Active Behaviors within XML Document Management

Angela Bonifati
Politecnico di Milano, Dipartimento di Elettronica e Informazione - Email: bonifati@elet.polimi.it

Extended Abstract

XML is rapidly emerging as the most widely adopted technology for information representation and exchange on the WWW [27]. Novel languages [2, 17, 9, 24, 28] for extracting and restructuring the XML content have been proposed, some in the tradition of database query languages, others more closely inspired by XML [21]. The discussions on standardization within the World Wide Web Consortium proceed very fast and a standard query language will be soon chosen and made available [1]. Sophisticated query engines, that allow user to deal with semi-structured data, will be crucial to exploiting the full power of XML. On the other side, traditional query engines are being used in practice for processing XML documents conforming to DTDs: translators of XML queries to SQL queries over tables are likely to be effective in most cases. Under this scenario, it looks clear that existing database technology, involving structured data, will both influence and be influenced by the XML Web revolution.

Our focus here is to examine how the mature technology from active databases can be mirrored in an XML enviroment. The work is ongoing, so some results have already been achieved, but others have only been foreshadowed.

Active rules are specifications of stereotypical reactions automatically performed by a DBMS in response to the detection of particular DB-related events. A lot of attention has been devoted from research community to active databases so far [13, 15, 16, 18, 26]. A wide range of potential applications of the active rule concept have been identified; they can be distinguished in two major classes: external applications and internal applications. External applications concern reactions to external real-world stimuli that have some effects on the enviroment of the database; among them, we envision: alerters, raising up in some "critical" situations; monitors, such as modifications of the database that are meaningful from outside the database; and constraint enforcement, such as e.g. automated filling up of some missing records. Internal applications typically address the realization of database functionalities, like materialized view maintenance, integrity checking, access control, log maintenance, recovery, or view update implementation.

We believe that active database rules represent a consolidated technology, and there are many advantages to study how it relates to XML data. The problem is two-fold: on one side, investigation of novel applications of triggers to XML document management and their implementation within a full-fledged document manager, on the other side integration and extension of existing active database technology to yield XML-coded information.

Active Capabilities upon XML Document Management Systems. As soon as XML will become more used, direct manipulation of XML documents (by users or applications) will become more common. In this environment, it is important to develop active document management systems, by adding reactive capabilities to XML repositories. Such systems have important potential applications and constitute a natural framework for the integration of services, which are currently offered by separate mechanisms. Within XML document systems, external rules are monitors of accesses and changes to documents (implementing the so called "push technology"), then constraint enforcement corresponds to check the validity of documents (super-imposed by DTDs and XML schemas) and\or to add to them the desiderable constraints, not supported by their schema definition language; internal rules are e.g. those for maintaining document consistency (including refreshing copies and views of documents), for checking and controlling document's quality and versioning and for abstracting, classifying, and archiving documents.

Comparative analysis of five XML query languages. This doctoral thesis starts investigating the semantics of five, representative XML query languages (Lorel [2], XML-QL [17], XML-GL [9], XQL [24], XSL [28]), highlighting their common features and their most incisive differences [7]. From this analysis, it results that query languages for XML constitute a language hierarchy similar to the one existing for relational and object-relational databases.

Extension of XML-GL to support XML Active Rules. The definition of active ECA rules for document management requires the design of several new concepts, such as an event and update model for XML documents, and their integration with XML query languages. In the thesis, we choose XML-GL [9], a graphical query language XML-GL and extend it with an event and update model [8]. An example of XML-GL active rule is depicted in Figure 1: it reacts to inserting of `<partner>` elements of proposals and propagates the insertions to the abstracts corresponding to those proposals.

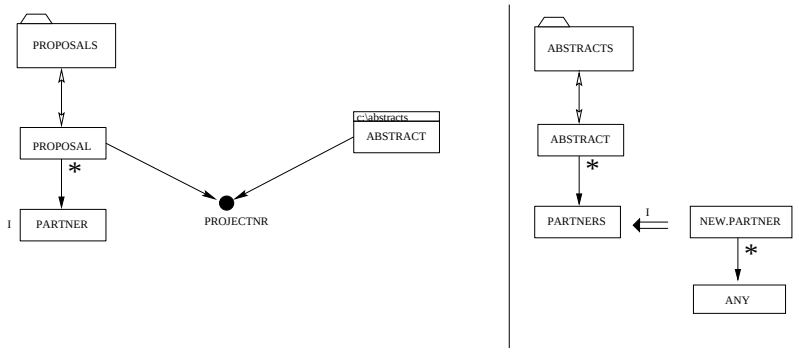

Figure 1: Example of Active XML-GL Rule

Our approach is independent from the particular adopted XML query language and will be easily extended to the standard, as quickly as it will be provided. In particular, we show how conditions and actions of Active XML-GL can be translated into Lorel [19], that results from the previous analysis the most expressive XML query language (translation for the rule of the example is shown in Figure 2).

```
EVENT:            insert (proposal.#.partner PP)
CONDITION-ACTION: update A.#.partners.partner += PP
                  from   abstract A,  proposal P
                  where  P.projectNr = A.projectNr
                  and    P.#.partner = PP
```

Figure 2: Example of Translation of an Active XML-GL Rule

We use edit scripts for representing the difference among different states of the same document, which can be arbitrarily produced by an editing session or by integrating the document editor with electronic mail. Detecting changes on a document is a general problem, considered in [14, 12, 11]. We assume that the problem can be solved by executing an *XML-diff* algorithm, which produces an optional (i.e., undefined on certain nodes) one-to-one identity relationship between nodes of the old and new version of the document, so that any two related nodes must be considered as two versions of the same node. Event detection is performed by simple look-up programs on the edit-script: these programs receive the event being monitored by a given rule and perform a query on a portion of the edit-script falling under the *scope* of the rule.

We define two alternative semantics for active rules (set-oriented and instance-oriented) in Active XML-GL, and specify the steps of *rule activation, triggering, consideration and execution.*

Properties of XML Active Rules. We show several examples of rules and discuss the execution of a set of rules, possibly cascading or in conflict, analysing their behavior with respect to the properties of termination, confluence, and observable determinism [3, 4, 5, 6, 20, 25]. These properties are defined regardless of the notion of edit-script, but the edit script considered by a given rule engine and in a given rule execution is one of the many equivalent edit scripts that can be produced by an XML-diff algorithm, which operates with a given optimization strategy for choosing among equivalent edit scripts. Therefore, we propose the important property of *edit-script independence* and exploit its application to various types of data, in particular to semi-structured data.

Informally, edit-script independence requires that the rules in a rule set monitor all the possible activities that may describe a document transformation in an equivalent way, and then that the rules

collectively behave in the same way once anyone of them is activated. The net effect semantics for the evaluation of side effects of update operations is very helpful for guaranteeing edit-script independence, because it combines the effect of multiple operations over the same portions of XML documents observed at each execution of a rule.

It is important to note that edit-script independence is orthogonal to the cited classical properties and holds for relational data and for (identity-based) and (value-based) objects. Under certain assumptions of documents validity and edit-script validity, we show and demonstrate in which cases edit-script independence is assured.

Future research directions. This thesis is under development and many other extensions and contributions need to be incorporated [1].

A prototype of Active XML-GL is being designed: it is based on graph grammars for capturing rules and then on the mapping on Lorel queries. Furthermore, we think to use underlying relational database active technology in the implementation of Active rules: Oracle 8i [23] support's for querying XML documents uses a relational engine and an overflow text file, so a mapping from active ECA rules to DB triggers and triggering adjuncts on the overflow text file has to be defined and exploited.

Many other issues remain to be covered and are the target of our current study: among them, the Active Document Base we are developing is light-weighted, in that it does not support all the capabilities of a fully-functional DBMS: a coupling with a transactional system and an embedding of security capabilities has to be added; then, an efficient indexing, a low-cost storage of XML data, a cost-based query optimizer, a multi-user support, logging and recovery have to be incorporated.

Finally, a further research direction has been only preliminarily sketched, and it seems very promising: it aims to express traditional DB triggers in ad-hoc XML needed information. This has a great impact on interoperability among databases and document bases and we believe that it will require a lot of study, before it will become fully understood. Sophisticated applications may be solicitated in order to make legacy databases more and more ubiquitously "Web-interfaced" and "XML-sensitive".

References

[1] W3C XML Activity, http://www.w3.org/XML /Activity.html.

[2] S. Abiteboul, D. Quass, J. McHugh, J. Widom and J. Wiener. The Lorel Query Language for Semistructured Data. In *International Journal on Digital Libraries*, 1(1):66-88, April 1997.

[3] A. Aiken, J. Widom and J. M. Hellerstein. Static Analysis Techniques for Predicting the Behavior of Active Database Rules. In *ACM Transactions on Database Systems*, 20(1):3-41, March 1995.

[4] E. Baralis, S. Ceri and S. Paraboschi. Compile-Time and Runtime Analysis of Active Behaviors. In *TKDE*, 10(3):353-370, 1993.

[5] E. Baralis, S. Ceri and S. Paraboschi. Modularization Techniques for Active Rules Design. In *ACM TODS*, 21(1):1-29, 1996.

[6] E. Baralis and J. Widom. An Algebraic Approach to Rule Analysis in Expert Database Systems. In *Proc. of the 20th VLDB*, pages 606-617, Santiago, Chile, September 1994.

[7] A. Bonifati and S. Ceri. Comparative Analysis of Five XML Query Languages. To appear on *ACM Sigmod Record*, March 2000.

[8] A. Bonifati, S. Ceri and S. Paraboschi. Active Management of XML Documents. Submitted to *2000 ACM SIGMOD Intl. Conference on Management of Data*, Dallas, Texas, May 14-19, 2000.

[9] S. Ceri, S. Comai, E. Damiani, P. Fraternali, S. Paraboschi and L. Tanca. XML-GL: A Graphical Language for Querying and Restructuring WWW Data. In *8th Intl. World Wide Web Conference*, WWW8, Toronto, Canada, May 1999, www8.org/fullpaper.html.

[10] S. S. Chawathe, S. Abiteboul, J. Widom. Representing and Querying Changes in Semistructured Data. In *Proc. of ICDE98*: pp.4-13, Orlando, Florida, February 1998.

[11] S. Chawathe. Comparing Hierarchical Data in External Memory. In *Proc. of 25th VLDB*, pages 90-101, Edinburgh, Scotland, UK, September 1999.

[12] S. S. Chawathe, H. Garcia-Molina. Meaningful Change Detection in Structured Data. In *Proc. of 1997 SIGMOD Intl. Conference on Management of Data*: pp. 26-37, Tucson, Arizona, May 1997.

[13] R. Cochrane, K. G. Kulkarni and N. Mendonça Mattos. Active Database Features in SQL3. In *Active Rules in Database Systems*:pp.197-219, N. Paton ed., Springer-Verlag, 1999.

[1] My contribution embraces all the steps of the work, including ideation and definition of active behaviors, extension of XML-GL language and its textual translation and then design of the system architecture.

[14] S. S. Chawathe, A. Rajaraman, H. Garcia-Molina, J. Widom. Change Detection in Hierarchically Structured Information. In *1996 SIGMOD Intl. Conference on Management of Data*: pp. 493-504, Montreal, June 1996.

[15] S. Ceri and J. Widom. Deriving Production Rules for Incremental View Maintenance. In *Proc. of the Seventeenth International Conference on Very Large Data Bases*, pages 577-589, Barcelona, Spain, September 1991.

[16] S. Ceri and J. Widom. Deriving Incremental Production Rules for Deductive Data. In *Information Systems*, 19(6):467-490, 1994.

[17] A. Deutsch, M. Fernandez, D. Florescu, Alon Levy and D. Suciu. XML-QL: A Query Language for XML. In *Proc. of the Query Languages workshop (QL98)*, Cambridge, Mass., Dec. 1998.

[18] P. Fraternali and S. Paraboschi. Chimera: A language for designing rule applications. In *Active Rules in Database Systems*, pages 309–322, Springer-Verlag, Berlin, 1999.

[19] R. Goldman, J. McHugh and J. Widom. From Semistructured Data to XML: Migrating the Lore Data Model and Query Language. In *Proc. of the 2nd International Workshop on Web and Databases (WEDB99)*, Philadelphia, Pennsylvania, June 1999.

[20] A. P. Karadimce, S. D. Urban. Active Rule Termination Analysis: An Implementation and Evaluation of the Refined Triggering Graph Method. In *Journal of Intelligent Information Systems*, Vol. 12, N. 1, April 1999, pp.27-60.

[21] M. Marchiori. *Proc. of QL'98 - The Query Languages Workshop*. Boston, MA, December 1998. Papers available online at http://www.w3.org/Tands/QL/QL98.

[22] J. McHugh and J. Widom. Query Optimization for XML. In *Proc. of 25th VLDB*, pages 315-326, Edinburgh, Scotland, UK, September 1999.

[23] Oracle Corporation, "XML Support in Oracle 8 and beyond", Technical white paper, http://www.oracle.com /xml/documents.

[24] J. Robie, J. Lapp and D. Schach. XML Query Language (XQL). In *Proc. of the Query Languages workshop (QL98)*, Cambridge, Mass., Dec. 1998.

[25] L. van der Voort and A. Siebes. Termination and Confluence of Rule Execution. In *Proc. of 2nd Intl. Conf. on Information and Knowledge Management*, Washington DC, November 1993.

[26] J. Widom. The Starburst Active Database Rule System. In *IEEE Transactions on Knowledge and Data Engineering*, 8(4):583-595, August 1996.

[27] XML 1.0. W3C recommendation, February 1998, http://www.w3.org/TR/REC-xml.

[28] Extensible Stylesheet Language Specification. W3C Working Draft, 21 April 1999, http://www.w3.org /TR/WD-xsl.