What do those weird XML types want, anyway?

Steven J. DeRose

Brown University and Inso Corporation

Providence, RI

USA

Steven_DeRose@Brown.edu

XML tries to bring to natural language documents ("texts" for short), some of what databases have had for decades: explicit structure whose properties can be known; independence of data and structure from reporting (which we foreigners call "formatting"); various kinds of isolation; and so on. But the data buried in those XML elements is weird stuff: deep hierarchies, arbitrary and unpredictable orderings and repetitions, a painful number of atomic types, and enough aggregates to make one's head hurt.

Among the larger problems for dealing with such data is that it has an infuriating combination of the properties of "structured" database data and "unstructured" natural language. I find these labels a bit misleading, so let's cash them out in a little more detail. Databases pin down data types, field sizes and the like, but "meaning" is often conveyed merely by the mnemonics of field names. Both "isbn" and "phone" are likely of the same data type, so have the same structure at one level; but of course distinguishing them makes a difference to querying. Object-oriented databases build more "meaning" in, partly via methods, but this "operational" meaning does not always map cleanly to the kinds of operations humans wish to perform.

In contrast, marked-up text has weak structure in the sense of data typing, but much more indication of the "structure" of interest to users exploring a huge information space. It seems to the document world that "phone-number-ness" has as much claim to being structural information, as does "10-digit numeric field". Texts also include highly structured information such as bibliographies (as well as long undifferentiated prose sections). So we quickly hit a terminological disconnect: what's structure in one world, is un-structure in the other; hence the compromise term "semi-structured," which probably satisfies no one completely.

I'll mention only in passing, various other typological differences in "structure". The order of objects is a fundamental part of the information in documents, though not in the most common database algebras: re-ordering the paragraphs of Hamlet fundamentally changes the structure present, in a way that re-ordering the fields of a relation simply does not. Likewise, documents abound with recursive partitions, or aggregates: any character in Hamlet's soliloquy is just as much a part of Scene 1, of Act 3, etc.

As a practical example of the messy phenomena of text, when I was first preparing for this talk I thought it would make sense to read some of the prior proceedings. After a *lot* of creative searching at Amazon and at LC, I found 18 volumes (including near-duplicates for 1990 and 1994, so really 16). Thus a recall of 64% (my IR friends would be upset if I didn't get "recall" or "precision" in here somewhere):

Very large data bases: proceedings / International Conference on Very Large Data Bases. 1977: Data base; v. 9, no. 2; 1977: SIGMOD record ; v. 9, no. 4.

Proceedings of the 25th VLDB Conference, Edinburgh, Scotland, 1999.

Notes: Title from cover. Vols. for 1983 and 1985 have solely the name of the conference as the title. Subtitle varies.

Proceedings of the ... International Conference on Very Large Data Bases.

Systems for large data bases: proceedings of the 2nd International Conference on Very Large Date [sic] Bases.

Very Large Data Bases: Proceedings International Conference on Very Large Data Bases / Published 1981.

Very Large Data Bases: 8th Intl Conference on Very Large Data Bases Mexico City, Mexico / Published 1982.

Proceedings VLDB 83 / Published 1983.

Very Large Data Basis Conference Proceedings: Singapore 84 (VLDB-84) Paperback / Published 1984.

Very Large Data Bases: Proceedings, 11th International Conference on Very Large Data Bases / Published 1985.

Very Large Data Bases: Proceedings, 12th International Conference on Very Large Data Bases / Published 1986.

Proceedings of the Thirteenth International Conference on Very Large Data Bases, Brighton, England, 1987 Peter M. Stocker, William Kent (Editor) / Published 1987.

Proceedings of the Fourteenth International Conference on Very Large Data Bases François Bancilhon, David J. DeWitt (Editor) / Published 1988.

Proceedings VLDB 89 International Conference on Very Large Data Bases / Published 1989.

Very Large Data Bases: 16th International Conference on Very Large Data Bases / Proceedings: August 13-16, 1990, Brisbane, Australia Dennis McLeod, *et al*. / Published 1990.

Very Large Data Base Conference Proceedings 1991 (#Vl91) / Published 1990 Proceedings of the Seventeenth International Conference on Very Large Data Bases: September 3-6, 1991: Barcelona (Catalonia, Spain) Guy M. Lohman, *et al.* / Published 1992.

Very Large Data Bases, '92: Proceedings of the 18th International Conference on Very Large Data Bases, August 23-27, 1992 Vancouver, Canada Li-Yan Yuan (Editor) / Published 1992.

Proceedings 19th International Conference on Very Large Data Bases / Published 1994.

Proceedings 19th International Conference on Very Large Data Bases : August 24th-27th 1993, Dublin, Ireland Rakesh Agrawal, *et al*. / Published 1994.

Proceedings of the 20th International Conference on Very Large Data Bases: 20th VLDB Conference September 12-15, 1994 Santiago-Chile (#Vl94) Jorge Bocca / Published 1994.

Proceedings of the International Conferences on Very Large Databases Held in Zurich, Switzerland: VLDB-95 / Published 1998.

Proceedings of the International Conferences on Very Large Databases Held in Bombay, India / Published 1996.

Proceedings of the Twenty-Fourth International Conference on Very Large Databases: New York, NY, USA 24-27 August 1998 (24th Conf) / Published 1998.

These are all obvious to us as humans (1983 is my favourite), but the variety of detail is astonishing (the more so because it is not unusual). Many of the nastiest problems of retrieval in large, but especially heterogeneous, text bases are hinted at here. Morphology ("database" vs. "databases" vs. "data base" vs. "data bases"). Alternate descriptions ("11th" vs. "1985" vs. "85"); different representations of the same data type ("24" vs. "24th" vs. "twenty-fourth"). Structural issues (dates within title vs. in publication date; different dates in both); missing or incomplete data (editors, authors, "*et al.*", locations); and much more.

A system smart enough to do this retrieval right in a single attempt, and to understand the internal structure of this data (that we humans perceive so readily), would go far towards meeting the retrieval needs of text base users and scholars. Oh, it should also catch the years I never was able to locate.

So far, this should be familiar turf. But note that text aggregates have another very annoying property: the data they serve to partition and label must also be treated as a contiguous whole for some purposes. "The text" spans partition (or "element", as we say in XML-land) boundaries, almost but not quite arbitrarily. Speech boundaries in a play usually imply larger discourse boundaries, but there are plenty of cases where one speaker picks up another's sentence -- and these phenomena are typically important. Even at the lowest levels, the boundaries are never sure. Many of the most important, most studied (for our purposes, most queried) texts have come to us in unsure or variant forms. Faithful representations note structure even within words: <sic corr="affect">effect</sic> is an obvious case, and one just as well marked up <sic corr="a">e</sic>ffect. So far, search and retrieval algorithms available to the text-computing user do not deal well with aggregates.

Even determining "the text" to index or query on is hard. A naive understanding of mark up says that what's between the pointy-brackets is meta data, and what's not, is content. But then how should queries involving "affect" and "effect" treat examples such as those just shown? Worse, how should queries operate around textual discontinuities such as footnotes? Should not the words adjacent to a footnote on each side be a phrase? Hypertext links add to the complexity because they can express content sequence and hierarchy, cross-reference, and many other relationships, but in pre-XLink Web technology we cannot say anything definitive *about* what they are saying.

I'll digress to suggest that some of these problems seem a consequence of a wholly inadequate model of text that has become entrenched in the word-processor world and influenced much thinking since. Word-processors typically view documents not as ordered hierarchies, but as lists of paragraphs (at best, some paragraphs may be styled with mnemonics such as "H1"). Aggregates for the most part do not exist: chapters, sections, even lists. This directly leads to bizarre behavior like the list-numbering anomalies of popular systems. Even a slight concession to structure would solve myriad problems. HTML slightly addresses this weakness, as well as various mechanical pains such as the need to parse binary hash rather than just characters. XML promises much more, as more meaningful tag-sets become standard in various genres and domains and as authors and software begin to make better use of them. C's oversimplification that a string is merely an array also didn't help (and seems to have led to more system vulnerabilities than just about anything else).

To resume, a further problem documents bring to the fore is polysemy of many kinds. At first it seems that at the bottom we hit something more tractable then multi-level aggregates: characters. Yet even these are not so simple: "12" and "B" and "0x'0b" and "twelve" and "dose" are all numbers, and people want them kept just that way, yet to compare equal...sometimes. "the ides of March" is a date (and a discontiguous one at that).

At the same level are problems of what logicians call "definite description", where multiple descriptors refer to the same object -- sometimes. Pronouns have of course received some treatment, but time-variant descriptions seem little addressed. Saying "I want to meet the Mayor" may be clear today, but next year when there is a new mayor it becomes ambiguous. Dealing with such "de dicto, de re" ambiguities in texts deserves a few IR and/or database dissertations.

A few levels up, we hit plays on language that may often be critical to understanding the text. A wonderful little book called "Oddities and Curiosities of Words and Literature" (C. C. Bombaugh, out of print but fairly easy to find used), gives many examples where text plays "structure" against "content", such as a 2-column letter of recommendation that reads entirely differently down the columns versus across. A more familiar case is "She went to Essex; she had always liked Essex", where the very ambiguity of "Essex" as place or person, is critical to understanding the text.

And finally, most pernicious of all is that even the best-intentioned, most thoroughly edited and analyzed texts, can only express some of the desired structures. This is of course no reason not to use the structure that *is* there; I am astonished by the number of recent papers where the system actively discards structural information that is there to start with, and then boasts of brilliant AI or heuristics to re-generate some portion of it. The reason is obvious: not all texts have any useful structural information available, and one wants to be inclusive. Yet discarding it when you have it, seems to me as absurd as building a database that cannot use field or object names,, but responds to queries by doing its best to guess which fields are phone numbers, first or last names, etc, on the ground that not all data is broken down just so.

Such phenomena are hard to manage; yet benefits of the traditional database strengths are desperately needed, for all the familiar reasons. Analysts commonly state that 90% of corporate data is in documents, not databases; But however much there may be, it is nowhere near so accessible or manageable.

How can we access this data in more useful ways, more akin to what we (after decades of hard work) can take for granted with databases? Too much of that data languishes in GIF files, bizarre formats, or, not much better, "plain text" where you can't tell the title from the colophon. What kinds of queries apply to XML data structures, and what new opportunities do they present? What can we expect from data a few years from now, and what can we hope to do with it once we have it?

In XML and SGML history the literary scholars discover and solve problems an average of 3 years before industrial users, so they give us a glimpse into the future. Structuring texts can make a difference toward making this truly enormous database called the Web, or ideally called human literature, all it might become.

XML is only step one (or perhaps step 3, following SGML and HTML): it gets rid of the most mechanical, mundane level of parsing and character set incompatibilities. But this does not solve the real problems: it merely clears away the ground cover that hides them. When we couldn't even read each other's files due to proprietary binary word-processor formats, it was hard to notice that even if we could, the documents didn't contain the information most useful to use for any task *other* than formatting. Now we are moving past that first hurdle, and the issues of schemas, semi-structured and semi-ambiguous data, hyperlinking, and retrieval in the face of all these, can come to the fore.

Web-crawlers are obviously not going to cut it, even if enhanced with the best of the capabilities I hope for. I used to say that crawlers were typically 6 months behind (audiences were aghast). I have new news: They no longer even try to keep up. I was wiring my house for Ethernet, and wanted some basic information -- I couldn't find it on the Web. So I got a tutorial from my faithful sysadmin, and then wrote it up and put it on my Web page. Several months later I thought to try searching for it: no luck. So I manually submitted the URL: a few days later a stream of e-mail responses to the page began poring in. Asking around, I discovered that only about 37% of the Web is indexed by even the largest crawlers.

I think the way forward with such data is to integrate tightly the quite different powers of structural algebras and natural language statistics; of "markup" and "content"; of links and hierarchies; in short, of language and data. Yet, although I think structured documents (what many called "semi-structured") is where it's at, and where the power for future retrieval lies, this very large database we called the Web also poses a painfully mundane problem that needs entirely different solutions; but that is another talk.

Bibliography

Abiteboul, Serge et al. 1997. "Querying Documents in Object Databases." In *International Journal on Digital Libraries* 1(1): 5–19.

Agosti, Maristelle and Alan Smeaton. 1996. *Information Retrieval and Hypertext*. Boston: Kluwer Academic Publishers. ISBN 0-7923-9710-X.

André, Jacques, Richard Furuta, and Vincent Quint (eds). 1989. *Structured Documents*. Cambridge: Cambridge University Press. ISBN 0-521-36554-6.

Bishop, Ann Peterson. 1997. "Digital Libraries and the Disaggregation of Knowledge: An Investigation of the Use of Journal Article Components by Researchers." National Synchronization Meeting. Digital Library Initiative, Pittsburgh PA, June 5.

Coombs, James H., Allen H. Renear, and Steven J. DeRose. 1987. "Markup Systems and the Future of Scholarly Text Processing." *Communications of the Association for Computing Machinery* 30 (11): 933-947.

Gibson, David, Jon Kleinberg, and Prabhakar Raghavan. 1998. "Inferring Web Communities from Link Topology." In *Proceedings of Hypertext '98,* Pittsburgh, PA. Association for Computing Machinery Press.

Hall, Wendy, Hugh Davis, and Gerard Hutchings. 1996. *Rethinking Hypermedia: The Microcosm Approach.* Boston: Kluwer Academic Publishers. ISBN 0-7923-9679-0.

Hitchcock, S. et al. 1997. "Citation Linking: Improving Access to Online Journals." In *Proceedings of ACM Digital Libraries '97.* New York: The Association for Computing Machinery.

Myaeng, Sung Hyon, Dong-Hyun Jang, Mun-Seok Kim, and Zong-Cheol Zhoo. 1998. "A Flexible Model for Retrieval of SGML Documents." Pp. 138-145 in SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. W. Bruce Croft et al. (eds). Melbourne Australia, August 24-28. NY: ACM Press.

Simons, Gary F. 1997. "Using Architectural Forms to Map SGML Data Into an Object-Oriented Database. In Proceedings of *SGML/XML '97.* Washington, D.C., December 7–12: 449–460. Sponsored by the Graphic Communications Association (GCA) and Co-sponsored by SGML Open.

Subramanian, Bharathi, Theodore W. Leung, Scott L. Vandenberg, and Stanley B. Zdonik. 1995. "The AQUA Approach to Querying Lists and Trees in Object-Oriented Databases." Presented at the International Conference on Data Engineering, Taipei, Taiwan. Available from the authors.

Tajima, Keishi, Yoshiaki Mizuuchi, Masatsugu Kitagawa, and Katsumi Tanaka. 1998. "Cut as a Querying Unit for WWW, Netnews, and E-mail." In *Proceedings of Hypertext 98*. Pittsburgh: June 20–24: 217–224. New York: ACM Press.

Trigg, Randall H. "Guided Tours and Tabletops: Tools for Communicating in a Hypertext Environment." In ACM Transactions on Office Information Systems, 6.4 (October 1988): 398-414.

Zamir, Oren and Oren Etzioni. 1998. "Web Document Clustering: A feasibility demonstration." In SIGIR '98.