mining techniques beyond the Naïve Bayes model (e.g., decision trees, neural nets, support vector machines, etc.). We will also work towards the development of a system to automatically mine arbitrary bodies of critical review text such as blogs, mailing lists, and wikis. We also hope to construct content and ethnographic analyses to help answer the "why" questions that pertain to the results.

References:

Argamon, **S**., and **Levitan**, **S**. (2005). Measuring the Usefulness of Function Words for Authorship Attribution. *Proceedings of the 17th Joined International Conference of ACH/ALLC*.

Downie, **J. S**., **Unsworth**, **J**., **Yu**, **B**., **Tcheng**, **D**., **Rockwell**, **G**., and **Ramsay**, **S. J**. (2005). A Revolutionary Approach to Humanities Computing?: Tools Development and the D2K Data-Mining Framework. *Proceedings of the 17th Joined International Conference of ACH/ALLC*.

Hu, **X**., **Downie**, **J. S**., **West**, K., and **Ehmann**, **A**. (2005). Mining Music Reviews: Promising Preliminary Results. *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR)*.

Sebastiani, **F**. (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34, 1.

Stamatatos, E., Fakotakis, N., and **Kokkinakis, G.** (2000). Text Genre Detection Using Common Word Frequencies. *Proceedings of 18th International Conference on Computational Linguistics*.

Markup Languages for Complex Documents – an Interim Project Report

Claus HUITFELDT
Department of Philosophy, University of Bergen, Norway

Michael SPERBERG-MCQUEEN
World Wide Web Consortium, MIT Computer Science and Artificial Intelligence Laboratory (CSAIL)

David DUBIN
Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign

Lars G. JOHNSEN
Department of Linguistics and Comparative Literature, University of Bergen, Norway

Background

Before the advent of standards for generic markup, the lack of publicly documented and generally accepted standards made exchange and reuse of electronic documents and document processing software difficult and expensive.

SGML (Standard Generalized Markup Language) became an international standard in 1986. But it was only in 1993, with the introduction of the World Wide Web and its SGML-inspired markup language HTML (Hypertext Markup Language), that generic markup started to gain widespread acceptance in networked publishing and communication.

In 1998, the World Wide Web Consortium (W3C) released XML (Extensible Markup Language). XML is a simplified subset of SGML, aimed at retaining HTML's simplicity for managing Web documents, while exploiting more of SGML's power and flexibility. A large family of applications and related specifications has since emerged around XML. The scope of XML processing and the

DH.indb 93 6/06/06 10:55:35

complexity of its documentation now surpasses its parent.

Although proprietary formats (like PostScript, PDF, RTF etc.) are still widely in use, there has been an explosion of markup languages and applications based on XML. Today, XML is not only an essential part of the enabling technology underlying the Web, but also plays a crucial role as exchange format in databases, graphics and multimedia applications in sectors ranging from industry, over business and administration, to education and academic research.

Problems addressed

For all the developments in XML since 1998, one thing that has not changed is the understanding of XML documents as serializations of tree structures conforming to the constraints expressed in the document's DTD (Document Type Definition) or some form of schema. This seems very natural, and on our analysis the tight integration of linear form (notation), data structure and constraint language is one important key to XML's success.

Notwithstanding XML's many strengths, there are problem areas which invite further research on some of the fundamental assumptions of XML and the document models associated with it. XML strongly emphasizes and encourages a hierarchical document model, which can be validated using a context-free grammar (or other grammars that encourage a constituent structure interpretation, like context sensitive and regular grammars).

Consequently, it is a challenge to represent in XML anything that does not easily lend itself to representation by context-free or constituent structure grammars, such as overlapping or fragmented elements, and multiple co-existing complete or partial alternative structures or orderings. For the purpose of our work, we call such structures complex structures, and we call documents containing such structures complex documents.

Complex structures are ubiquitous in traditional documents — in printed as well as in manuscript sources. Common examples are associated with the physical organization of the document and the compositional structure of the text, in other words, such things as pages, columns and lines on the one hand, and chapters, sections and sentences on the other. Sentences and direct speech tend to overlap in prose, verse lines and sentences in poetry, speeches and various other phenomena in drama. Complex structures occur frequently also in databases, computer games, hypertext and computer-based literature.

In the last few years problems pertaining to complex structures have received increasing attention, resulting in proposals for

- conventions for tagging complex structures by existing notations, by extending such notations, or by designing entirely new notations;
- alternative data structures;
- explications of the semantic relationships cued by markup in a form that is more easily machine-processable.

The MLCD (Markup Languages for Complex Documents) project aims to integrate such alternative approaches by developing both an alternative notation, a data structure and a constraint language which as far as possible is compatible with and retains the strengths of XML-based markup, yet solves the problems with representation and processing of complex structures.

MLCD started in 2001 and is expected to complete its work in 2007. The project is a collaboration between a group of researchers based at several different institutions. The remainder of this paper presents an interim report from the project

Data Structure

One of the early achievements of MLCD was the specification of the GODDAG (Generalized Ordered-Descendant Directed Acyclic Graph) structure. It was originally based on the realization that overlap (which was the first kind of complex structure we considered) can be represented simply as multiple parentage.

A GODDAG is a directed acyclic graph in which each node is either a leaf node, labeled with a character string, or a nonterminal node, labeled with a generic identifier. Directed arcs connect nonterminal nodes with each other and with leaf nodes. No node dominates another node both directly and indirectly, but any node may be dominated by any number of other nodes.

We distinguish a restricted and a generalized form

DH.indb 94 6/06/06 10:55:35

of GODDAG. Conventional XML trees satisfy the requirements of generalized as well as restricted GODDAGs. In addition, restricted GODDAGs lend themselves to representation of documents with concurrent hierarchies or arbitrarily overlapping elements, whereas generalized GODDAGs also allow for a convenient representation of documents with multiple roots, with alternate orderings, and discontinuous or fragmented elements.

The similarities between trees and GODDAGs allow similar methods of interpreting the meaning of markup: properties can be inherited from a parent, overridden by a descendant, and so on. There is some chance for conflict and confusion, since with multiple parents, it is possible that different parents have different and incompatible properties.

Recent work has revealed a weakness in the current specification of GODDAG, which leads to problems with the representation of discontinuous elements. In the full version of this paper we will present the results of our work towards a solution to these problems.

Notation

It is always possible to construct GODDAGs from XML documents. In the general case, they will be trees, which are subsets of GODDAGs. It is also possible to construct GODDAGs from the various mechanisms customarily used in order to represent complex structures in XML. However, these mechanisms depend on application-specific processing and vocabularies, and tend to be cumbersome.

Thus, one may either try to establish standards for the representation of complex structures in XML, or provide an alternative notation which lends itself to a more straightforward representation of complex structures. We believe that these options are complimentary, and that both should be pursued.

Thus, we have defined an alternative notation to XML, TexMECS. The basic principles of its design are:

- For documents that exhibit a straightforward hierarchical structure, TexMECS is isomorphic to XML.
- Every TexMECS document is translatable into a GODDAG structure without application-specific processing.
- Every GODDAG structure is representable as a TexMECS document.

A particular advantage of TexMECS is a simple and straightforward notation for what we have called complex structures.

We also plan to design algorithms for translating widely recognized XML conventions for representation of complex structures into GODDAGs, and vice versa. In the full version of this paper we will report on our latest work in this area.

Constraint Language

One of the most important remaining tasks for the MLCD project is the identification of a constraint mechanism which relates to GODDAGs as naturally as constituent structure grammars relate to trees, which constitute a subset of GODDAGs. Constraint languages for XML documents exist in the form of XML DTDs, XML Schema, Relax NG and others. These methods invariably define context-free grammars allowing the representation of XML documents in the form of parse trees. However, since GODDAG structures are directed acyclic graphs more general than trees, they cannot easily be identified with parse trees based on context-free grammars.

Several possible ways forward exist and remain to be explored. MLCD has decided to focus on two approaches, one grammar-based and one predicate-based.

The grammar-based approach starts from the observation that GODDAGs can be projected into sets of tangled trees. One way to achieve at least partial validation of complex documents, therefore, is to write grammars for each such tree and validate each projection against the appropriate grammar. Each such grammar will treat some start- and end-tags in the usual way as brackets surrounding structural units, but treat other start- and end-tags as if they were empty elements. This allows some measure of control over the interaction and overlapping of specific elements in different grammars; whether it provides enough control remains to be explored.

Another approach to validation is to abandon the notion of document grammars, and regard validation simply as the establishment of some set of useful invariants. A schema then takes the form of a set of predicates; the document is valid if and only if all of the required predicates are true in that document. In the XML context,

DH.indb 95 6/06/06 10:55:35

this approach is represented by Schematron. It is clear that it can also be applied to documents with complex structures, if the language used to formulate the required predicates is extended appropriately.

In the full version of this paper we will report on our attempts to pursue each of these two approaches.

References

Barnard, D.; Burnard, L.; Gaspart, J.; Price, L.; Sperberg-McQueen, C.M. and Varile, G.B. (1995) Hierarchical encoding of text: Technical problems and SGML solutions. *Computers and the Humanities*, 29, 1995.

Barnard, D.; Hayter, R.; Karababa, M; Logan, G. and McFadden, J. (1988) SGML-based markup for literary texts: Two problems and some solutions. *Computers and the Humanities*, 22, 1988.

Dekhtyar, A. and Iacob, I.E. (2005) A framework for management of concurrent XML markup. *Data and Knowledge Engineering*, 52(2):185–215, 2005.

DeRose, S. (2004) Markup overlap: A review and a horse. In *Extreme Markup Languages 2004*, Montreal, 2004. IDEAlliance. Available on the Web at <http://www.mulberrytech.com/Extreme /Proceedings/html/2004/DeRose01 /EML2004DeRose01.html>.

Dubin, D., Sperberg-McQueen, C. M., Renear, A., and Huitfeldt, C. (2003) "A logic programming environment for document semantics and inference", *Literary and Linguistic Computing*, 18.2 (2003): 225–233 (a corrected version of an article that appeared in 18:1 pp. 39-47).

Durusau, P. and O'Donnell, M.B. (2004) *Tabling* the overlap discussion. In *Extreme Markup Languages 2004*, Montreal, 2004. IDEAlliance. Available on the Web at <http://www.mulberrytech.com/Extreme /Proceedings/html/2004/Durusau01 / EML2004Durusau01.html>

Hilbert, M.; Schonefeld, O, and Witt, A. (2005) Making CONCUR work. In *Extreme Markup Languages 2005*, Montreal, 2005. IDEAlliance. Available on the Web at <http://www.mulberrytech.com/Extreme /Proceedings/html/2005/Witt01 /EML2005Witt01.xml>.

Huitfeldt, C. (1995) Multi-dimensional texts in a one-dimensional medium. *Computers and the Humanities*, 28:235–241, 1995.

Huitfeldt, C. (1999) *MECS—A multi-element code system*. Working papers of the Wittgenstein Archives at the University of Bergen. Wittgenstein Archives at the University of Bergen, Bergen, 1999.

Huitfeldt, C. and Sperberg-McQueen, C.M. (2001) Texmecs: An experimental markup meta-language for complex documents. Available on the Web at <http://helmer.aksis.uib.no/claus/mlcd/papers/texmecs.html>, 2001.

Huitfeldt, C. (2003) "Scholarly Text Processing and Future Markup Systems" in Georg Braungart, Karl Eibl, Fotis Jannidis (eds.): *Jahrbuch für Computerphilologie 5 (2003)*, Paderborn: mentis Verlag 2003, pp. 217-233. Available on the Web at <http://computerphilologie.uni-muenchen.de/jg03/huitfeldt.html>

Jaakkola, J. and Kilpeläinen, P. (1998) *Sgrep home page*, 1998. Available on the Web at <http://www.cs.helsinki.fi/u/jjaakkol/sgrep.html>.

Jagadish, H. V.; Laks V. S.; , Scannapieco, M.; Srivastava, D. and Wiwatwattana, N. (2004) Colorful XML: One hierarchy isn't enough. In *Proceedings of the 2004 ACM SIGMOD International conference on management of data, Paris*, New York, 2004. Association for Computing Machinery Special Interest Group on Management of Data, ACM Press

Nicol, G. (2002) Core range algebra: Toward a formal theory of markup. In *Extreme Markup Languages 2002*, Montreal, 2002. IDEAlliance. Available on the Web at <http://www.mulberrytech.com/Extreme / Proceedings/html/2002/Nicol01 /EML2002Nicol01.html>.

Piez, W. (2004) Half-steps toward LMNL. In Proceedings of Extreme Markup Languages 2004, Montreal, Quebec, August 2004.

Renear, A.; Mylonas, E. and Durand, D. (1993) Refining our notion of what text really is: The problem

DH.indb 96 6/06/06 10:55:35

of overlapping hierarchies. In N. Ide and S. Hockey, editors, *Research in Humanities Computing*, Oxford, 1993. Oxford University Press. Available on the Web at: <http://www.stg.brown.edu /resources/ stg/monographs/ohco.html>.

Sasaki, F. (2004) Secondary information structuring: A methodology for the vertical interrelation of information resources. In *Extreme Markup Languages 2004*, Montreal, 2004. IDEAlliance. Available on the Web at <http://www.mulberrytech. com/Extreme /Proceedings/html/2004/Sasaki01/ EML2004Sasaki01.html>.

Sperberg-McQueen, C.M., and Burnard, L. (eds.) (2001) Guidelines for Electronic Text Encoding and Interchange. Chicago and Oxford: TEI P4, 2001.

Sperberg-McQueen, C. M., Huitfeldt, C. and Renear, A. (2000) "Meaning and interpretation of markup". *Markup Languages: Theory and Practice 2*, 3 (2000), 215–234.

Sperberg-McQueen, C. M. and Huitfeldt, C. (1999) Concurrent document hierarchies in MECS and SGML. *Literary & Linguistic Computing*, 14(1): 29–42, 1999. Available on the Web at <http://www. w3.org/People/cmsmcq/2000/poddp2000.html>.

Sperberg-McQueen, C. M. and Huitfeldt, C. (2000) Goddag: A data structure for overlapping hierarchies. In P. King and E. Munson, editors, *DDEP-PODDP 2000*, number 2023 in Lecture Notes in Computer Science, pages 139–160, Berlin, 2004. Springer. Available on the Web at <http://www.w3.org/People /cmsmcq/2000/poddp2000.html>.

Tennison, J. and Piez, W. (2002) Lmnl syntax. Available on the Web at <http://www.lmnl.net/prose /syntax/ index.html>, 2002.

Witt, A. (2004) Multiple hierarchies: new aspects of an old solution. In *Extreme Markup Languages 2004*, Montreal, 2004. IDEAlliance. Available on the Web at <http://www.mulberrytech.com/Extreme /Proceedings/html/2004/Witt01 /EML2004Witt01. html>.

Semantic Timeline Tools for History and Criticism

Matt JENSEN
NewsBlip

New concepts in the visualization of time-based events are introduced and applied to the fields of historiography and criticism. These techniques (perpendicular timelines, dynamic confidence links, and time-slice relationship diagrams) extend the semantic power of timelines so that they can show the development of complex concepts and interpretations of underlying events. An interactive software tool called "TimeVis" illustrates these techniques with both 2D and 3D views.

History is a referent discipline. Later events build on earlier events, though in unpredictable and complicated ways. Historiography and literary criticism are the histories of accumulated comments on a subject. The underlying history and literature (the "base events") occur in one era, and commentary and subsequent events ("secondary events") are added later. However, commentary is not made in the same order as the base events; scholars might spend decades analyzing a writer's later works, and subsequently change emphasis to her earlier works.

Visualizing such referent-based relationships through time is very difficult with a single, conventional timeline. The concept of stacked timelines of different eras [Jen03] was introduced to align commentary and consequent events with their referents (Figure 1). This is useful when secondary events are evenly distributed, but less useful when they are concentrated on subsets of the base events. Crossing lines are difficult to interpret, and important early events can end up leading to a forest of arrows. What is more, the x-axes of the two timelines have no relation to each other. This lack of relation is in fact the cause of the criss-crossing lines.

This paper describes three new timeline techniques that can be applied to the study of history, criticism, and other fields with a temporal or referent component. Each technique serves a different research need.

DH.indb 97 6/06/06 10:55:36