



Boolean Constraints for XML Modeling

Sven Hartmann

*Department of Informatics
Clausthal University of Technology
Germany*

Sebastian Link

*School of Information Management
Victoria University of Wellington
New Zealand*

Thu Trinh

*School of Engineering and Advanced Technology
Massey University
New Zealand*



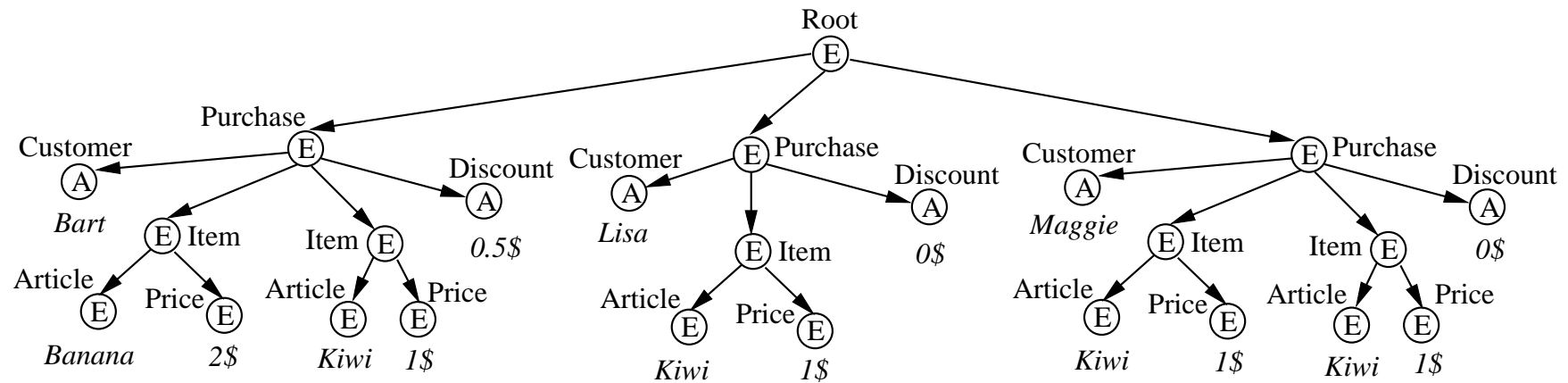
Motivation

- XML: de-facto standard for Web data exchange and integration
- high degree of syntactic flexibility, low degree of semantic capabilities
- challenge for computer scientists: provide full-fledged tools that can store, manage and process XML data in its native format
 - S. Abiteboul, P. Buneman, D. Suciu. *Data on the Web: From relations to semistructured data and XML*. Morgan Kaufmann, 1999.
 - S. Ceri, P. Fraternali, S. Paraboschi. *XML: Current developments and future challenges for the database community*. EDBT, 3-17, 2000.
 - W. Fan. *XML constraints*. DEXA Workshops, 805-809, 2005.
 - D. Suciu. *On database theory and XML*. SIGMOD Record, 30(3):39-45, 2001.
 - V. Vianu. *A web odyssey: from Codd to XML*. PODS, 1-15, 2001.
 - J. Widom. *Data management for XML: Research direction*. Data Engineering Bulletin, 22(3):44-52, 1999.
- integrity constraints enhance semantic capabilities:
 - find natural classes of constraints that can be maintained efficiently
 - balance trade-off between expressiveness and efficiency

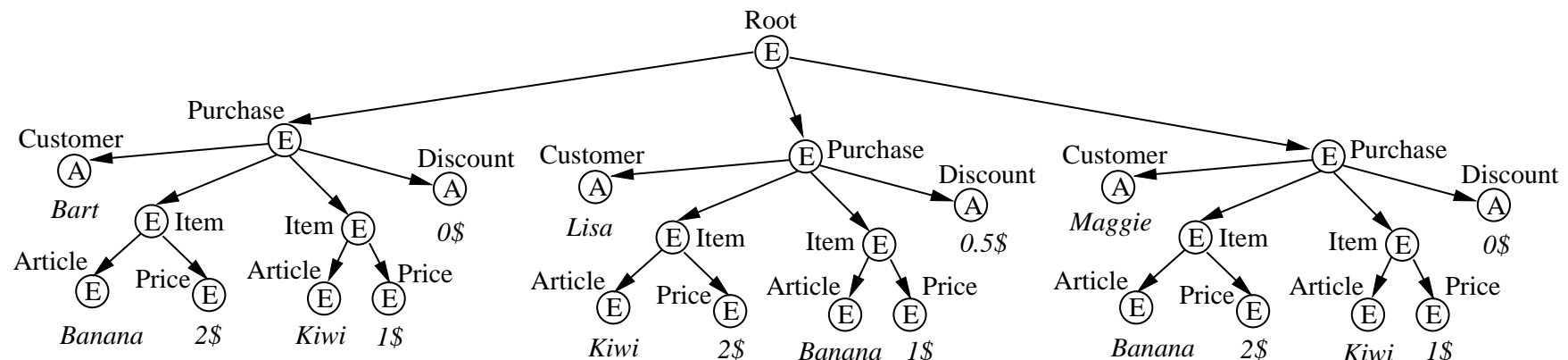


Different Concepts of XML Functional Dependencies

- the same article has the same price (subject in the literature):



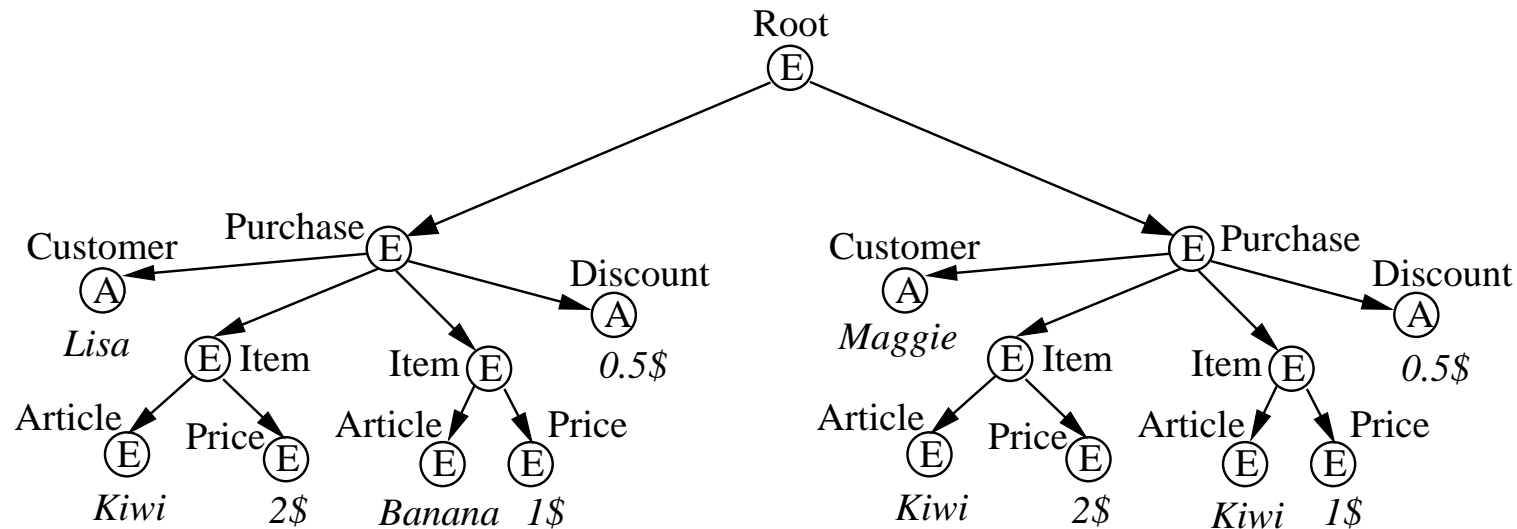
- whenever same items are bought, the same discount applies:





A fundamental Problem

- implicit knowledge applicable to modeling, designing, processing
- express (much) AND reason (efficiently) about domain knowledge

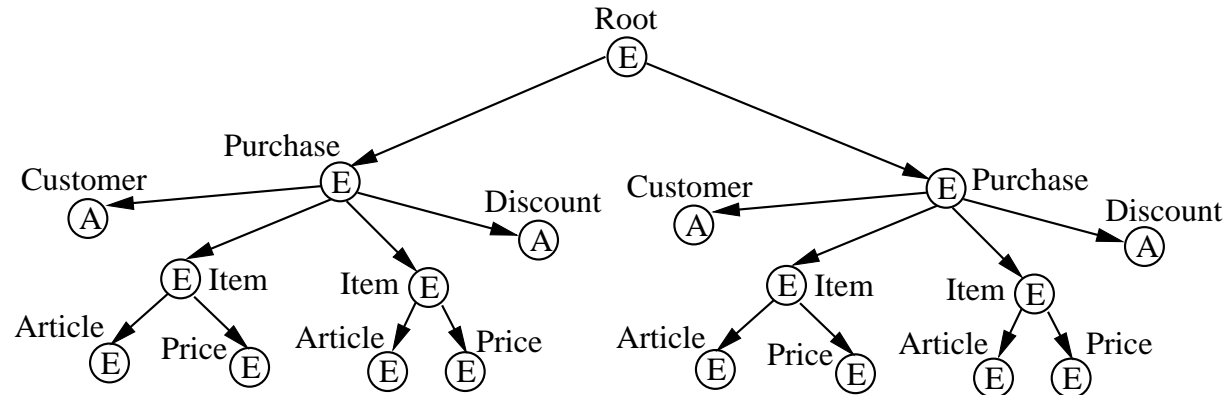
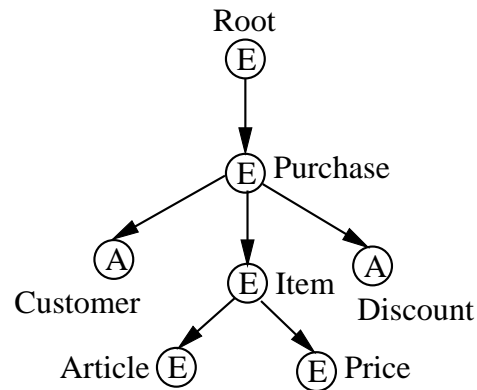


- Rule 1: the same customer receives the same discount (loyalty)
- Rule 2: customers that buy all the same items receive the same discount
- Rule 3: when same discount applied, then purchases belong to same customer or consist of the same items



XML Trees, Data Trees and Schema Trees

- *XML trees*: node-labelled tree T where *kind* labels node as element or attribute, and *name* gives node an element- or attribute-name
- *XML data tree*: leaf nodes obtain string value by *val*
- *XML schema tree*: no vertex has two successors of same kind and name, and *freq* labels each arc with frequency in $\{?, 1, *, +\}$
 - arcs terminating in attribute nodes have frequency ? or 1





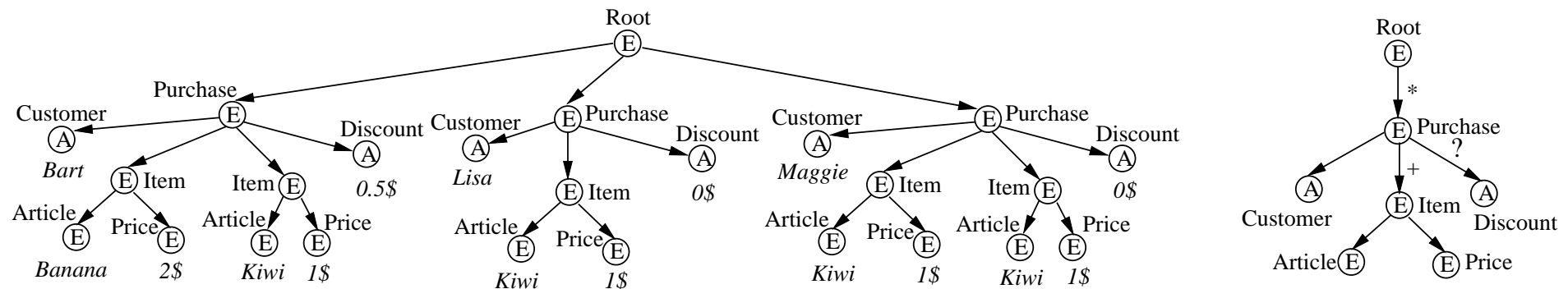
Homomorphisms and Isomorphisms

- homomorphism ϕ maps T' -vertices to T -vertices such that
 - T' -arcs (v', w') become T -arcs $(\phi(v'), \phi(w'))$,
 - *root-preserving*,
 - *kind-preserving*,
 - *name-preserving*.
- homomorphism ϕ is *isomorphism* if
 - ϕ bijective and
 - ϕ^{-1} homomorphism
- isomorphic data trees T' and T are *equivalent* (or copies of one another) if the isomorphism is *evaluation-preserving*



Compatibility

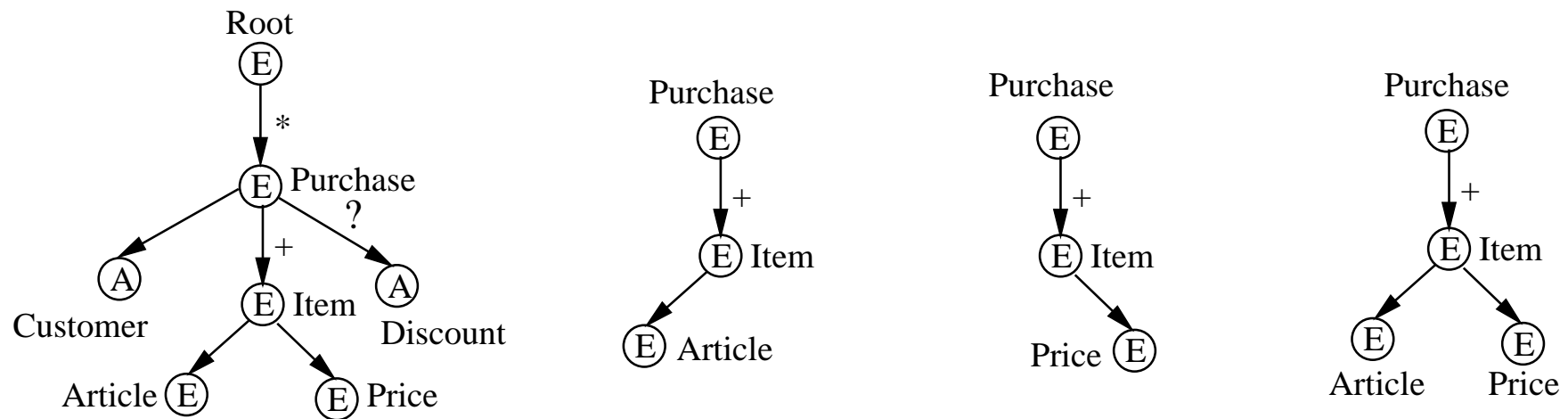
- T' is *compatible* with T if there is a homomorphism $\phi : V_{T'} \rightarrow V_T$ such that for each vertex v' of T' and each arc $a = (\phi(v'), w)$ of T , the number of arcs $a' = (v', w'_i)$ mapped to a is
 - at most 1 if $freq(a) = ?$,
 - exactly 1 if $freq(a) = 1$,
 - at least 1 if $freq(a) = +$,
 - and arbitrarily many if $freq(a) = *$





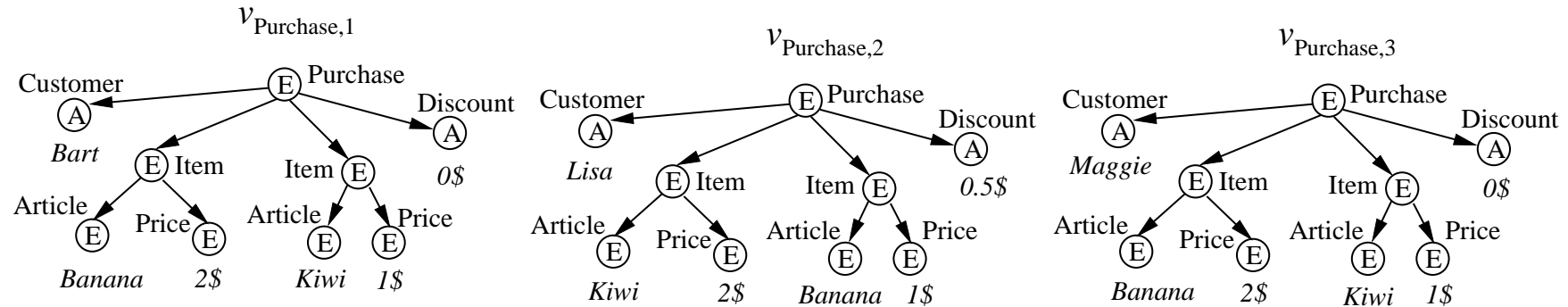
Subgraphs

- $Sub_T(v)$: the v -subgraphs of T (unions of dipaths from v to a leaf)
- v -subgraph U denoted by $[[l_1, \dots, l_k]]$ with leaves l_i of U
- total v -subgraph $T(v)$, empty v -subgraph $[[\emptyset]]$
- projection $T' \upharpoonright_U$ of data tree T' to r_T -subgraph U of schema tree T : union of all copies of some r_T -subgraph of U in T'





Which v -subgraphs identify Pre-Images up to Equivalence?



- the projection on $\llbracket \text{Article} \rrbracket$ and the projection on $\llbracket \text{Price} \rrbracket$ do not allow us to distinguish between the second and the third $v_{Purchase}$ pre-image

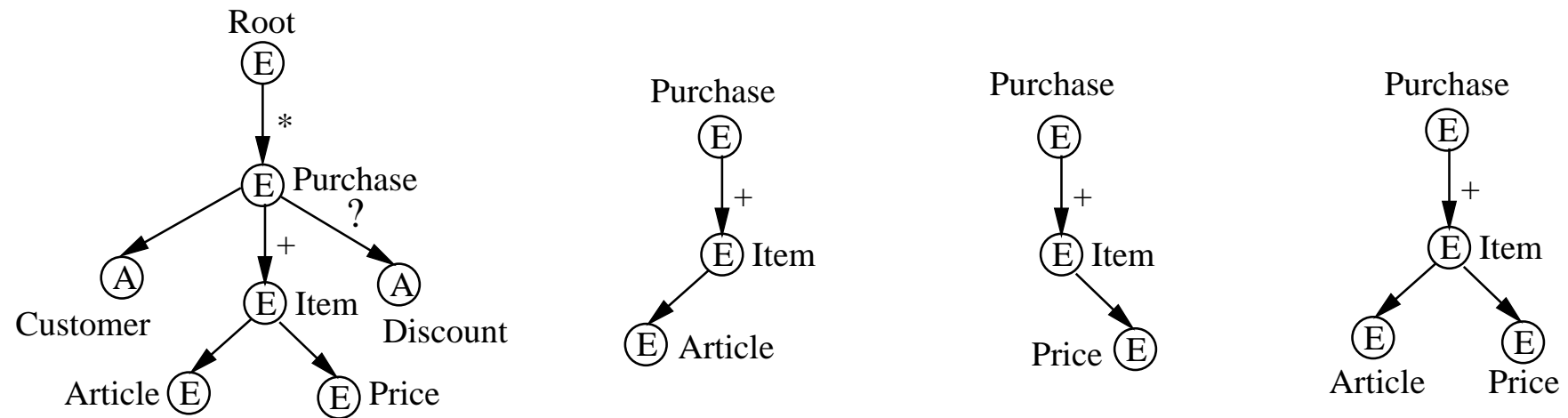


Essential v -Subgraphs

- $X, Y \in Sub_T(v)$ are *reconcilable* iff there are v -subgraphs X' of X and Y' of Y such that
 - X' and Y' share no arc of $*$, $+$ -frequency and
 - $X' \sqcup Y' = X \sqcup Y$ holds
- X and Y not reconcilable iff we can find T -compatible data tree T' and two pre-images W, W' of $T(v)$ in T' such that $W|_X$ is equivalent to $W'|_X$, $W|_Y$ is equivalent to $W'|_Y$, but $W|_{X \sqcup Y}$ is not equivalent to $W'|_{X \sqcup Y}$
- *essential subgraphs*: smallest set $\mathcal{E}(v) \subseteq Sub_T(v)$ such that
 - all unary v -subgraphs are in $\mathcal{E}(v)$,
 - if $X, Y \in \mathcal{E}(v)$ are not reconcilable, then $X \sqcup Y \in \mathcal{E}(v)$



Example



[[Customer]], [[Article]], [[Price]], [[Discount]], and [[Article,Price]] from the essential $v_{Purchase}$ -subgraphs



Boolean Constraints

- Let T be an XML schema tree, and $v \in V_T$ a vertex of T . The set of *Boolean constraints* over the vertex v is defined as the smallest set $BC(v)$ with the following properties:
 - if $X \in \mathcal{E}(v)$, then $v : X \in BC(v)$,
 - if $v : \varphi \in BC(v)$, then $v : \neg\varphi \in BC(v)$, and
 - if $v : \varphi, v : \psi \in BC(v)$, then $v : (\varphi \wedge \psi) \in BC(v)$.
- Examples:
 - $\varphi_1 = v_{\text{Purchase}} : \llbracket \text{Customer} \rrbracket$,
 - $\varphi_2 = v_{\text{Purchase}} : \llbracket \text{Customer} \rrbracket \vee (\llbracket \text{Article} \rrbracket \wedge \llbracket \text{Price} \rrbracket)$,
 - $\varphi_3 = v_{\text{Purchase}} : \llbracket \text{Customer} \rrbracket \vee \llbracket \text{Article, Price} \rrbracket$,
 - $\varphi_4 = v_{\text{Purchase}} : \llbracket \text{Article, Price} \rrbracket \Rightarrow \llbracket \text{Discount} \rrbracket$, and
 - $\varphi_5 = v_{\text{Purchase}} : \llbracket \text{Discount} \rrbracket \Rightarrow (\llbracket \text{Customer} \rrbracket \vee \llbracket \text{Article, Price} \rrbracket)$.



Satisfaction of Boolean Constraints

- T schema tree, $v \in V_T$, T' is T -compatible data tree
- distinct pre-images W_1, W_2 of $T(v)$ in T' satisfy φ over v :
 - if $\varphi = v : X$, then $W_1|_X$ and $W_2|_X$ are equivalent,
 - if $\varphi = v : \neg\psi$, then W_1, W_2 do not satisfy $v : \psi$,
 - if $\varphi = v : (\psi_1 \wedge \psi_2)$, then W_1, W_2 satisfy both $v : \psi_1$ and $v : \psi_2$
- T' satisfies φ over v iff for all distinct pre-images W_1, W_2 of $T(v)$ in T' we have that W_1, W_2 satisfy φ



Correspondence to Propositional Logic

- $\tau : \mathcal{E}(v) \rightarrow \mathcal{V}$ bijection where \mathcal{V} set of propositional variables
- extend to $\tau : BC(v) \rightarrow \mathbb{F}_{\mathcal{V}}$ with $\varphi \mapsto \varphi'$ via:
 - if $\varphi = X \in \mathcal{E}(v)$, then $\varphi' = \tau(X)$
 - for $\varphi = \neg\psi$ we have $\varphi' = \neg\psi'$, and
 - for $\varphi = (\psi_1 \wedge \psi_2)$ we have $\varphi' = (\psi'_1 \wedge \psi'_2)$
- $\Sigma' = \{\sigma' \mid \sigma \in \Sigma\}$
- $\Sigma'_v = \{\tau(X) \Rightarrow \tau(Y) \mid X, Y \in \mathcal{E}(v), X \text{ covers}^1 Y\}$
- Equivalent are:
 - (i) Σ implies φ ,
 - (ii) $\Sigma' \cup \Sigma'_v$ logically implies φ'



The Example from a Logical Perspective

- define $\tau : \mathcal{E}(v_{\text{Purchase}}) \rightarrow \mathcal{V}$ by
 - $\tau(\llbracket \text{Customer} \rrbracket) = V_1$, $\tau(\llbracket \text{Article} \rrbracket) = V_2$,
 - $\tau(\llbracket \text{Price} \rrbracket) = V_3$, $\tau(\llbracket \text{Discount} \rrbracket) = V_4$, and
 - $\tau(\llbracket \text{Article}, \text{Price} \rrbracket) = V_5$.
- as propositional formulae we obtain:
 - $\Sigma' = \{V_1 \Rightarrow V_4, V_5 \Rightarrow V_4\}$,
 - $\Sigma'_{v_{\text{Purchase}}} = \{V_5 \Rightarrow V_2, V_5 \Rightarrow V_3\}$, and
 - $\varphi' = V_4 \Rightarrow (V_1 \vee V_5)$.
- the truth assignment θ defined by $\theta(V_i) = 1$ if and only if $i \in \{2, 3, 4\}$ satisfies all the formulae in $\Sigma' \cup \Sigma'_{v_{\text{Purchase}}}$, but does not satisfy φ'



Reasoning about Boolean Constraints

- two pre-images in T' agree on projections to precisely those essential subgraphs whose corresponding propositional variable V is assigned the truth value 1 by θ
- *coNP*-complete in general
- however: off-the-shelf SAT solvers applicable
- several tractable subclasses can be identified:
 - functional dependencies
 - degenerated multivalued dependencies
 - 2-literal constraints



Conclusion and Future Work

- introduced the class of Boolean constraints into XML
- based on homomorphisms between XML schema and data trees
- justified their definition by demonstrating which v -subgraphs of XML schema tree T determine pre-images of $T(v)$ up to equivalence
- capture propositional reasoning about a fixed schema node
- Boolean constraints over different vertices v ?
- multiset/set/list semantics?
- redundancies and update anomalies?