Merging Uncertain Multi-Version XML Documents

M. Lamine BA, Talel Abdessalem & Pierre Senellart

ACM DocEng 2013 - 1st International Workshop on Document Changes (Florence, Italy)

September 10^{th}, 2013

Merging feature: a need in open environments

- Merging documents related to the same topic or sharing a large common part, e.g., Wikipedia articles


It has been suggested that this article be merged with *Schrödinger picture*, *Heisenberg picture* and *Mathematical formulation of quantum mechanics#Pictures of dynamics* to *Dynamical pictures (quantum mechanics)*. (Discuss) *Proposed since September 2013.*

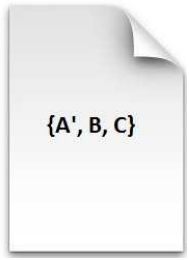
Merging feature: a need in open environments

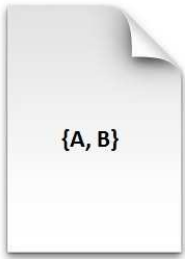
- Merging documents related to the same topic or sharing a large common part, e.g., Wikipedia articles

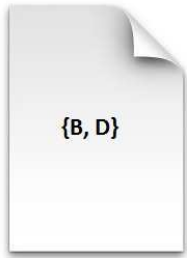
- Recommend the outcome of the merging of contributions of the most trustworthy contributors

- Proposal of a merging operation over multi-version tree-structured documents with uncertain data

Usual Merging Process over Documents

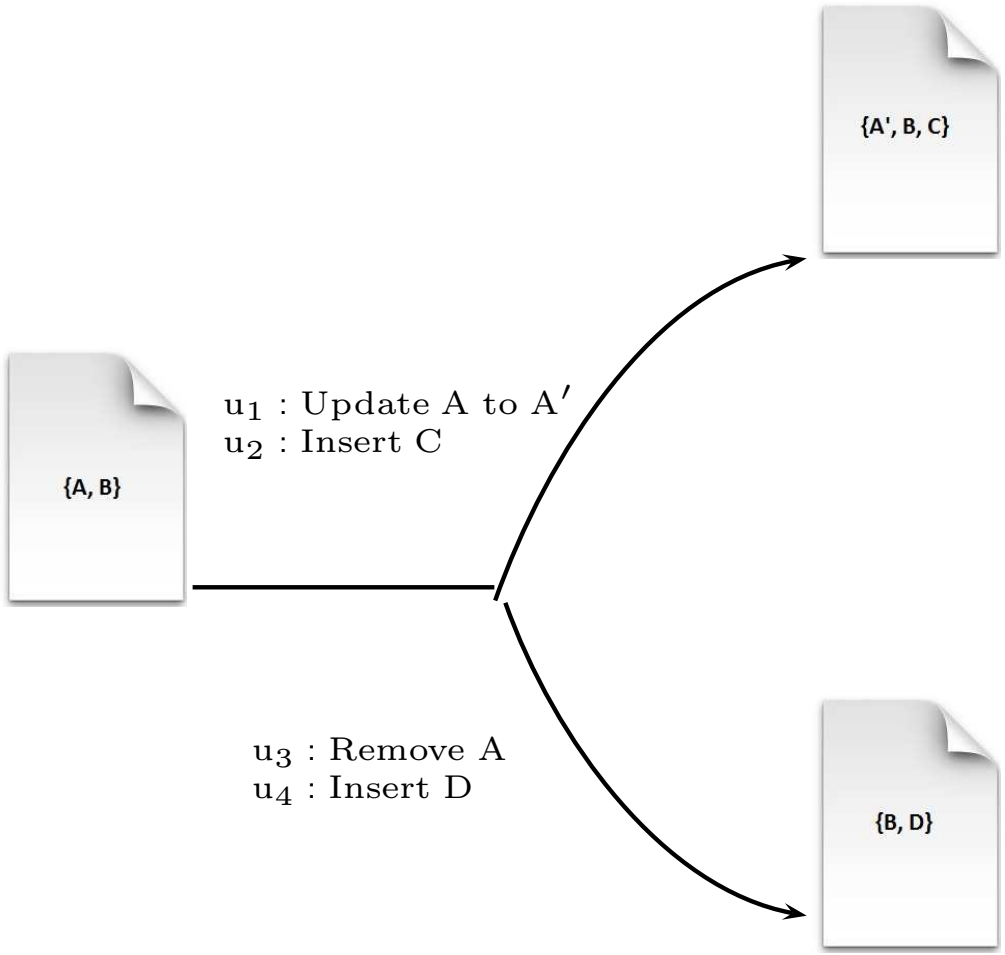






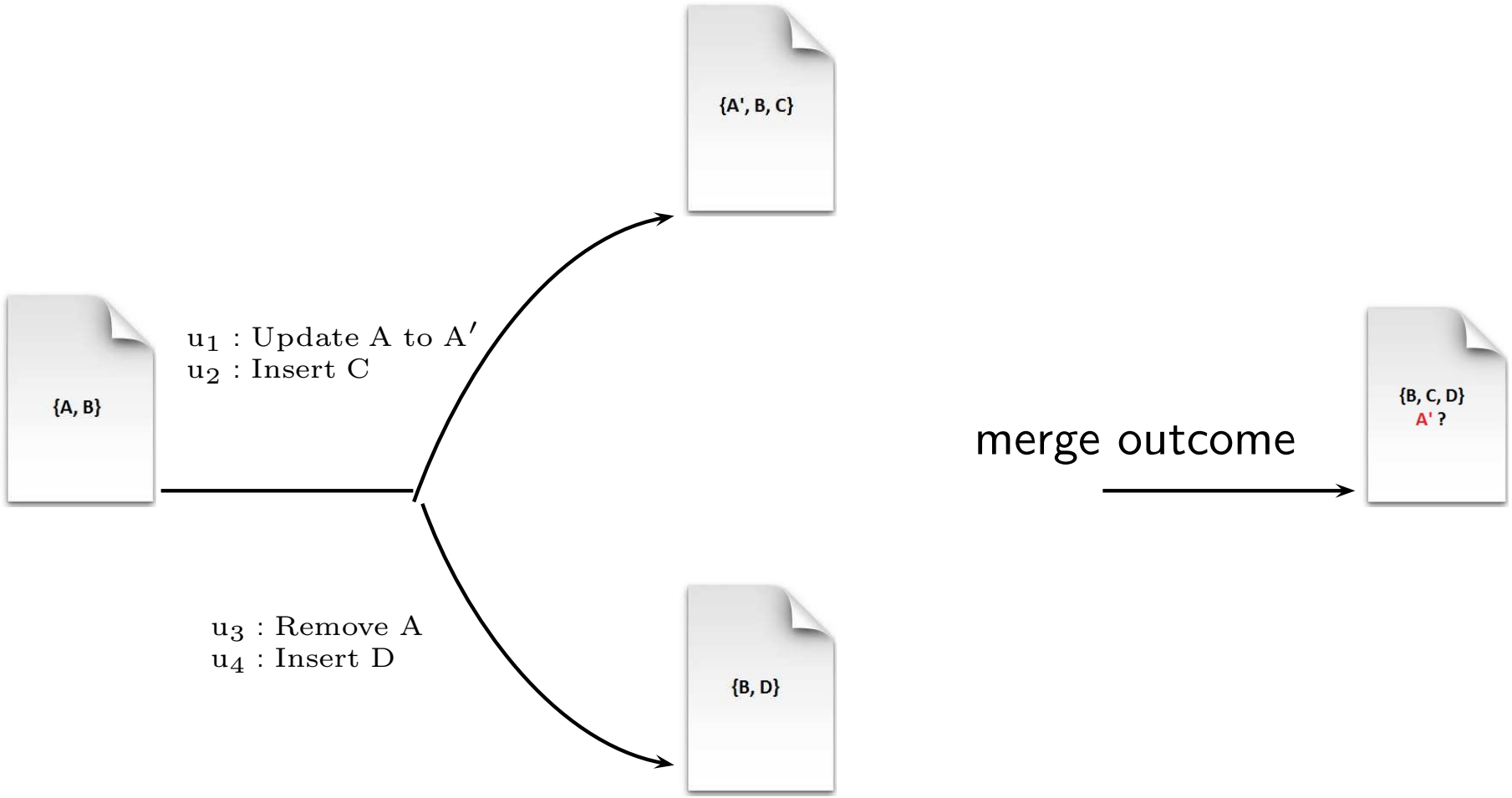
Usual Merging Process over Documents

(i) Change Detection (diff algorithm): $\{u_1, u_2\}$ and $\{u_3, u_4\}$



Usual Merging Process over Documents

(i) Change Detection (diff algorithm): $\{u_1, u_2\}$ and $\{u_3, u_4\}$
(ii) Three merge scenarios: {A', B, C, D}, {B, C, D} and {A, B, C, D}



State-of-the-art XML Merging algorithms

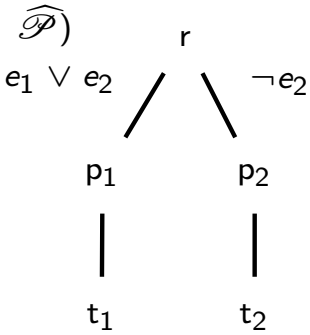
- Diff algorithms, for instance [Lindholm et al., 2006], for documents having tree-like structure
- Two-way merging [Suzuki, 2002], [Ma et al., 2010] vs. Three-way merging [Lindholm, 2004], [Abdessalem et al., 2011]
- All deterministic approaches require human input in the presence of uncertainties, e.g. conflicts handling, for the merge outcome
- Probabilistic merging, proposed in [Ma et al., 2010], does not retain enough information for retrieving back individual merged versions

Outline

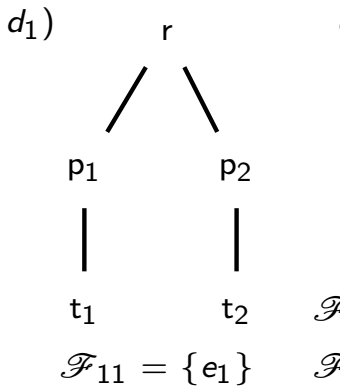
Uncertain Tree-Structured Data

Probabilistic XML [Kimelfeld & Senellart.(2013)]

- Unordered, unranked, and labeled **XML trees** with annotated edges
  - annotations are **propositional formulas** of random Boolean variables



PrXML$^{\text{fie}}$ p-Document



$Pr(e_1) = 0.2$

$Pr(e_2) = 0.8$

$Pr(d_1) = Pr(e_1) \times Pr(\neg e_2)$

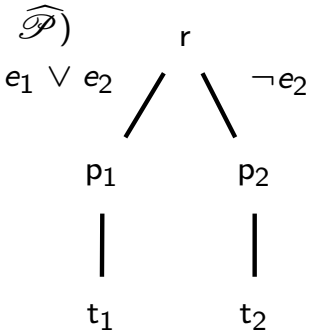
$Pr(d_2) = (Pr(\neg e_1) \times Pr(e_2)) + (Pr(e_1) \times Pr(e_2))$

Possible worlds and their probabilities

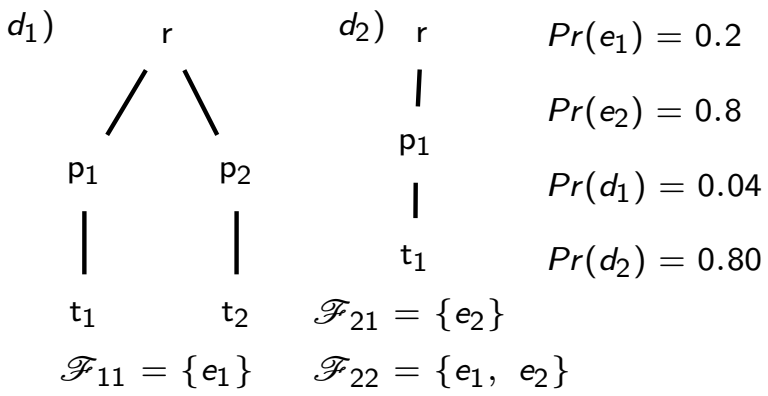
Uncertain Tree-Structured Data

Probabilistic XML [Kimelfeld & Senellart.(2013)]

- Unordered, unranked, and labeled **XML trees** with annotated edges
  - annotations are **propositional formulas** of random Boolean variables



PrXML$^{\text{fie}}$ p-Document

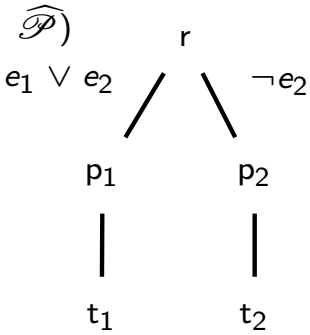


Possible worlds and their probabilities

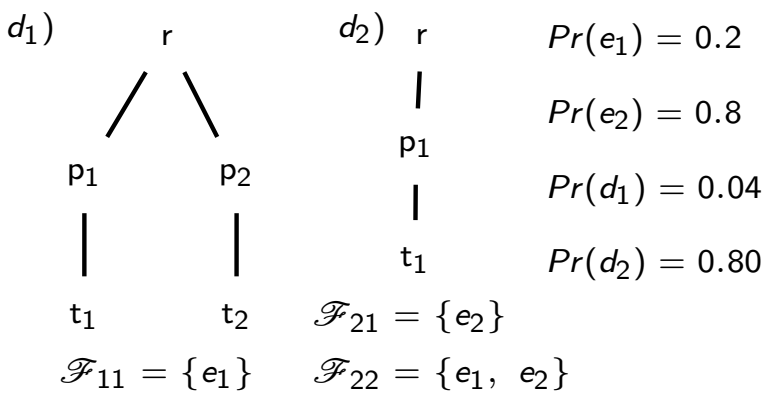
Uncertain Tree-Structured Data

Probabilistic XML [Kimelfeld & Senellart.(2013)]

- Unordered, unranked, and labeled **XML trees** with annotated edges
  - annotations are **propositional formulas** of random Boolean variables



PrXML$^{\text{fie}}$ p-Document



Possible worlds and their probabilities

- Enumerating all possible worlds and their probabilities
- Enable also to model uncertain updates on (uncertain) nodes [Kharlamov et al.(2010)]
- ☞ Integrate such a representation in a typical version control process

Uncertain Multi-Version XML Document

Uncertain Version Control Model

Defines two equivalent views over any uncertain multi-version XML tree

- set $\mathscr{V}$ of random variables $e_0, e_1 \ldots e_n$ modeling the tree states
- infinite set $\mathscr{D}$ of all (unordered) XML trees including the versions

$(\mathscr{G}, \Omega)$: Logical View

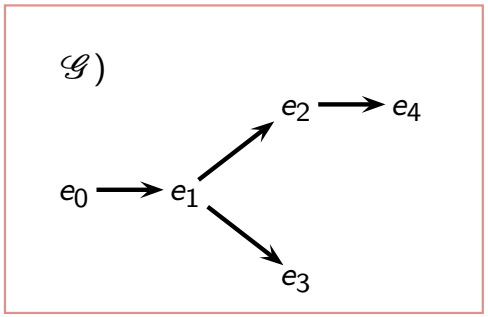
- DAG $\mathscr{G}$ built on variables in $\mathscr{V}$
- Mapping $\Omega : 2^{\mathscr{V} \setminus \{e_0\}} \rightarrow \mathscr{D}$ which computes the possible versions according to sets of valid events

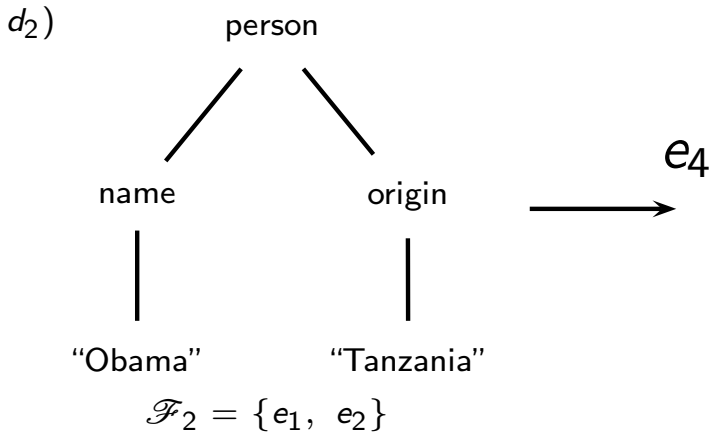
$(\mathscr{G}, \widehat{\mathscr{P}})$: Probabilistic XML Encoding

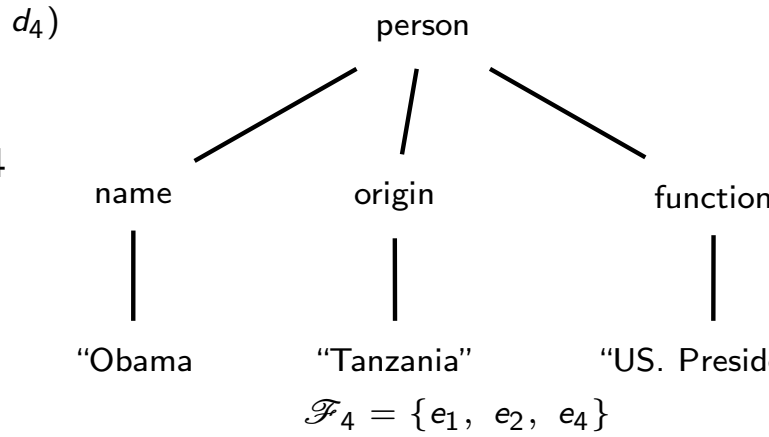
- Similar DAG $\mathscr{G}$ of random variables in $\mathscr{V}$
- Probabilistic XML tree $\widehat{\mathscr{P}}$ which defines the same probability distribution as Ω mapping

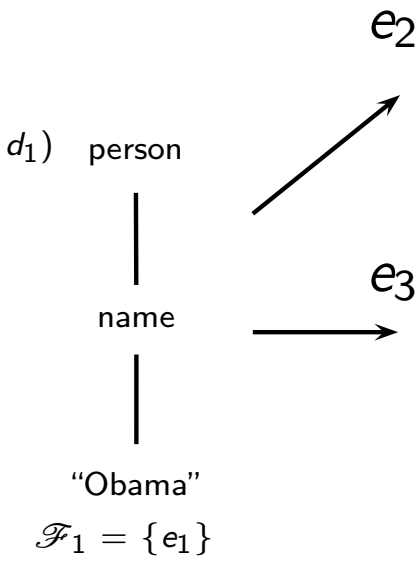
Uncertain Multi-Version XML Document

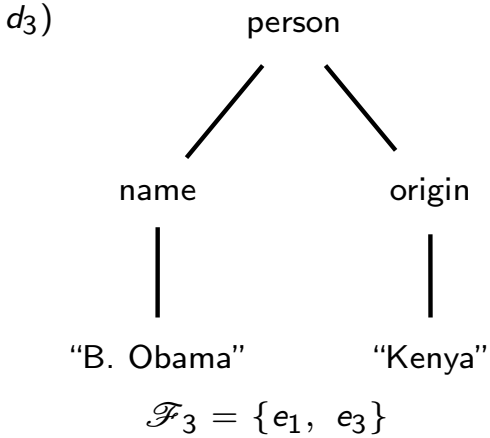
Uncertain Version Control Model (Example)





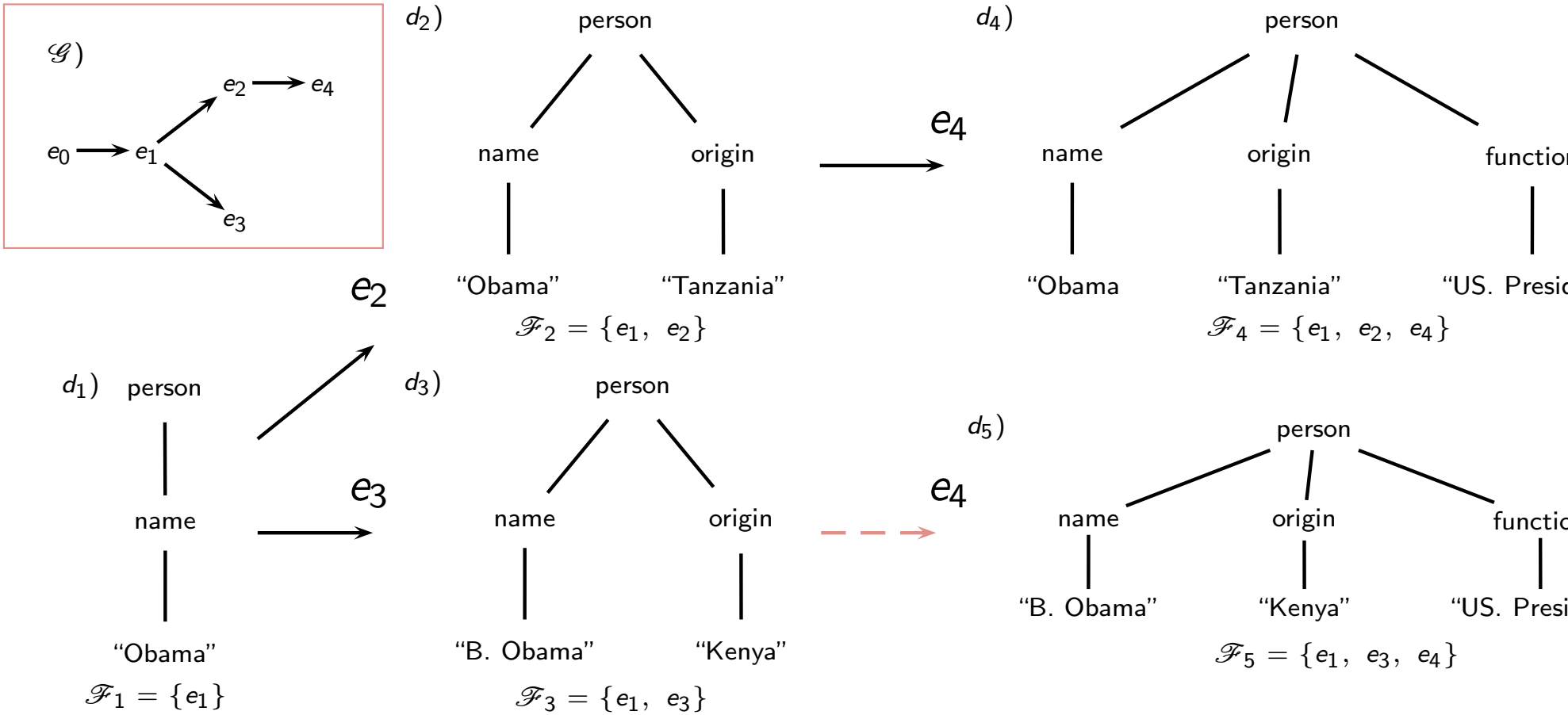






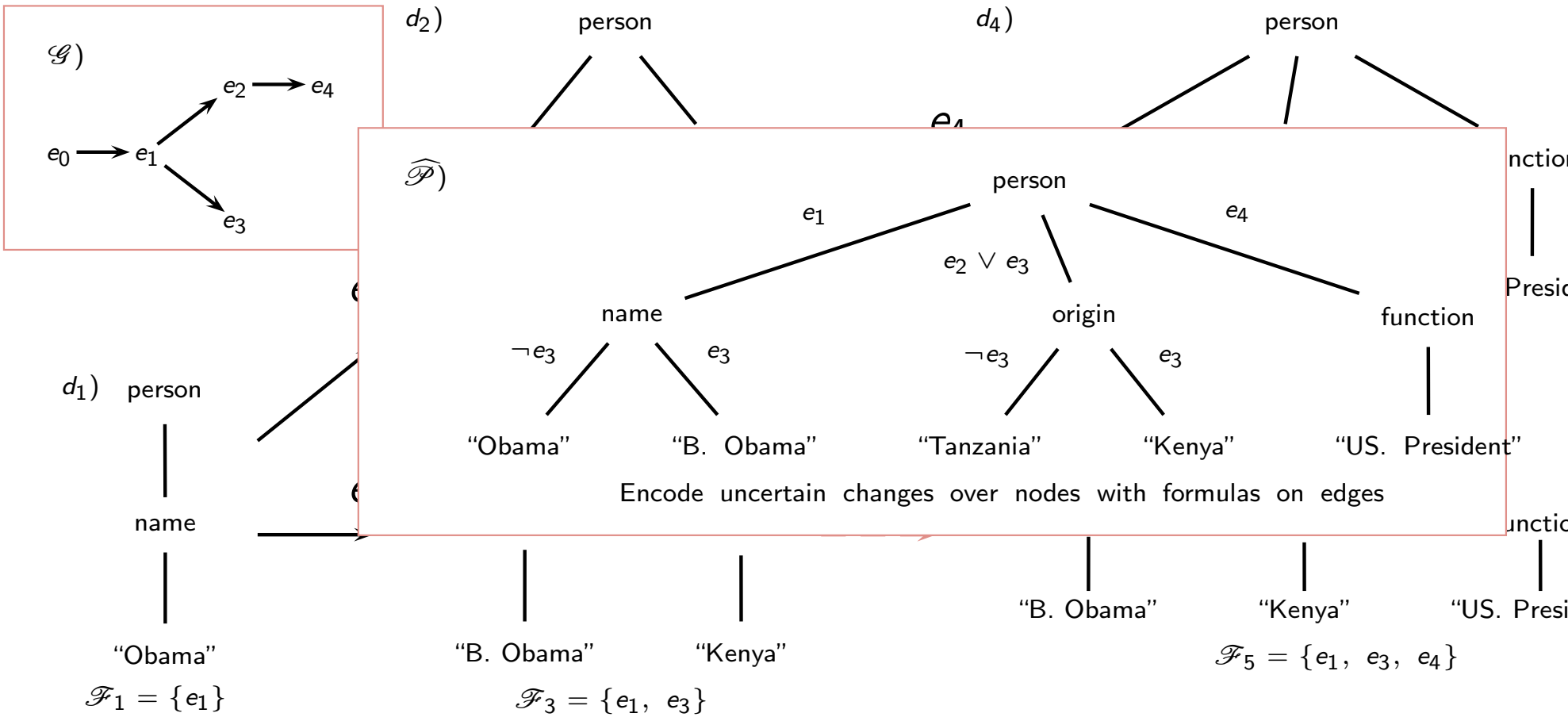
Uncertain Multi-Version XML Document

Uncertain Version Control Model (Example)



Uncertain Multi-Version XML Document

Uncertain Version Control Model (Example)



Uncertain XML Merging Operation

Edit Detection

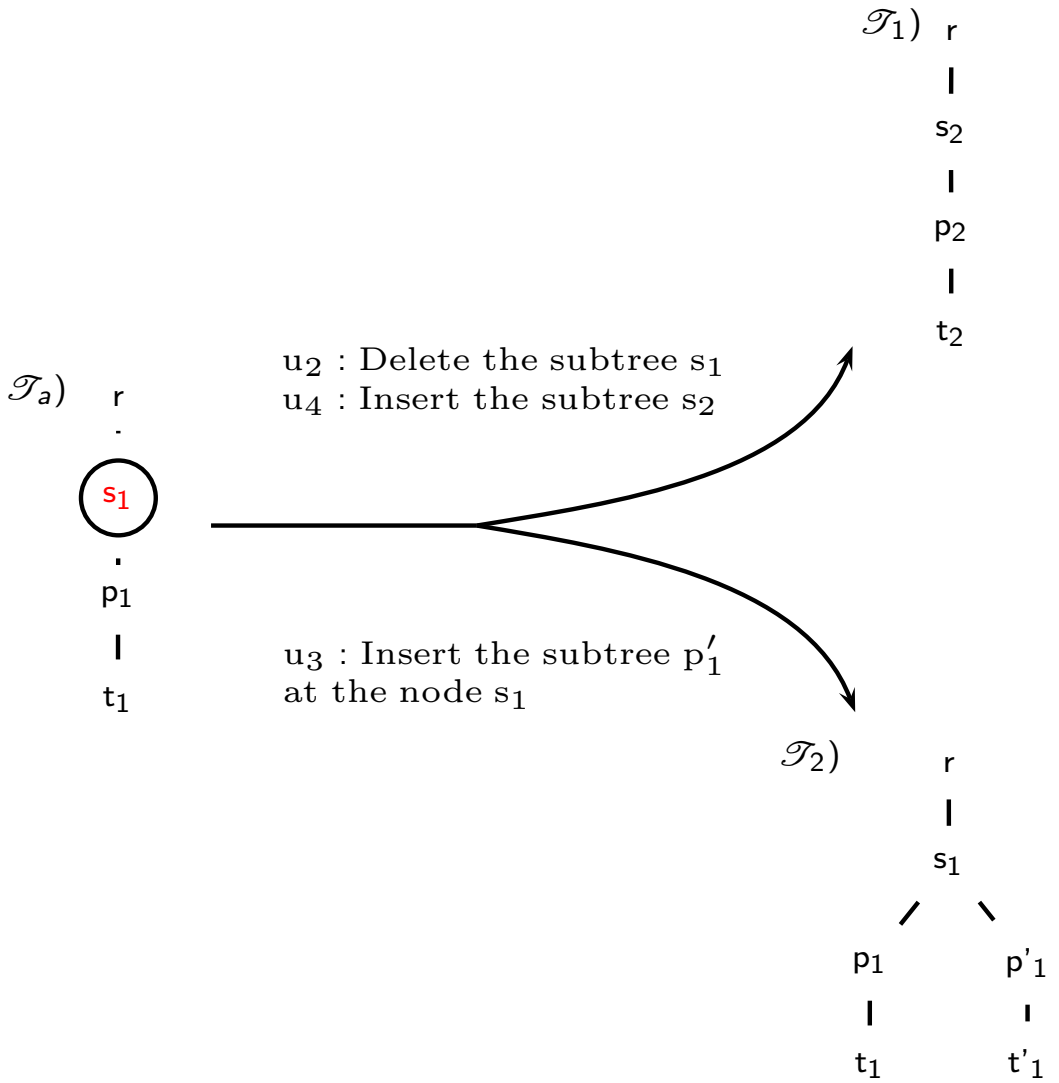
- Three-way procedure *diff3*() detecting node insertions and deletions based on *diff2*() sub-routines

$$diff3(\mathscr{T}_1, \mathscr{T}_2, \mathscr{T}_a) = diff2(\mathscr{T}_a, \mathscr{T}_1) \cup diff2(\mathscr{T}_a, \mathscr{T}_2)$$

- *diff3*() output is an edit script consisting of equivalent, conflicting and independent edits

Uncertain XML Merging Operation

Edit Detection



$diff2(\mathscr{T}_a, \mathscr{T}_1) = \{u_2, u_4\}$
$diff2(\mathscr{T}_a, \mathscr{T}_2) = \{u_3\}$
$diff3(\mathscr{T}_1, \mathscr{T}_2, \mathscr{T}_a) = \{u_2, u_4, u_3\}$

u_2 and u_3 are conflicting edits
u_4 is an independent edit
s_1 is a conflicting node

Uncertain XML Merging Operation

Edit Detection

- Three-way procedure *diff3*() detecting node insertions and deletions based on *diff2*() sub-routines

$$diff3(\mathscr{T}_1, \mathscr{T}_2, \mathscr{T}_a) = diff2(\mathscr{T}_a, \mathscr{T}_1) \cup diff2(\mathscr{T}_a, \mathscr{T}_2)$$

- *diff3*() output is an edit script consisting of equivalent, conflicting and independent edits

- $\Delta^{\mathscr{C}}$ is considered as the restriction of *diff3*() to the set of conflicting edits (over conflicting nodes) for the merging

Uncertain XML Merging Operation

Formal Definition(I)

Given the triple (e_1, e_2, e'), an uncertain merge operation is formalized as $\mathfrak{MRG}_{e_1,e_2,e'}$

- events e_1 and e_2 identify the two (uncertain) versions to be merged
- e' (a merge event) is a new event assessing the amount of uncertainty in the merge

An uncertain merging operation $\mathfrak{MRG}_{e_1,e_2,e'}$ on $\mathscr{T}_{mv}$ maps in a logic sense to the formula below

$$\mathfrak{MRG}_{e_1,e_2,e'}(\mathscr{T}_{mv}) := (\mathscr{G} \cup (\{e'\}, \{(e_1, e'), (e_2, e')\}),\ \Omega').$$

Uncertain XML Merging Operation

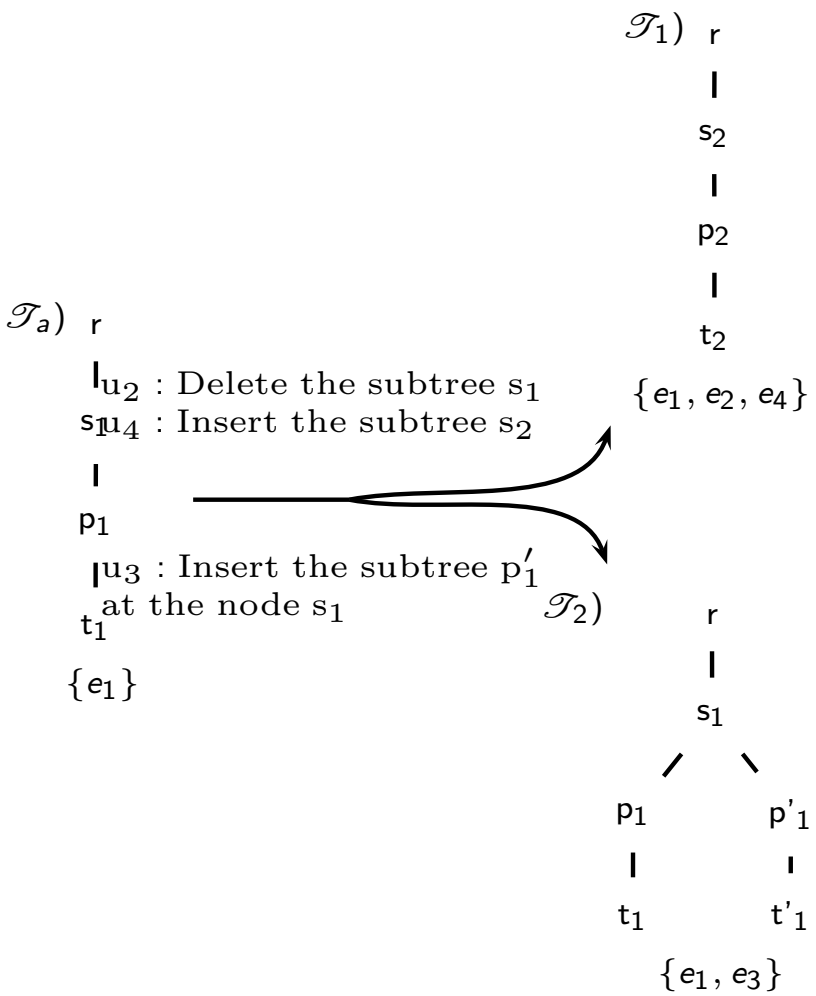
Formal Definition(II)

Let $\mathscr{A}_{e_1} = \{e \mid e \in \mathscr{G},\ e \prec e_1\}$, $\mathscr{A}_{e_2} = \{e \mid e \in \mathscr{G},\ e \prec e_2\}$,
$\mathscr{A}_s = \mathscr{A}_{e_1} \cap \mathscr{A}_{e_2}$
For all subset $\mathscr{F} \in 2^{\mathscr{V} \cup \{e'\}}$, Ω' is computed based on Ω mapping as follows

- if $e' \notin \mathscr{F}$: $\Omega'(\mathscr{F}) := \Omega(\mathscr{F})$;
- if $\{e_1, e_2, e'\} \subseteq \mathscr{F}$: $\Omega'(\mathscr{F}) := \Omega(\mathscr{F} \setminus \{e'\})$;
- if $\{e_1, e'\} \subseteq \mathscr{F} \wedge e_2 \notin \mathscr{F}$: $\Omega'(\mathscr{F}) := [\Omega((\mathscr{F} \setminus \{e'\}) \setminus (\mathscr{A}_{e_2} \setminus \mathscr{A}_s))]^{\Delta_2 - \Delta^{\mathscr{C}}}$;
- if $\{e_2, e'\} \subseteq \mathscr{F} \wedge e_1 \notin \mathscr{F}$: $\Omega'(\mathscr{F}) := [\Omega((\mathscr{F} \setminus \{e'\}) \setminus (\mathscr{A}_{e_1} \setminus \mathscr{A}_s))]^{\Delta_1 - \Delta^{\mathscr{C}}}$;
- if $\{e_1, e_2\} \cap \mathscr{F} = \emptyset \wedge e' \in \mathscr{F}$:
  $\Omega'(\mathscr{F}) := [\Omega((\mathscr{F} \setminus \{e'\}) \setminus ((\mathscr{A}_{e_1} \setminus \mathscr{A}_s) \cup (\mathscr{A}_{e_2} \setminus \mathscr{A}_s)))]^{\Delta_3 - \Delta^{\mathscr{C}}}$;

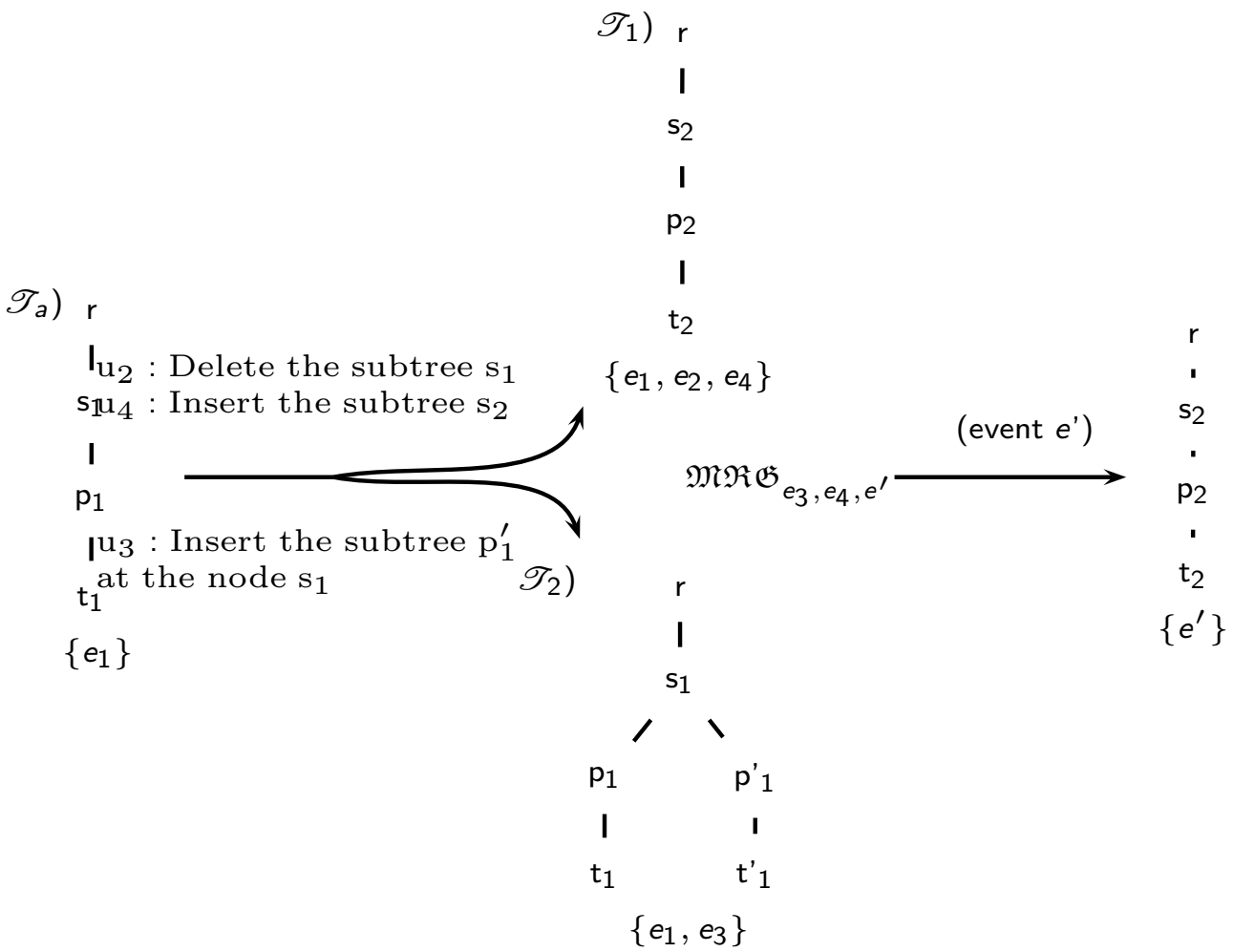
Uncertain XML Merging Operation

Formal Definition(II)



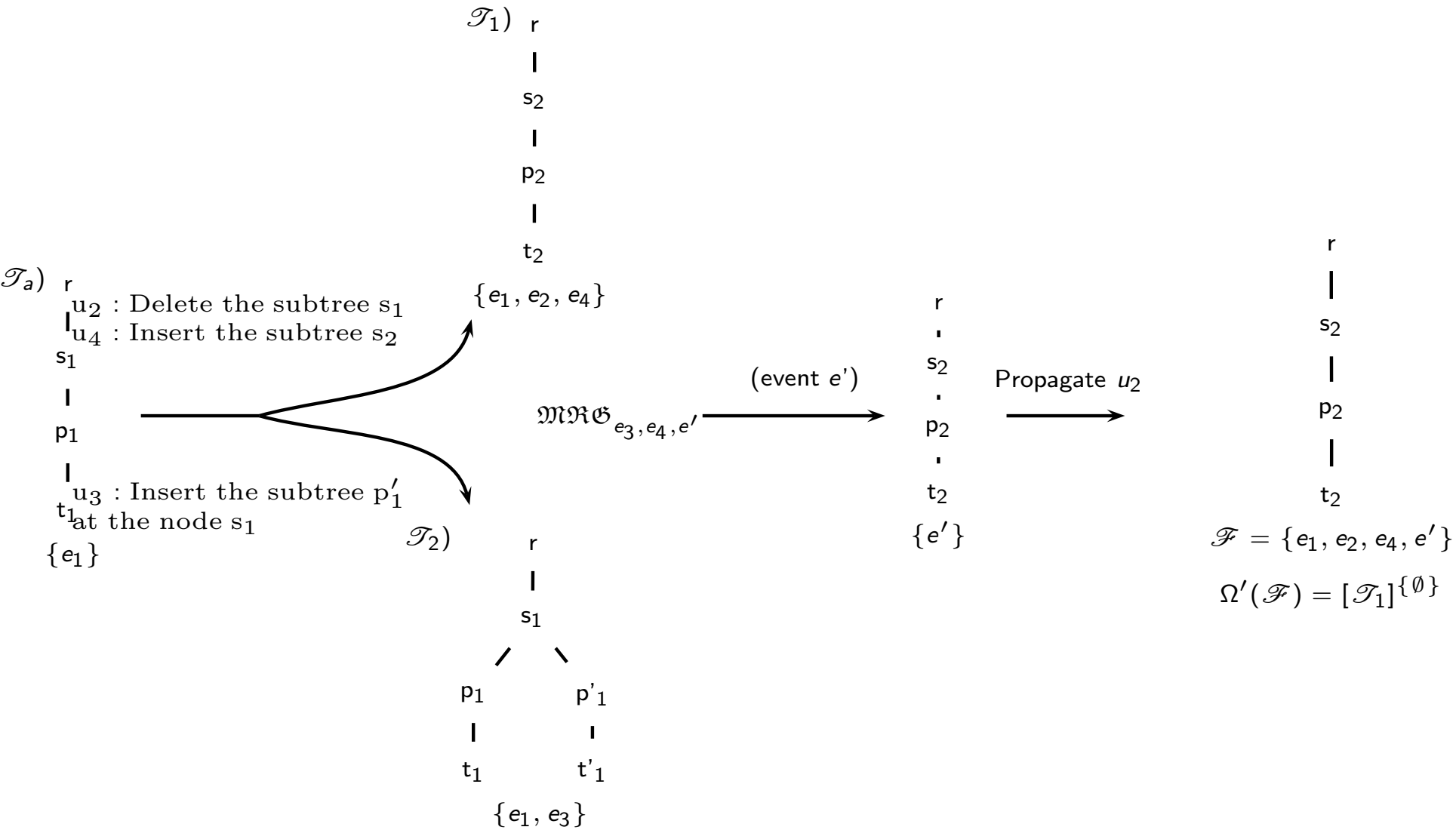
Uncertain XML Merging Operation

Formal Definition(II)



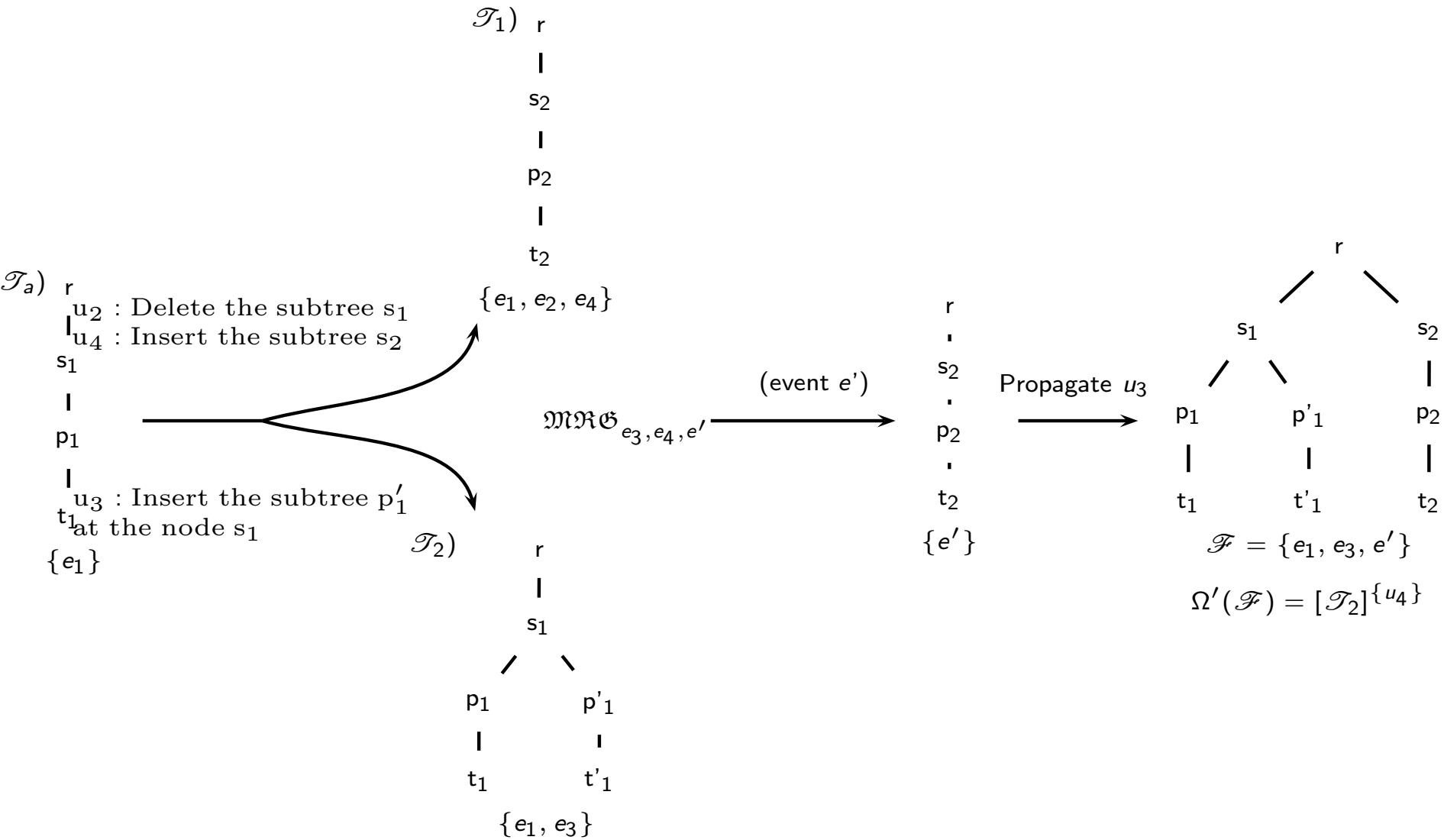
Uncertain XML Merging Operation

Formal Definition(II)



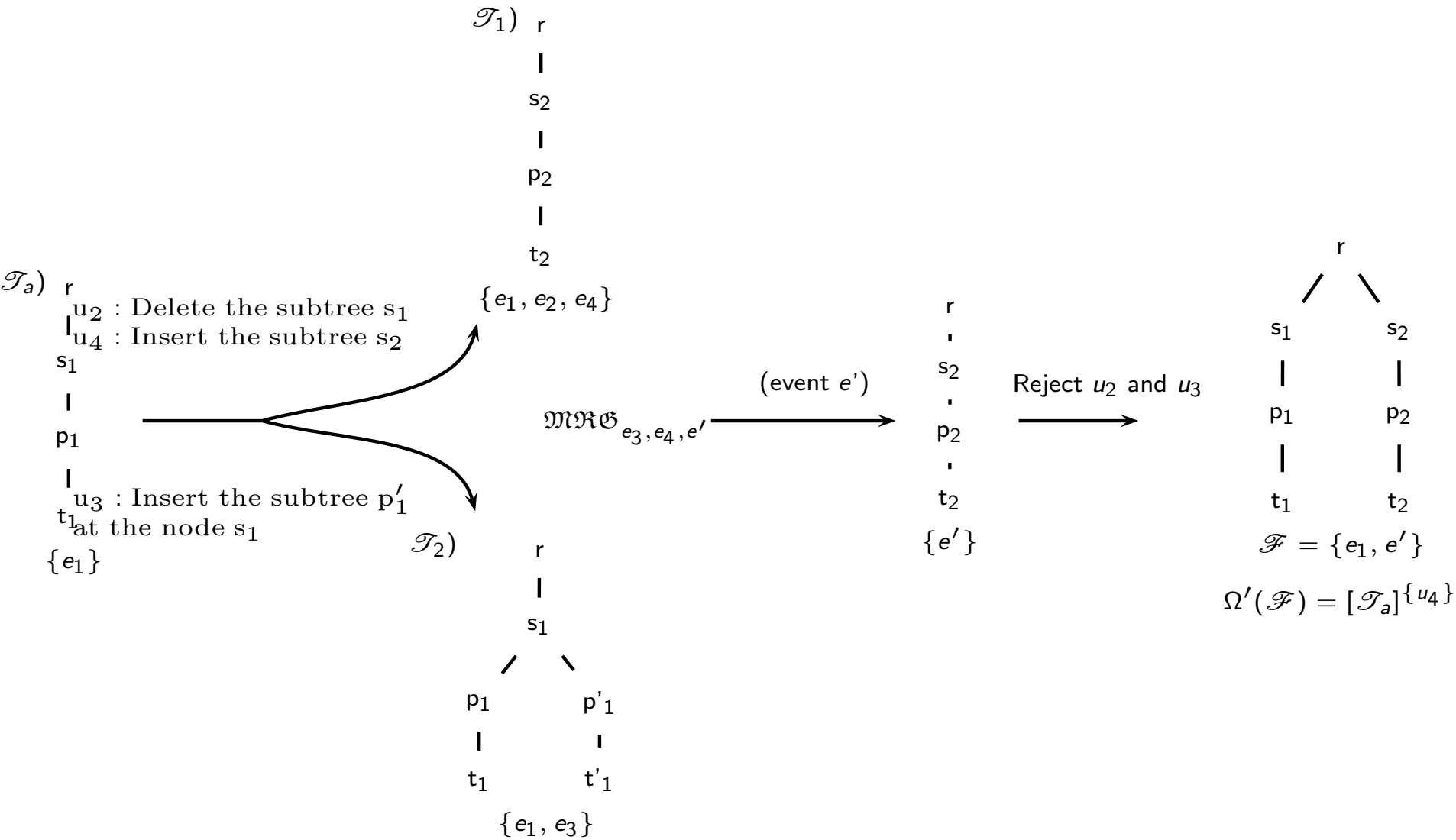
Uncertain XML Merging Operation

Formal Definition(II)



Uncertain XML Merging Operation

Formal Definition(II)



Merging Probabilistic XML (MergePrXML)

Conflicting and Non-conflicting Nodes

- MergePrXML distinguishes conflicting nodes to non-conflicting ones

Under the probabilistic XML Encoding $\widehat{\mathscr{T}_{mv}}$, a given x in $\widehat{\mathscr{P}}$ is a conflicted node with respect to $\mathfrak{MRG}_{e_1,e_2,e'}$ when its lineage $fie^{\uparrow}(x)$ is such that –

1. $fie^{\uparrow}(x) \models \nu_s$;
2. $fie^{\uparrow}(x) \not\models \nu_1$ (or $fie^{\uparrow}(x) \not\models \nu_2$) and;
3. $\exists y \in \widehat{\mathscr{P}}$, $\mathtt{desc}(x, y)$: $fie^{\uparrow}(y) \not\models \nu_s$ and $fie^{\uparrow}(y) \models \nu_2$ (or $fie^{\uparrow}(y) \models \nu_1$)

- $fie^{\uparrow}(x) = \bigwedge_{z \in \widehat{\mathscr{P}},\ z \preceq x}(fie(z))$ and ν_s, ν_1, ν_2 are valuations over $\mathscr{A}_s$, $\mathscr{A}_{e_1}$, $\mathscr{A}_{e_2}$ respectively
- MergePrXML implements $\mathfrak{MRG}_{e_1,e_2,e'}$ as an update operation in $\widehat{\mathscr{P}}$ that only modifies formulas of non-conflicting nodes of the merge

Merging Probabilistic XML (MergePrXML)

Merge Algorithm (I)

Input: $(\mathscr{G}, \widehat{\mathscr{P}})$, e_1, e_2, e'

Output: Merging Uncertain XML Versions in $\widehat{\mathscr{T}_{mv}}$

$\mathscr{G} := \mathscr{G} \cup (\{e'\}, \{(e_1, e'), (e_2, e')\})$;

foreach *non-conflicted node x in* $\widehat{\mathscr{P}} \setminus \widehat{\mathscr{P}}_{|\mathscr{C}_{\{e_1,\ e_2\}}}$ **do**

  `replace`$(fie(x),\ e_1,\ (e_1 \vee e'))$;

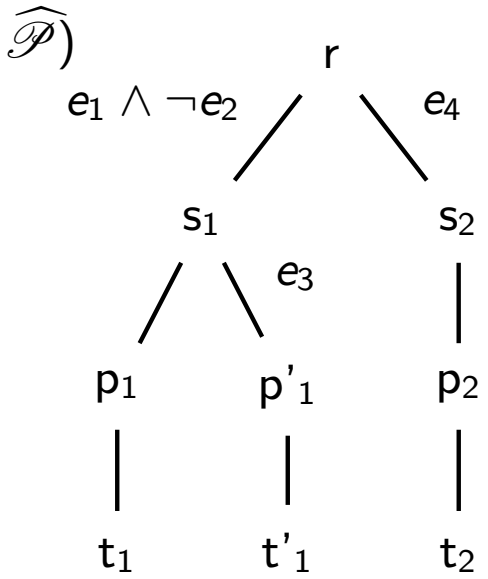
  `replace`$(fie(x),\ e_2,\ (e_2 \vee e'))$;

return $(\mathscr{G}, \widehat{\mathscr{P}})$

✉ MergePrXML performs the merge in time proportional to the size of the formulas of nodes impacted by the updates in merged branches

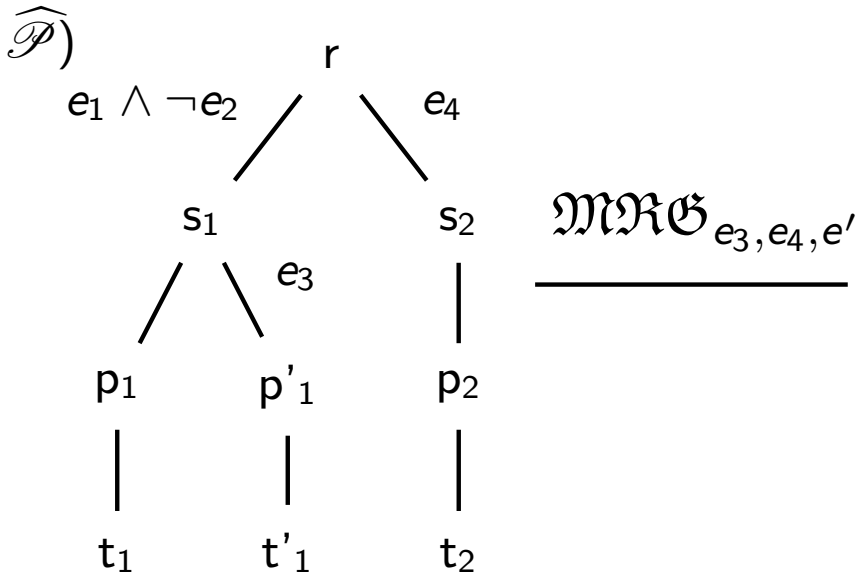
Merging Probabilistic XML (MergePrXML)

Merge Algorithm (II)



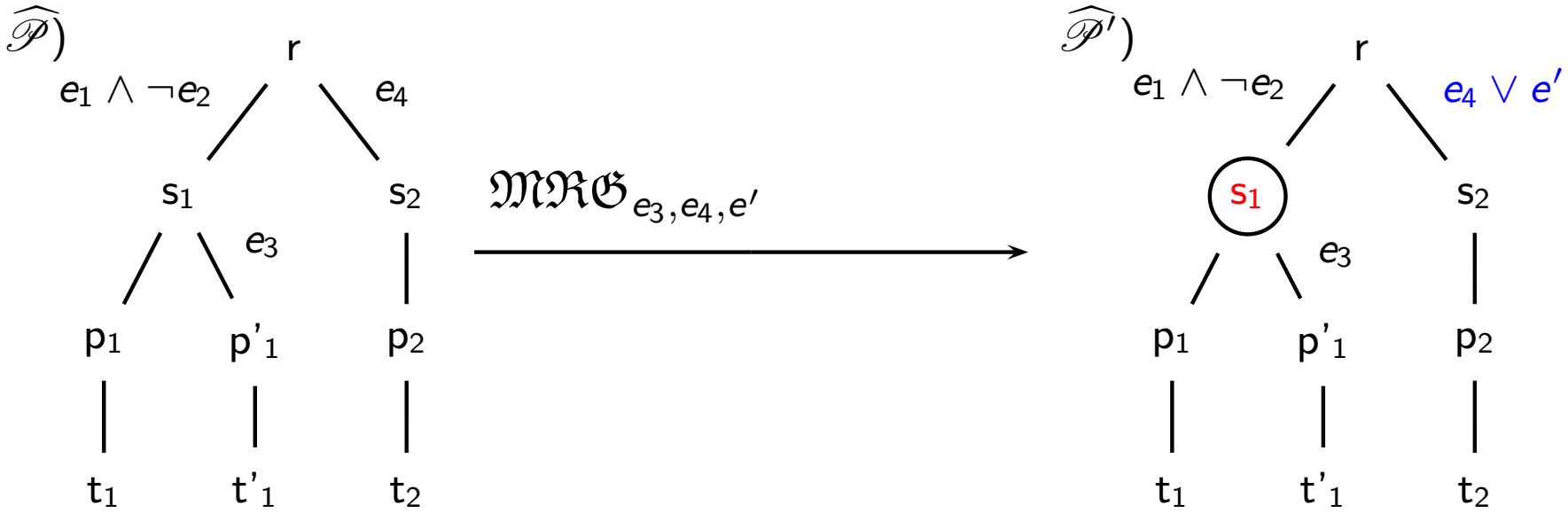
Merging Probabilistic XML (MergePrXML)

Merge Algorithm (II)



Merging Probabilistic XML (MergePrXML)

Merge Algorithm (II)



Conclusion and Further Works

- Merging operation over tree-structured multi-version documents with uncertain data
  - implementation of the common deterministic merge scenarios
  - modelling of the amount of uncertainty in the merged versions
- Efficient merging algorithm over Probabilistic XML encoding

References I

Abdessalem, T., Ba, M. L., and Senellart, P. (2011).
A probabilistic XML merging tool.
In *Proc. EDBT*.

Ba, M. L., Abdessalem, T., and Senellart, P. (2011).
Towards a version control model with uncertain data.
In *PIKM*.

Ba, M. L., Abdessalem, T., and Senellart, P. (2013).
Uncertain version control in open collaborative editing of tree-structured documents.
In *Proc. DocEng*.

Kharlamov, E., Nutt, W., and Senellart, P. (2010).
Updating Probabilistic XML.
In *Proc. Updates in XML*.

Kimelfeld, B. and Senellart, P. (2013).
Probabilistic XML: Models and Complexity.
In *Advances in Probabilistic Databases for Uncertain Information Management*. Springer-Verlag.

Lindholm, T. (2004).
A three-way merge for XML documents.
In *Proc. DocEng*.

Lindholm, T., Kangasharju, J., and Tarkoma, S. (2006).
Fast and simple XML tree differencing by sequence alignment.
In *Proc. DocEng*.

References II

Ma, J., Liu, W., Hunter, A., and Zhang, W. (2010).

An XML based framework for merging incomplete and inconsistent statistical information from clinical trials.
In Ma, Z. and Yan, L., editors, *Software Computing in XML Data Management*. Springer-Verlag.

Suzuki, N. (2002).

A Structural Merging Algorithm for XML Documents.
In *Proc. ICWI*.