Graphical Query Languages for Semi-structured Information

Sara Comai

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

Extended Abstract

The amount of different kinds of data available electronically has increased rapidly in the last years. Beside strongly structured data obeying a rigid schema (e.g. data stored in databases), lots of data which do not present an a priori-known internal structure are spreading. Much of such data, though unstructured at a first sight, present a logical organization and can be considered *semi-structured* [1]. The most known example of semi-structured data is represented by the Web. For such data source, the need of retrieving information is becoming more and more relevant: to date, the only way to extract Web information is to use keyword-based search engines and to browse individual pages. However, the results supplied by common search engines (e.g. Altavista [9], Lycos [19], and so on) are often perceived by users as unsatisfactory: only HTML objects are indexed, the result of Web searching is often a flat list of URLs, without any structure, and, above all, several documents not related to the search term are found. Studies on semi-structured data may solve these deficiencies, by providing methods for extracting information similar to those of the traditional databases, i.e. for *querying* data, taking into account both structure and data semantics.

The aim of this work is proposing description and manipulation languages for querying and restructuring WWW data, and, more in general, semi-structured data. Two query languages are proposed: WG-Log [5] and XML-GL [4][1]. WG-Log is an approach to querying WWW sites and heterogeneous data sources in general, while XML-GL is a sophisticated, but intuitive, visual language for querying documents represented in XML, a recent standard recommended by the W3C for exchanging data and also for publishing Web documents.

Both languages are based on *graphs*. The advantages of adopting a graph representation are manifolds. First of all, graphical representations are more readable than a textual, linear representation of the information. Then, graphs have a high expressive and modeling power, allowing to represent complex objects and any kind of relationships among them. For example, the hypertextual structure of a WWW site can be seen as a graph where nodes correspond to pages, while edges correspond to links between pages. Finally, graph modeling can be exploited also to generate languages for performing inferences, as for example in the object-oriented languages Good [15], Graphlog [6] and G-Log [21]. In such systems, the schemata and the instances of a database are represented by graphs, and the languages defined for them transform these graphs in such a way that the resulting graph represents the result of a query or update, or a constraint.

Querying first-generation Web data with WG-Log. WG-log has its formal basis in G-log [21], a database language to represent and query complex objects by means of directed labeled graphs. In order to represent and query WWW data, WG-Log extends G-Log by including both *logical* and *structural* (navigational) concepts, and by allowing to specify some typical hypermedia design notation [13, 16] like index pages or entry points, which are essentially related to the hypertext *presentation style*. Therefore, it differs from other related approaches like W3QL [17], WebSQL [20] and WebLog [18], which consider only structural information; or from the Tsimmis [12] system, which, on the contrary, represents logical information to be queried with the powerful OQL-like language *Lorel* [2].

WG-Log schemata and instances are graphs. A WG-Log query consists of one or more colored graphs, each representing a *rule*, i.e. an implication as in logic programming. The general structure of a rule is the following: the body (colored red or with thin lines) dictates the conditions that have to be satisfied by the instance, and the head (colored green or with thick lines) specifies the information to be derived. As in logic programming languages, WG-Log rules can be combined into sets and programs. An example of WG-Log rule is depicted in Figure 1. Intuitively, this query establishes an entry-point to the list of restaurant pages offering a menu. WG-Log allows also to formulate views using the same graphical formalism. Views provide flexibility and increase the expressive power of the language: new access structures

[1] I have contributed to the definition of the WG-Log query language and of its architecture, to the definition of the XML-GL syntax and semantics.

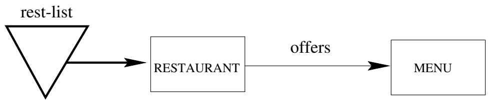

Figure 1: Example of WG-Log query

to a site can be defined and additional information can be inserted; moreover, views may show results in a different way, by restructuring and reorganizing the content of an existing site. In fact, this mechanism can be exploited to reuse existing sites' content to produce new sites. A similar approach has been proposed also by [10], who designed the query language StruQL.

Differently from other related projects, our approach to the representation of semi-structured data is *explicitly schema-based*. Schema-aware query formulation has seldom been applied on the WWW, on the grounds that Web sites are considered too loose and dynamic to allow schematization [23], and thus also to understand the semantics of data. Instead, we believe that a schema-based approach is indeed possible. In fact, the availability of effective design tools for hypermedia has profoundly affected the design of Web sites. Several methodologies such as HDM [13] or RMM [16] provide to the Web site designer some more or less formal means to express data semantics. Based on these methodologies, research prototypes of Web site generators [11] and many commercial Web authoring environments are currently available and widely used. Therefore, most of the data available on the Web are actually semi-structured and a schema of WWW sites can often be derived.

Some other approaches are schema-based: for example, in the *Araneus* project [3] Web site crawling is employed to induce schemata of Web pages, to be combined into a site-wide schema. Such schema is then exploited for building queries: a special-purpose language, *Ulixes* is used to build relational views of WWW data over such schema and the standard SQL language is used to query these views.

The introduction of the concept of schema in our approach takes into account the semi-structured nature of the WWW, but at the same time provides many useful advantages for an efficient processing of the queries. It allows: graceful tolerance of data dynamics, i.e. an easy mechanism for schema updates resulting from instance evolution over time; efficient checking of instance correctness w.r.t. a given schema; direct semantics-aware querying.

Beside the WG-Log language, also a client-server architecture for querying Web data with WG-Log has been proposed. It has also been extended to work in a distributed architecture with heterogeneous data sources (e.g. relational, object-oriented databases, unstructured and semi-structured data sources such as Web sites). Since WG-Log contains many features of semi-structured data and its query language is very flexible, it can be used as a common data model for representing heterogeneous data sources, allowing for a uniform access to such information to both end users and software application layers.

Querying second-generation Web data with XML-GL. The lack of semantics in the WWW data has been addressed also by the W3C community, and recent proposals are providing the concept of schema for WWW data, in a similar way as in WG-Log. Indeed, many new standards try to enrich the WWW data with structure and semantics, for example by introducing meta-data. The most important example is XML (eXtensible Markup Language) [24], a data format designed to make information self-describing. In XML, the tags to be used can be decided by the document creator and thus can be used to say what the information is (e.g. price, quantity, color, and so on). Moreover, XML documents can be associated with a Document Type Definition (DTD) which defines their common syntactic structure, i.e. which tags are allowed inside a document, in which order and how they can be nested within one another. Adding DTDs and semantically meaningful tags to documents opens up the possibility for searching documents on the WEB based on their structure and meaning, as opposed to simple keyword-based search which has characterized the first generation of search engines. Nevertheless, XML is a general markup language for documents, thus it can be adopted not only for representing Web information, but also documental information, i.e. any kind of semi-structured information.

Several languages have been proposed for querying XML, proposing alternative syntaxes and semantics. Examples of existing proposals were discussed in a dedicate workshop [26]: among them, XML-QL [8], Lorel [14] (adapted to query also XML data), XQL [22], XQuery [7], and the W3C standard XSL [25]. Differently from all these approaches, XML-GL is a graphical query language. The motivation for a graphic formalism, as opposed to a conventional linguistic formalisms, lies in the fact that XML documents themselves can be interpreted as graphs; therefore, we have proposed a natural evolution of the WG-log notation as an XML query language, which seems to be the most intuitive and natural. In order to query XML documents, some extensions are needed w.r.t. the data model and the query language of WG-Log, to consider the features of XML (e.g. mutual exclusion or order among elements). However, the query execution process of WG-Log can be easily adapted to fit XML data sources.

An XML-GL query is composed of a pair of graphs (or, for more complex queries, a pair of sets of graphs) drawn side-by-side. The graph on the left side extracts information from the document collection and specifies the conditions that these documents must satisfy. The graph on the right side indicates which elements retrieved in the left-hand part should appear in the result, and constructs or restructures the information to be produced as output. Intuitively, the left hand side graph(s) of an XML-GL query corresponds to the body of a WG-Log rule, while the right hand side of the XML-GL query corresponds to the head of a WG-Log rule graph. An example of XML-GL query is depicted in Figure 2, finding all the *manufacturer* elements having a *model* element nested into it, embedding a *rank* element whose content is <= 10. In the result the *manufacturer* elements with all their content are retained.

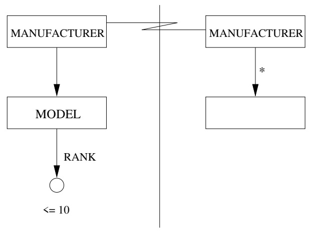

Figure 2: Example of XML-GL query

Ongoing and Future Work. The subjects of this thesis represent an important introductory study from which many extensions, improvements and actual applications may start. Both the proposed languages are based on the G-Log language, which is computationally complete. Therefore, we expect to easily extend the expressive power of WG-Log and XML-GL to take into account other possible features. In particular, for XML we will consider queries over arbitrarily linked documents and metadata, and flexible query interpretation and expansion for non-exact document matching. Indeed, we believe that flexibility is an important issue for querying semi-structured data such as XML. Therefore, we are analyzing the suitability of different semantics: in particular, we intend to provide a fuzzy version of the language, able, on the basis of a parametric semantics, to return documents fitting criteria of similarity to the query conditions, with respect to a specified similarity measure.

Currently, we are also concentrating on the deployment of the XML-GL language within a WEB-based visual environment. The implementation will rely on the engine of the query language Lorel [2]. The query interface will support the automatic display of DTDs and/or documents and a collection of graphic primitives for clipping and dragging schema elements and for incrementally constructing the query graphs.

References

[1] S. Abiteboul. Querying semi-structured data. In *Proc. ICDT'97*, 1997.

[2] S. Abiteboul, D. Quass, J. McHugh, J. Widom, and J. Wiener. The lorel query language for semistructured data. *Int. Journal on Digital Libraries*, 1997.

[3] P. Atzeni, G. Mecca, P. Merialdo, A. Masci, and G. Sindoni. The Araneus Web-Base Management System. In *Proc. Int. Conf. Sigmod'98, Exhibits Program*, pages 544–546, Seattle, June, 1998.

[4] S. Ceri, S. Comai, E. Damiani, P. Fraternali, S. Paraboschi, and L. Tanca. Xml-gl: a graphical language for querying and restructuring xml documents. In *Proc. WWW8*, Toronto, Canada, May 1999.

[5] S. Comai, E. Damiani, R. Posenato, and L. Tanca. A schema-based approach to modeling and querying www data. In *Proc. of FQAS'98*, Roskilde, May 1998. LNAI 1495 - http://thesis.sci.univr.it/Wg-log/.

[6] M.P. Consens and A.O. Mendelzon. The GraphLog visual query system. *SIGMOD Record*, 19(2):388, June 1990.

[7] S.J. DeRose. Xquery: A unified syntax for linking and querying general xml documents. In *Query Languages 98* [26].

[8] A. Deutsch, M. Fernandez, D. Florescu, A. Levy, and D. Suciu. Xml-ql: A query language for xml. In *Proc. QL'98* [26]. http://www.w3.org/TR/1998/NOTE-xml-ql-19980819/.

[9] Digital Equipment Corporation. Altavista search index, 1998. http://www.altavista.digital.com.

[10] M. Fernandez, D. Florescu, A. Levy, and D. Suciu. A query language for a web-site management system. *Sigmod Record*, 26(3):4–11, 1997.

[11] P. Fraternali and P. Paolini. Autoweb: Automatic Generation of Web Applications from Declarative Specifications, 1997. http://www.ing.unico.it/Autoweb.

[12] H. Garcia-Molina and J. Hammer. Integrating and accessing heterogeneous information sources in tsimmis. In *Proc. of ADBIS 97*, St. Petersburg, Russia, 1997.

[13] F. Garzotto, L. Mainetti, and P. Paolini. Hypermedia Design, Analysis and Evaluation Issues. *Communications of the ACM*, 38(8):74–86, 1995.

[14] R. Goldman, J. McHugh, and J. Widom. From semi-structured data to xml: Migrating the lore data model and the query language. In *Proc. Webdb'99*, Philadelphia, Pennsylavanya, June 1999.

[15] M. Gyssens, J. Paredaens, and D. Van Gucht. A graph-oriented object database model. In *Proc. of the Intl. Conf. PODS'90*, 1990.

[16] T. Isakowitz, E.A. Sthor, and P. Balasubranian. RMM: a methodology for structured hypermedia design. *Communications of the ACM*, 38(8):34–44, 1995.

[17] D. Konopnicki and O. Shmueli. W3ql: A query system for the world wideweb. In *Proc. of VLDB'95*, Zurich, 1995.

[18] L. Lakshmanan, F. Sadri, and I. Subramanian. A declarative language for querying and restructuring the web. In *Proc. RIDE-NDS*. IEEE Computer Soc. Press, 1996.

[19] Inc. Lycos. The lycos catalog of the internet, 1998. http://www.lycos.com.

[20] A. Mendelzon, G. Mihaila, and T. Milo. Querying the world wide web. In *Proc. of the Conf. on Parallel and Distributed Information Systems*, Toronto, Canada, 1996.

[21] J. Paredaens, P. Peelman, and L. Tanca. G-log a declarative graph-based language. *IEEE TKDE*, 7:436:453, 1995.

[22] Jonathan Robie, Joe Lapp, and David Schach. Xml query language (xql). In *Query Languages 98* [26].

[23] D. Suciu. Management of semistructured data. *SIGMOD Record*, 26(4), 1997.

[24] W3C. Extensible markup language (xml) 1.0., Feb. 1998. http://www.w3C.org/TR/REC-xml.

[25] W3C. An introduction to xsl, 1998. http://www.w3C.org/Style/xssl.

[26] W3C. *Query Languages 98*, Cambridge, Mass., Dec. 1998.