TREX: DTD-Conforming XML to XML Transformations

Aoying Zhou, Qing Wang, Zhimao Guo, Xueqing Gong, Shihui Zheng
Hongwei Wu, Jianchang Xiao, Kun Yue
Fudan University, China
{ayzhou,qingwang,zmguo,gongxq,shzheng0,hwwu,jcxiao,kuny}@fudan.edu.cn

Wenfei Fan
Bell Laboratories, USA
wenfei@research.bell-labs.com

1. Overview

There have been increasing demands for a system to support *DTD-conforming XML to XML transformations:* given any target DTD D and a source XML document S, extract data from S and construct a target XML document T such that T conforms to the predefined D. The need for this is evident in, *e.g.,* data exchange, security views, and data integration. Popular XML query languages (*e.g.,*XQuery, XSLT) cannot guarantee DTD conformance in XML to XML transformations. Type inference and (static) checking are too expensive to be used in practice; worse still, they provide no guidance for how to ensure DTD-conformance.

In response to the need we propose *TREX* (TRansformation Engine for XML), a middleware system for DTD-conforming XML to XML transformations. TREX is based on the novel notion of *XTG* (XML Transformation Grammar), which extends a DTD by incorporating XML queries into element type definitions. This allows one to specify how to extract relevant data from a source XML document via the queries, and to construct a target XML document directed by the embedded DTD. TREX efficiently evaluates XTGs by implementing several optimization techniques. XTGs and TREX provide the first systematic method and practical system to support DTD-conforming XML transformations.

2. Features of TREX

XTG and TREX have a number of salient features (see [1] for detailed discussions).

DTD-conformant specifications. With XTGs one can easily specify XML to XML transformations. The queries associated with an element type definition (DTD grammar rule) find relevant source data to generate the children of a target element, and the grammar rule is enforced when the children are created to guarantee DTD-conformance.

Data-driven semantics. XTGs are capable of handling complex DTDs. During the generation of a target document, the decisions on the choice of a *nondeterministic* (disjuncion) grammar rule and on the expansion of the target XML tree in the *recursive* case are based on the source data.

Batch and lazy modes. TREX is quite flexible by offering two evaluation modes. In the *batch* mode, it generates a complete XML document. In the *lazy* mode, it constructs a partial XML (DOM) tree, interacts with users, and expands the tree based on users' requests and interests.

Query composition, tuning and caching. TREX employs several optimization techniques: composing related XML queries, simplifying XPath expressions, and caching query results to avoid unnecessary recomputation. With these TREX efficiently support XTGs.

Although TREX is similar in spirit to the XML publishing system for relational data reported in [2], it conquers new challenges in the context of XML to XML transformations, at both the conceptual level and the implementation level.

3. System Demonstration

A prototype of TREX has been implemented in Java, using Kweelt [4] as the underlying engine for XML queries. We choose Quilt [3] to express XML queries in XTGs rather than XQuery/XSLT because we could access the source code of Kweelt to incorporate our optimization algorithms. With the prototype, the demonstration is to show how to transform DBLP publication data to an XML document that conforms to a predefined recursive and nondeterministic DTD, in both batch and lazy modes. This verifies the effectiveness of our specification and optimization techniques.

Acknowledgment: This work is supported by NSFC under Grant 60228006. Aoying Zhou is also supported by National 973 Program of China under Grant TG1998030404, and Wenfei Fan is supported in part by NSF Career Award IIS-0093168.

4. References

[1] Full version. http://www.bell-labs.com/user/wenfei/papers/sigmod03-demo.pdf.

[2] M. Benedikt, C. Y. Chan, W. Fan, R. Rastogi, S. Zheng, and A. Zhou. DTD-directed publishing with attribute translation grammars. In *VLDB*, 2002.

[3] D. Chamberlin, J. Robie, and D. Florescu. Quilt: An XML query language for heterogeneous data sources. In *WebDB*, 2000.

[4] SourceForge. Kweelt. http://kweelt.sourceforge.net.

SIGMOD 2003, June 9-12, 2003, San Diego, CA.
Copyright 2003 ACM 1-58113-634-X/03/06 ...$5.00.