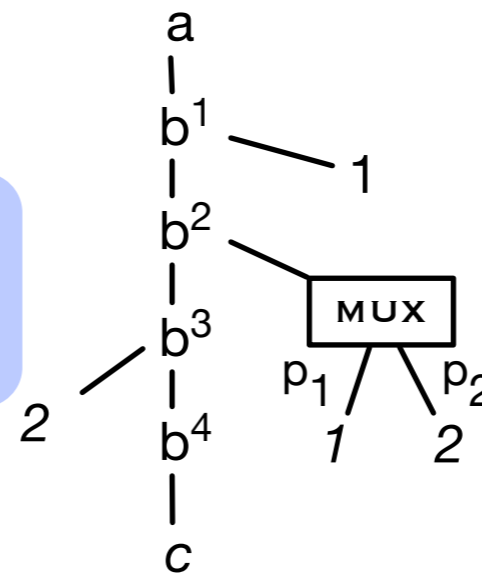
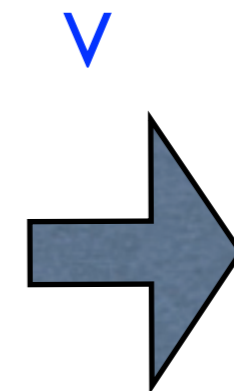
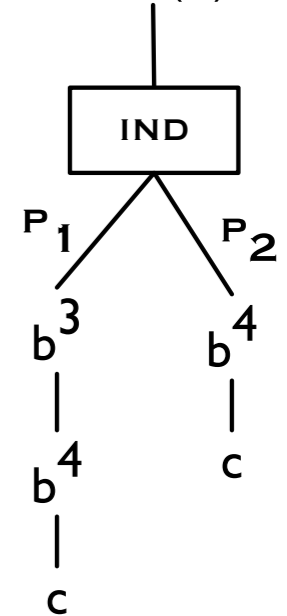


Rewriting R:



V and R are prob. independent

D_V : Doc(V)



$$(c, p_1 + p_2 - p_1 \times p_2)$$

View p-document D_V

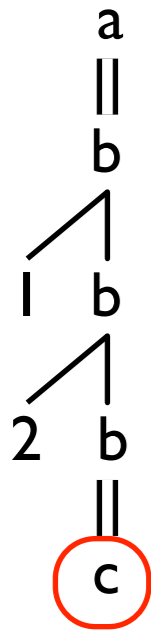
- knows that b^3 and b^4 are prob. related,
- but it does not know that relation is via MUX

Problem:

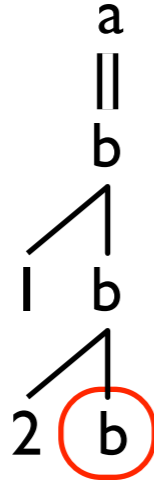
View has **not enough information** to compute probabilities of queries

Probabilistic Independence is not Sufficient

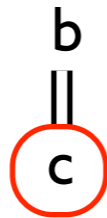
Query Q:



View V:

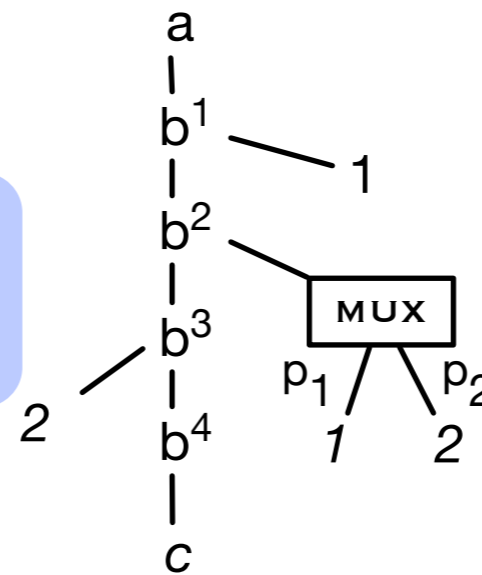
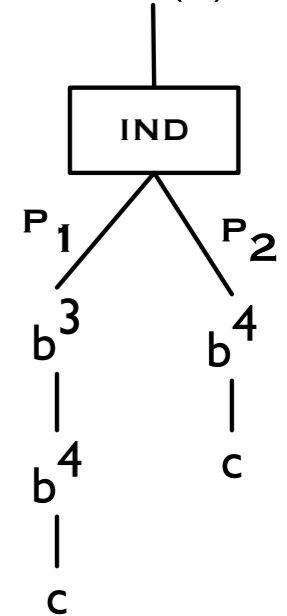


Rewriting R:



V and R are prob. independent

D_V : Doc(V)



Q

$$(c, p_1 + p_2 - p_1 \times p_2)$$

View p-document D_V

- knows that b^3 and b^4 are prob. related,
- but it does not know that relation is via MUX

Problem:

View has **not enough information** to compute probabilities of queries

Why this happens?

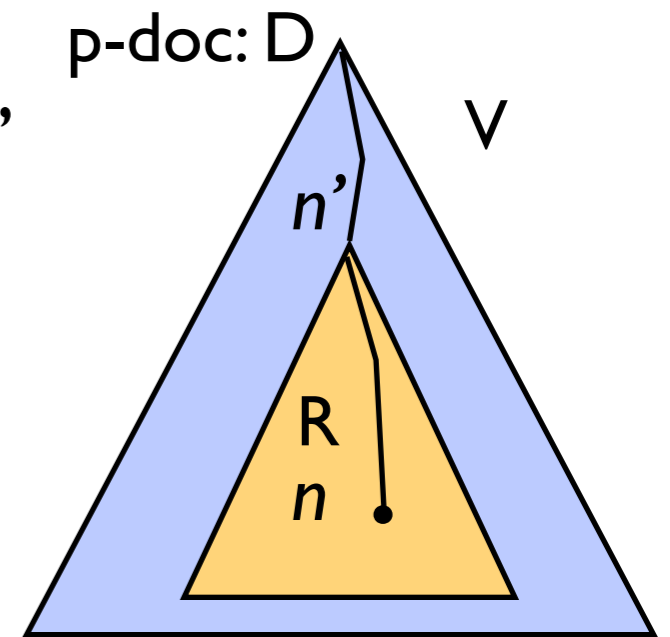
Here: predicates in Q below // -edge
In general: we do not know

Restricted Compensation

- **Simple query:**
 - no `//`-edges or
 - no predicates after the first `//`-edge
- **Restricted compensation** of V with R :
 - view V is simple or
 - rewriting R has no `//`-edges

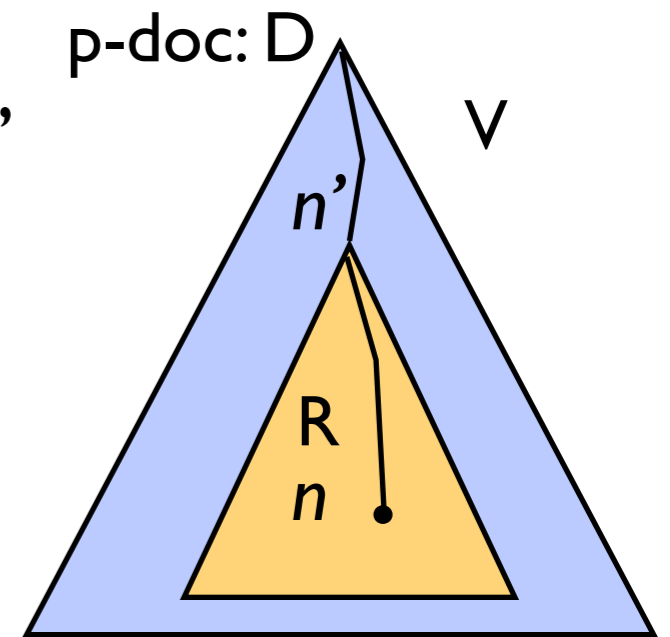
Probabilities for Restricted Compensation

- Theorem: If V has no predicated on the output node,
- compensation of V with R is restricted and
- V is probabilistically independent from R , then



Probabilities for Restricted Compensation

- Theorem: If V has no predicated on the output node,
- compensation of V with R is restricted and
- V is probabilistically independent from R , then



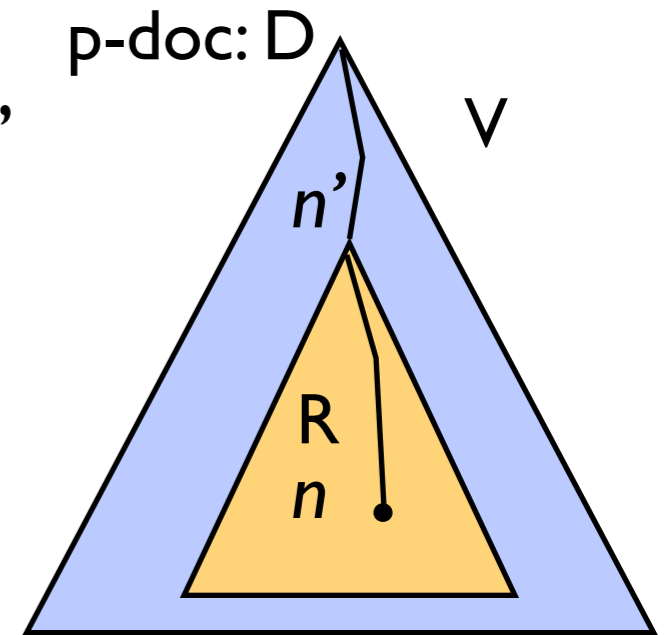
$\Pr(n \in Q(D)) =$ Prob. that a node n is returned by the query

$\Pr(n' \in V(D)) \times$ Prob. to return a node n' by the view V

$\Pr(n \in R(D_V))$ Prob. to find n in D_V by the compensation R

Probabilities for Restricted Compensation

- Theorem: If V has no predicated on the output node,
 - compensation of V with R is restricted and
 - V is probabilistically independent from R , then



$\Pr(n \in Q(D)) =$ Prob. that a node n is returned by the query

$\Pr(n' \in V(D)) \times$ Prob. to return a node n' by the view V

$\Pr(n \in R(D_V))$ Prob. to find n in D_V by the compensation R

Theorem can be generalized to views with predicates on output nodes

Outline

- Rewriting over probabilistic XML
- Rewriting by compensation
- Rewriting by intersection

Outline

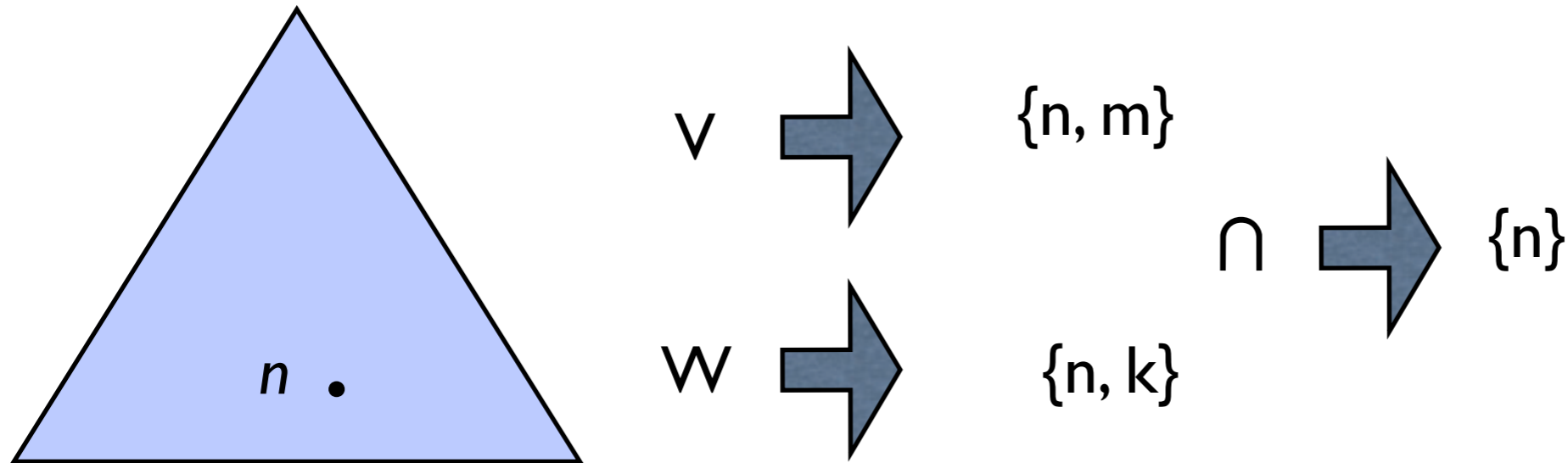
- Rewriting over probabilistic XML
- Rewriting by compensation
- Rewriting by intersection

Outline

- Rewriting over probabilistic XML
- Rewriting by compensation
- Rewriting by intersection

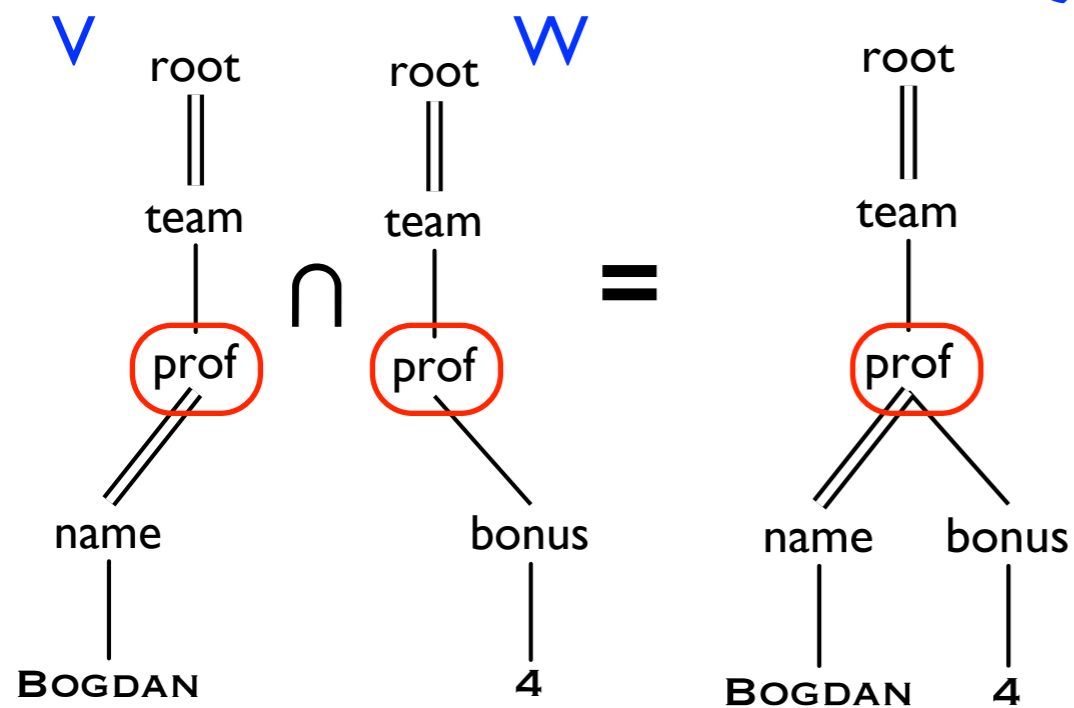
Rewriting by Intersection

[Cautis&al'11]



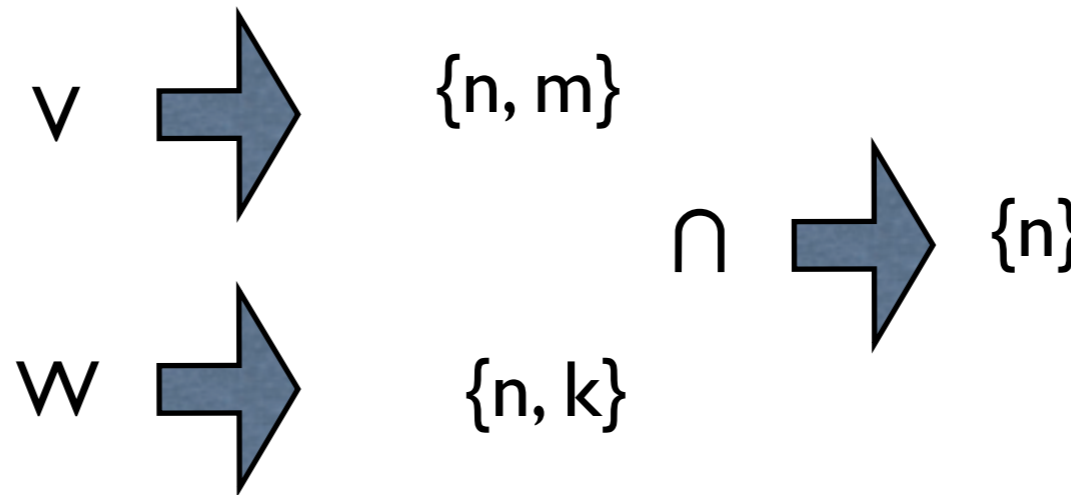
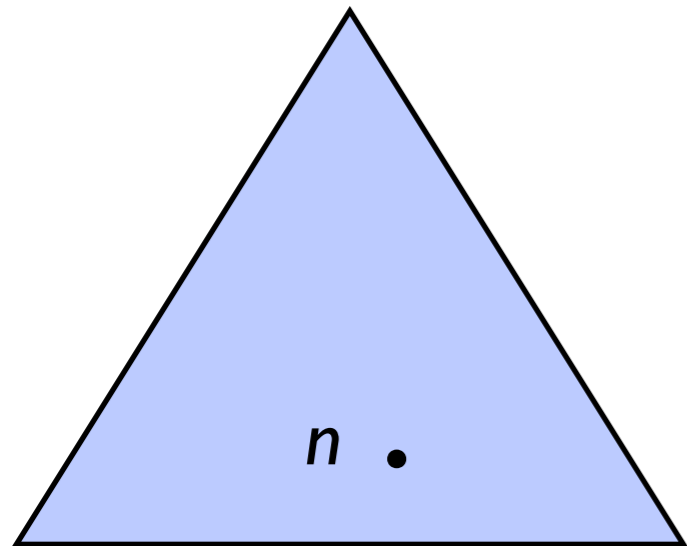
Intersection of views:

$$V \cap W \equiv Q$$



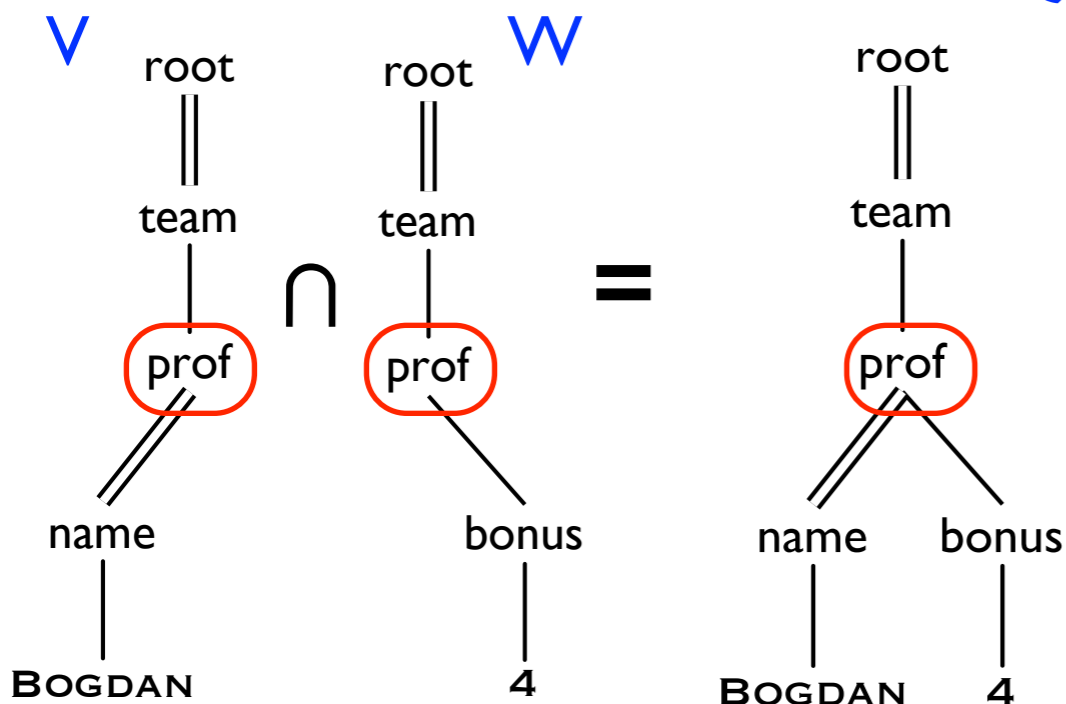
Rewriting by Intersection

[Cautis&al'11]



Intersection of views:

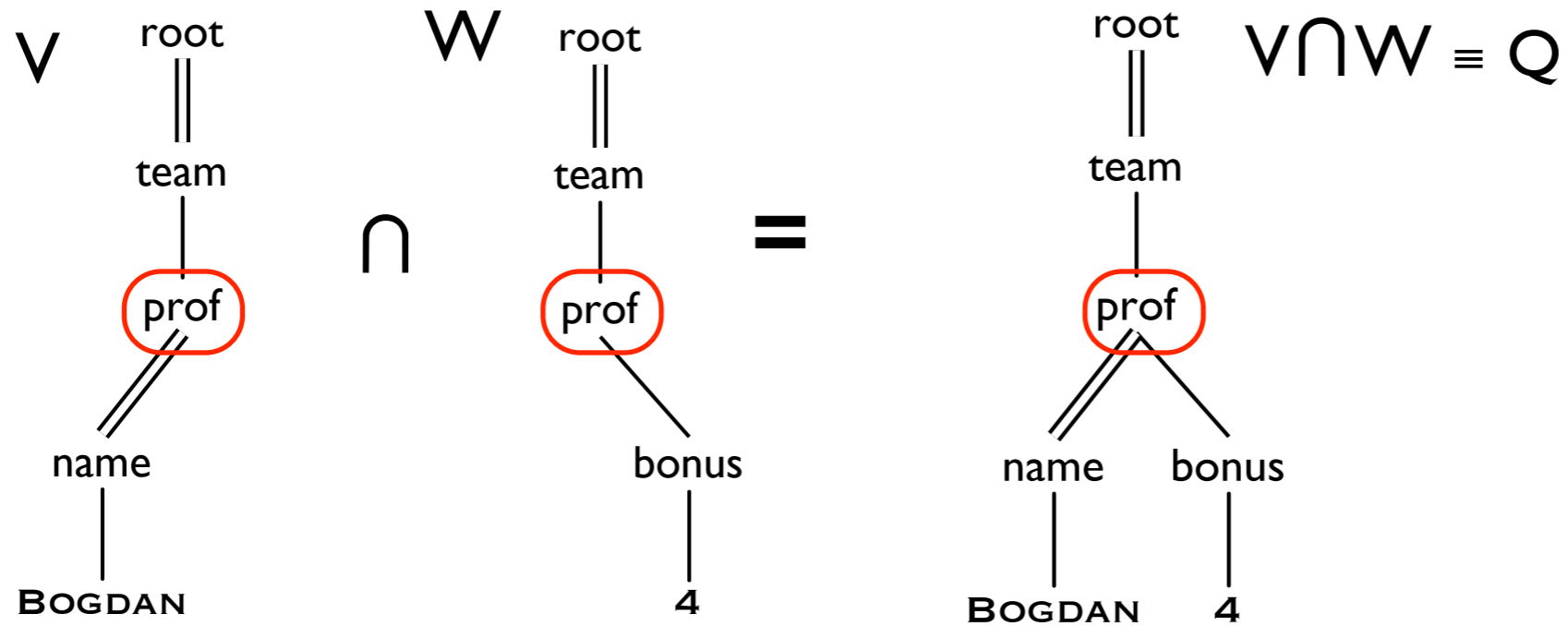
$$V \cap W \equiv Q$$



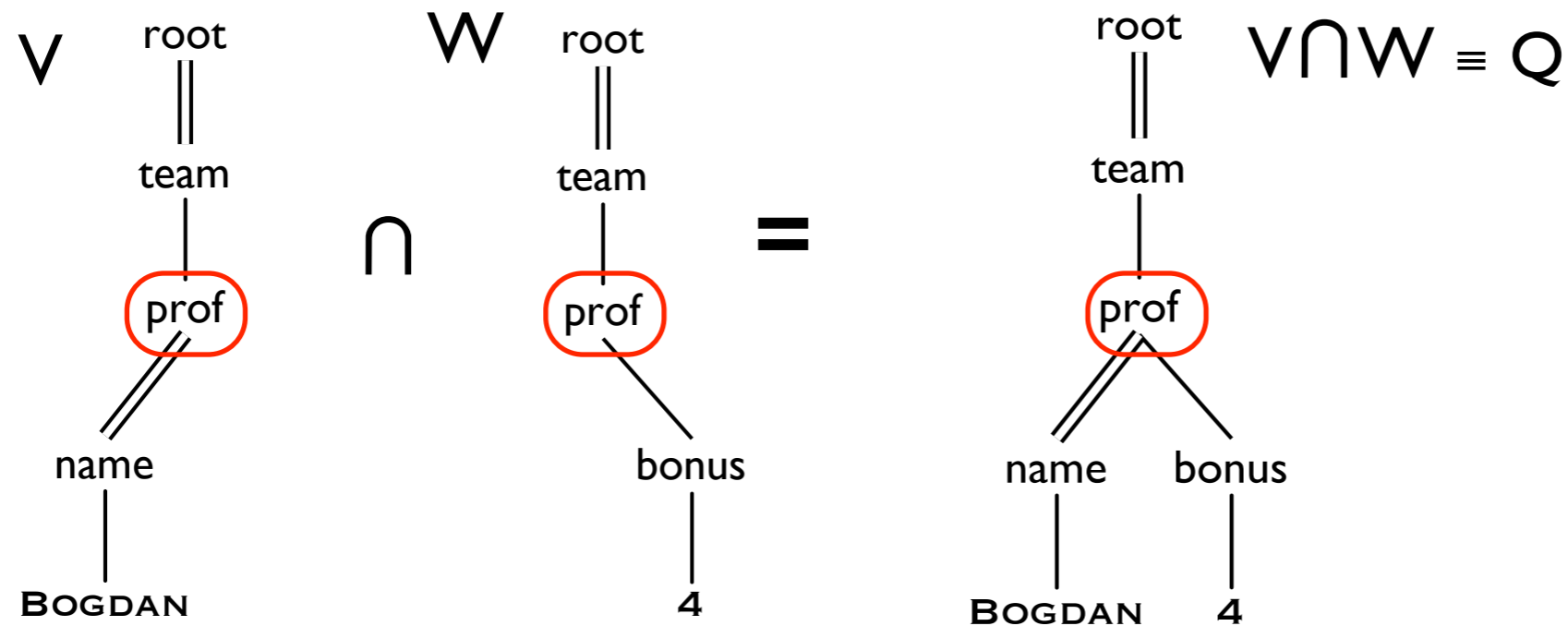
- Intersection of tree-patterns is defined using **interleaving**
- Given Q and a set of views S deciding existence of $\{V_1, \dots, V_n\} \subseteq S$ s.t.

$$V_1 \cap \dots \cap V_n \equiv Q$$
 is **coNP-hard**
- If views are **extended skeletons**, then the decision is PTIME

A Natural Approach

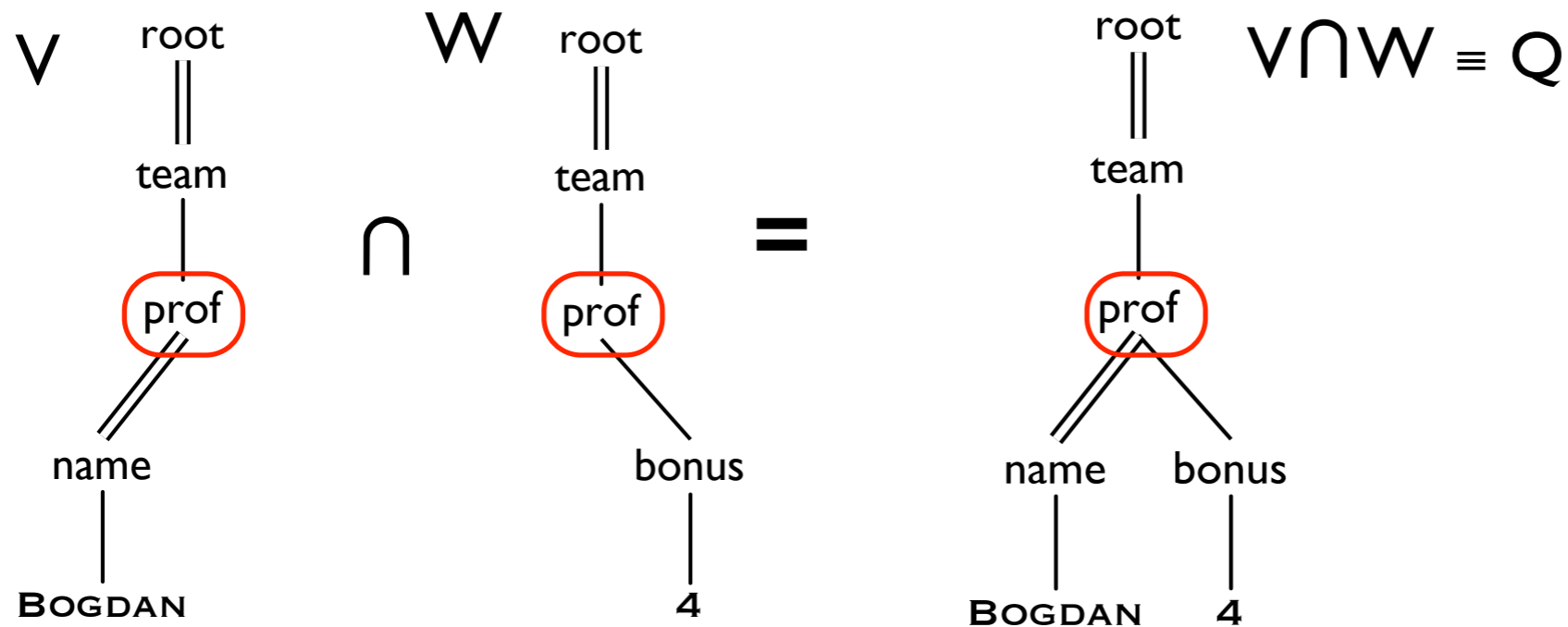


A Natural Approach



$\Pr(n \in V \cap W (D)) =$ Prob. that a node n is returned by the query
 $\Pr(n \in V (D)) \times$ Prob. that a node n is returned by the view V
 $\Pr(n \in W (D)) /$ Prob. that a node n is returned by the view W
 $\Pr(n \in D)$ Prob. that a node n is in D

A Natural Approach



$\Pr(n \in V \cap W (D)) =$ Prob. that a node n is returned by the query
 $\Pr(n \in V (D)) \times$ Prob. that a node n is returned by the view V
 $\Pr(n \in W (D)) /$ Prob. that a node n is returned by the view W
 $\Pr(n \in D)$ Prob. that a node n is in D

Does this approach work in general?

Probabilities for Independent Intersections

- Theorem:
 - If V_1, \dots, V_n are pairwise independent and
 - $Q \equiv V_1 \cap \dots \cap V_n$
 - and there is V such that $\Pr(n \in D) = \Pr(n \in V)$
 - then
$$\Pr(n \in Q \mid D) = \Pr(n \in V_1 \mid D) \times \dots \times \Pr(n \in V_n \mid D) / \Pr^{n-1}(n \in V \mid D)$$

Probabilities for Independent Intersections

- **Theorem:**
 - If V_1, \dots, V_n are pairwise independent and
 - $Q \equiv V_1 \cap \dots \cap V_n$
 - and there is V such that $\Pr(n \in D) = \Pr(n \in V)$
 - then
$$\Pr(n \in Q(D)) = \Pr(n \in V_1(D)) \times \dots \times \Pr(n \in V_n(D)) / \Pr^{n-1}(n \in V(D))$$
- **Theorem:** (by reduction from k -dimensional perfect matching)
 - Given a set of views S and query Q w/o //-edges
 - Deciding existence of pairwise independent subset $M \subseteq S$ such that $Q \equiv \cap M$ is NP-hard

Conclusion and Future Work

- For **compensation**:
 - In our setting view based query answering: PTIME in data + query.
Direct query evaluation: PTIME in data, EXP in query
- For **intersection**:
 - PTIME in data complexity while reasoning over views is intractable
 - For intersection probability computation is conceptually simple:
take a product of probabilities and divide by a probability.
Direct query evaluation based on dynamic programming
- **Next steps**:
 - Work on more general settings
 - How views can help to probabilistic XML with global dependencies?

Thank you!



Webdam Project

Foundations of Web Data Management

ERC FP7 grant, agreement n. 226513

<http://webdam.inria.fr/>

References

- [\[Kimelfeld&al'07\]](#) - Benny Kimelfeld, Yehoshua Sagiv: Matching Twigs in Probabilistic XML. VLDB 2007: 27-38
- [\[Senellart&al'07\]](#) - P. Senellart, S. Abiteboul: On the complexity of managing probabilistic XML data. PODS 2007
- [\[Xu,Ozsoyoglu'05\]](#) - W. Xu, Z. Ozsoyoglu.: Rewriting XPath queries using materialized views. In: Proc. VLDB. (2005) 121–132
- [\[Cautis&al'11\]](#) - Querying XML data sources that export very large sets of views. TODS (2011)