Concurrent Markup Hierarchies: a Computer Science Approach

Ionut E. Iacob
Department of Computer Science
University of Kentucky
Email: eiaco0@cs.uky.edu

Alex Dekhtyar
Department of Computer Science
University of Kentucky
Email: dekhtyar@cs.uky.edu

Abstract

It is known that text has not, in general, a regular structure. However, since its invention and despite to the fact that it represents hierarchical structures, XML has gained a lot of popularity among humanities researchers: XML is easy to use and it comes with a handful of free processing tools. A variety of solutions were proposed *to represent overlapping structures* in XML. More or less easy to maintain from the point of view of data management, none of these solutions provides full support for two of the most demanded processing tasks: querying and presentation (XSL-like transformation).

We propose a processing framework for complex document-centric XML which generalizes the traditional way of XML data management to support overlapping markup processing. Our framework provides support for overlapping structures representation in XML, querying, authoring, and presentation of overlapping hierarchies.

1 Introduction

The newborn TEI Overlapping Markup Special Interest Group comes to support the fact that overlapping XML structures are of great interest for text encoding community. Why is XML so popular? First at all, XML is the legitim descendant of SGML which was also popular among humanities. Then there is the fact that XML comes with a handful of processing (free) tools. This fact is clearly expressed by TEI's "Strategic Considerations in Migration of TEI Documents from SGML to XML"[1]. More specifically, DOM, SAX, XPath, and XSL and the companion software are very attractive for humanities computing. In addition, XML is flexible, intuitive, and readable: it is text, isn't it? However, there is an annoying detail about XML that does not fit into the same picture with text encodings: XML allows only properly nested markup structures. However, overlapping structures (concurrent hierarchies) often occur in applications. Czymiel points out in [1] that the proposed solutions for the overlapping markup problem fall in three categories: XML based workaround (milestones and fragmentation suggested by TEI[8]), new markup languages (LMNL[10], MECS[4], and TexMecs[5]), or content and structure separation (standoff markup, JITTs[3]). None of the solutions previously presented contains complete answers for the problems of management of concurrent XML data.

We propose a processing framework for overlapping hierarchies in XML that covers the core XML processing tasks: representation and parsing, data structure, querying, and presentation.

[1] `http://www.tei-c.org/Activities/MI/miw02.html`

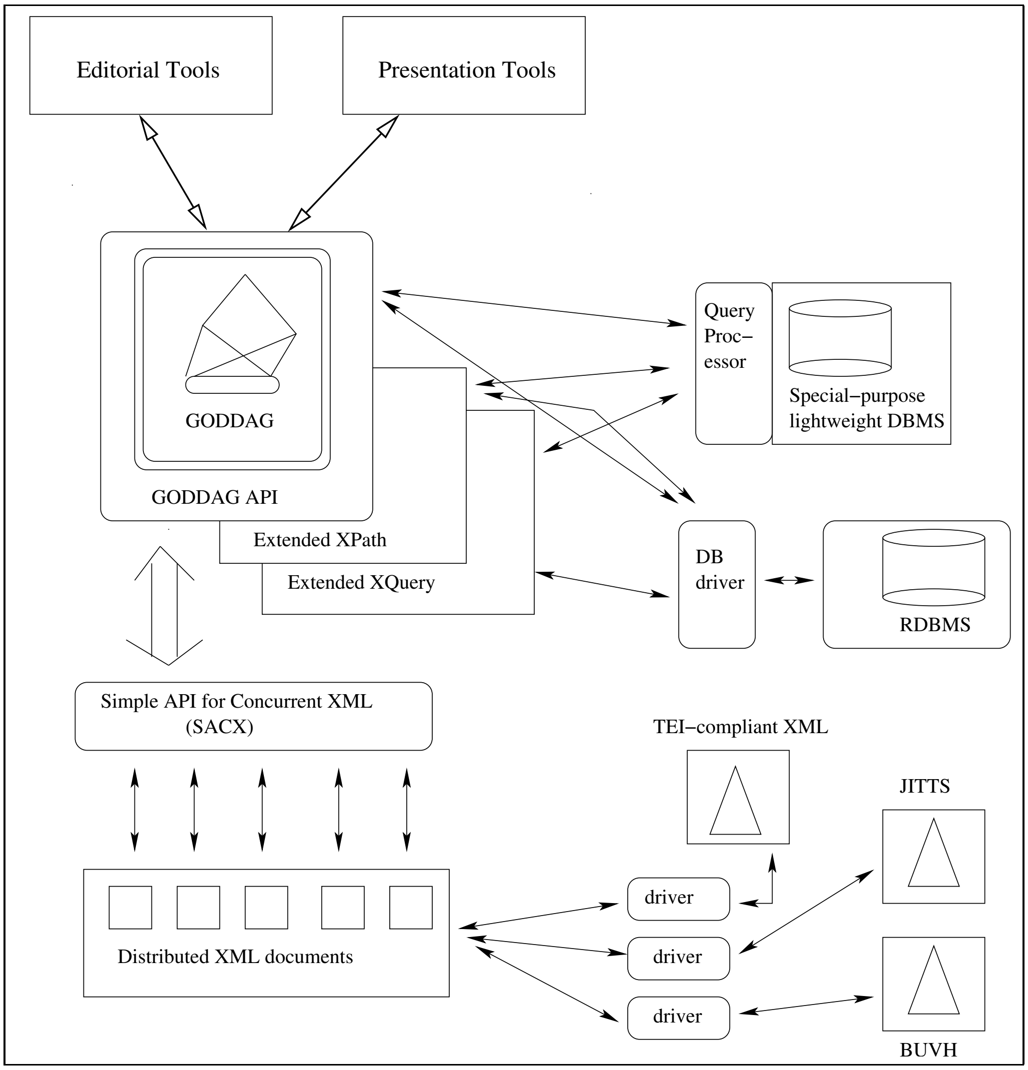

Figure 1: A framework for management of Concurrent XML Hierarchies

2 A Framework for Management of Concurrent Markup Hierarchies

The framework we propose (Figure 1) generalizes the traditional XML processing framework: parsing XML document into a DOM data structure (or, alternatively, constructing DOM from an XML database), then use the DOM API support for editing, querying, and transforming the XML document.

The core of our processing framework is the data structure for storing concurrent XML markup: the GODDAG data structure first introduced by Sperberg-McQueen and Huitfeldt in [9]. We enhanced the GODDAG with API and we have designed and implemented a parser for building a GODDAG [6] from separate XML files, one file per hierarchy (this would present the advantage of a basic concurrency control over authoring the document encodings). In general, a GODDAG data structure can be build using specialized drivers for different concurrent markup representations.

In [7] we present an extension of the XPath query language for querying concurrent markup represented as a GODDAG. As GODDAG represents a "multidimensional" genralization of DOM, our extension of XPath generalizes XPath to deal with concurrent hierarchies. In the absence of multiple hierarchies, GODDAG reduces to DOM whereas extended XPath reduces to XPath (the extended XPath semantics is given at: http://dblab.csr.uky.edu/ eiaco0/docs/expath/). With the parser and the query language we provide answers to two of the open problems in [9]. Moreover, the presentation issue (XSL) is implicitly solved as we employ patterns expressed in the extended XPath language we propose.

The XML editorial tools are based on the GODDAG API: text (PCDATA) updates, markup insertion and deletion, and searching (using the XPath extension).

For representing and storing concurrent XML markup we defined the notion of *distributed XML document* [2]: a virtual collection of XML documents, one document per hierarchy. The distributed XML document is obtained via drives from various representations: BUVH and JITTs introduced by Durusau and O'Donnell, XML documents with fragmentation and/or milestones (as in TEI).

Finally, we are currently working on implementing persistent storage support for concurrent XML hierarchies: a specialized database for storing XML with overlapping structures. Our plans include providing support for storing XML with overlapping structures in a relational database.

The framework for processing concurrent XML markup is successfully implemented in the ARCHway and Electronic Boethius projects (http://www.rch.uky.edu) at the University of Kentucky. The APIs and (part of the) software programs are available at: http://dblab.csr.uky.edu/~eiaco0/research/cmh.

References

[1] A. Czmiel. XML for Overlapping Structures (XfOS) using a non XML Data Model. In *Proc., Joint Conference of the ALLC and ACH*, 2004.

[2] A. Dekhtyar and I. E. Iacob. A Framework for Management of Concurrent XML Markup. *Data and Knowledge Engineering*, 2004. accepted.

[3] P. Durusau and M. B. O'Donnell. Concurrent Markup for XML Documents. In *Proc. XML Europe*, 2002.

[4] C. Huitfeldt. MECS - a multi-element code system, 1998.

[5] C. Huitfeldt and C. M. Sperberg-McQueen. TexMECS: An experimental markup metalanguage for complex documents. http://www.hit.uib.no/claus/mlcd/papers/texmecs.html, February 2001.

[6] I. E. Iacob, A. Dekhtyar, and K. Kaneko. Parsing Concurrent XML. In *Proceedings, 6th ACM International Workshop on Web Information and Data Management (WIDM 2004), Washington, DC.*, November 2004.

[7] I. E. Iacob, A. Dekhtyar, and W. Zhao. XPath Extension for Querying Concurrent XML Markup. Technical Report TR 394-04, University of Kentucky, Department of Computer Science, February 2004. http://www.cs.uky.edu/~dekhtyar/publications/TR394-04.ps.

[8] C. M. Sperberg-McQueen and L. Burnard(Eds.). Guidelines for Text Encoding and Interchange (P4). http://www.tei-c.org/P4X/index.html, 2001. The TEI Consortium.

[9] C. M. Sperberg-McQueen and C. Huitfeldt. GODDAG: A Data Structure for Overlapping Hierarchies, Sept. 2000. Early draft presented at the ACH-ALLC Conference in Charlottesville, June 1999.

[10] J. Tennison, G. T. Nicol, and W. Piez. Layered Markup and Annotation Language (LMNL). http://www.lmnl.org. First introduced at the Extreme Markup Languages Conference 2002, Montréal.