Chapter 16
Document and Schema XML Updates

Dario Colazzo
Laboratoire de Recherche en Informatique (LRI-CNRS), Université de Paris-Sud, France

Giovanna Guerrini
DISI – Università degli Studi di Genova, Italy

Marco Mesiti
DICo – Università degli Studi di Milano, Italy

Barbara Oliboni
DI – Università degli Studi di Verona, Italy

Emmanuel Waller
Laboratoire de Recherche en Informatique (LRI-CNRS), Universite de Paris-Sud, France

ABSTRACT

Purpose of this chapter is to describe the different research proposals and the facilities of main enabled and native XML DBMSs to handle XML updates at document and schema level, and their versions. Specifically, the chapter will provide a review of various proposals for XML document updates, their different semantics and their handling of update sequences, with a focus on the XQuery Update proposal. Approaches and specific issues concerned with schema updates will then be reviewed. Document and schema versioning will be considered. Finally, a review of the degree and limitations of update support in existing DBMSs will be discussed.

INTRODUCTION

XML (W3C, 1998) is nowadays everywhere employed for the representation and exchange of information on the Web. The document structure is often described through a DTD or an XML schema (W3C, 2004). Both XML native (e.g., Tamino, eXist, TIMBER) and enabled (e.g., Oracle, IBM DB2, SQL Server) DBMSs have been so far proposed (Bourret, 2007) for storing and querying XML documents. Some enabled DBMSs support XML schema for the specification of a mapping between the XML schema and internal relational or object-relational representation of XML documents (Florescu &

Kossman, 1999). Relevant, from many points of view, is the use of XML schemas in data management. They can be used for the optimization of query execution, for the specification of access control policies and indexing structures.

Despite the high dynamicity of the contexts where XML documents are employed, XML updates have received less attention than XML queries and the W3C proposal for XML document updates has appeared as a recommendation only recently (W3C, 2008). Different alternative proposals for XML updates have been elaborated in the meantime; among the most recent let us mention XQuery! (Ghelli et al., 2006) and FLUX (Cheney, 2007, 2008). A subtle issue in XQueryUpdate and XQuery! is bound to the two-phase semantics of updates that first collects updates into a *pending update list* by evaluating an expression without altering the data, and then performs all of the updates at once. An additional *snap* operator provides programmer control over when to apply pending updates. The issues of determining whether updates commute (Ghelli et al., 2008) have then been investigated. Motivated by the need of detecting data dependencies between reads and updates of XML documents, so to optimize the execution of update operations, approaches for detecting conflicts among updates (Raghavachari & Shmueli, 2006) have also been investigated.

Updates on XML schemas have received even less attention despite the great impact they may have in the database organization. Indeed, a schema change may affect the data structures developed for the organization, as well as the efficient retrieval and protection of the documents within contained. Even if commercial tools (e.g. Stylus Studio, XML Spy) have been developed for graphically designing XML Schema Definitions (XSDs), they do not support the specification of schema updates nor the semi-automatic revalidation of documents within contained. Commercial DBMSs, like Oracle 11g, Tamino, DB2 v.9, support XSDs at different levels, but the support for schema

evolution is quite limited (actually, a simplified version of adaptation is supported only by Oracle 11g through the *copyEvolve* function).

Given an XML database, updates both to documents and to the schema can lead to *evolution* or to *versioning*. With evolution the original data and schema are replaced by the updated ones upon some update primitives are applied on it. With versioning the original documents and schema are preserved and a new updated version of the document/schema created. This raises the problem of handling different versions of the same document/schema. Specifically, query execution needs to consider mappings among versions and update compositions.

Purpose of this chapter is to describe the different research proposals and the facilities of existing systems to handle XML updates at document and schema level, and their versions. Specifically, the chapter will provide a review of various language proposals for the specification of XML document updates, their different semantics and their handling of update sequences, with a focus on the XQuery Update proposal. Approaches and specific issues concerned with schema updates will then be reviewed along with document and schema versioning issues. Finally, we will discuss the current support of major DBMSs to handle updates on XML documents and schema.

LANGUAGES FOR DOCUMENT UPDATES

In this section we overview the main current proposals for XML update languages. Before getting into details of their main and distinguishing features, we first enumerate and discuss the fundamental update operations that are generally supported by every current XML update language. Then, we discuss the characteristics that every language should present and we get into details of the four most complete and interesting proposals: the W3C proposal XQuery Update Facility

Related Content

XML-Based Analysis of UML Models for Critical Systems Development
Jan Jurjensand Pasha Shabalin (2005). *Advances in UML and XML-Based Software Evolution (pp. 257-274).*
www.irma-international.org/chapter/xml-based-analysis-uml-models/4938

Support for Architectural Design and Re-Design of Embedded Systems
Alessio Bechiniand Cosimo A. Prete (2005). *Software Evolution with UML and XML (pp. 321-351).*
www.irma-international.org/chapter/support-architectural-design-design-embedded/29618

A Reuse Definition, Assessment, and Analysis Framework for UML
Donald Needham, Rodrigo Caballero, Steven Demurjian, Felix Eickhoffand Yi Zhang (2005). *Advances in UML and XML-Based Software Evolution (pp. 286-307).*
www.irma-international.org/chapter/reuse-definition-assessment-analysis-framework/4940

An Example-Based Generator of XSLT Programs
José Paulo Lealand Ricardo Queirós (2013). *Innovations in XML Applications and Metadata Management: Advancing Technologies (pp. 1-20).*
www.irma-international.org/chapter/example-based-generator-xslt-programs/73170

Augmenting UML to Support the Design and Evolution of User Interfaces
Chris Scogingsand Chris Phillips (2005). *Advances in UML and XML-Based Software Evolution (pp. 275-285).*
www.irma-international.org/chapter/augmenting-uml-support-design-evolution/4939