Beyond XML and RDF: The Versatile Web Query Language Xcerpt*

Sacha Berger
Institute for Informatics,
University of Munich
Oettingenstrasse 67
81543 Munich, Germany
Sacha.Berger@ifi.lmu.de

François Bry
Institute for Informatics,
University of Munich
Oettingenstrasse 67
81543 Munich, Germany
Francois.Bry@ifi.lmu.de

Tim Furche
Institute for Informatics,
University of Munich
Oettingenstrasse 67
81543 Munich, Germany
Tim.Furche@ifi.lmu.de

Benedikt Linse
Institute for Informatics,
University of Munich
Oettingenstrasse 67
81543 Munich, Germany
Benedikt.Linse@ifi.lmu.de

Andreas Schroeder
Institute for Informatics,
University of Munich
Oettingenstrasse 67
81543 Munich, Germany
Andreas.Schroeder@ifi.lmu.de

ABSTRACT

Applications and services that access Web data are becoming increasingly more useful and wide-spread. Current main-stream Web query languages such as XQuery, XSLT, or SPARQL, however, focus only on one of the different data formats available on the Web. In contrast, Xcerpt is a *versatile* semi-structured query language, i.e., a query language able to access all kinds of Web data such as XML and RDF in the same language reusing common concepts and language constructs. To integrate heterogeneous data and as a foundation for Semantic Web reasoning, Xcerpt also provides rules. Xcerpt has a visual companion language, visXcerpt, that is conceived as a mere rendering of the (textual) query language Xcerpt using a slightly extended CSS. Both languages are demonstrated along a realistic use case integrating XML and RDF data highlighting interesting and unique features. Novel language constructs and optimization techniques are currently under investigation in the Xcerpt project (cf. `http://xcerpt.org/`).

Categories and Subject Descriptors:
H.2.3[**Database Management**]: Languages—query languages

General Terms: Languages, Performance

Keywords: XML, RDF, Web, query languages, versatility, Xcerpt

1. XCERPT: A VERSATILE WEB QUERY LANGUAGE

Web querying has received considerable attention from academia and industry culminating in the recent development of the W3C Web query languages XQuery and SPARQL. These main-stream languages, however, focus only on one of the different data formats available on the Web. Integration of data from different sources and in different formats becomes a daunting task that requires knowledge of several query languages and to overcome the impedance mismatch between the query paradigms in the different languages. Xcerpt [7] addresses this issue by garnering the entire language towards versatility in format, representation, and schema of the data, cf. [4]. It is a *semi-structured query language*, but very much unique among such languages (for an overview see [1]):

(1) In its use of a *graph data model*, it stands more closely to semi-structured query languages like Lorel than to recent mainstream XML query languages.

(2) In its aim to address all *specificities of XML*, it resembles more mainstream XML query languages such as XSLT or XQuery.

(3) In using (slightly enriched) *patterns* (or templates or examples) of the sought-for data for querying, it resembles more the "query-by-example" paradigm [8] than mainstream XML query languages using navigational access.

(4) In offering a *consistent extension of XML*, it is able to incorporate access to data represented in richer data representation formats. Instances of such features are element content, where the order is irrelevant, and non-hierarchical relations.

(5) In providing (syntactical) extensions for querying, among others, RDF, Xcerpt becomes a *versatile query language*, cf. [4].

(6) In its strict separation of querying and construction.

visXcerpt [2] is Xcerpt's visual companion language related to it in an unusual way: visXcerpt is a *visual* query language obtained by mere rendering of Xcerpt without changing the language constructs or the runtime system for query evaluation. This rendering is mainly achieved via CSS styling of Xcerpt's constructs. The authors believe that this approach is promising, as it makes those languages easy to learn—and easy to develop.

2. SETTING OF THE DEMONSTRATOR

Excerpts from DBLP[1] and from a computer science taxonomy form the base for the scenario considered in the application. DBLP is a collection of bibliographic entries for articles, books, etc. in the field of Computer Science. DBLP data is a representative for standard Web data using a mixture of rather regular XML content com-

*This research has been funded by the European Commission and by the Swiss Federal Office for Education and Science within the 6th Framework Programme project REWERSE number 506779 (cf. `http://www.rewerse.net/`).

WWW 2006, May 23–26, 2006, Edinburgh, Scotland.
ACM 1-59593-332-9/06/0005.

[1] `http://www.informatik.uni-trier.de/~ley/db/`

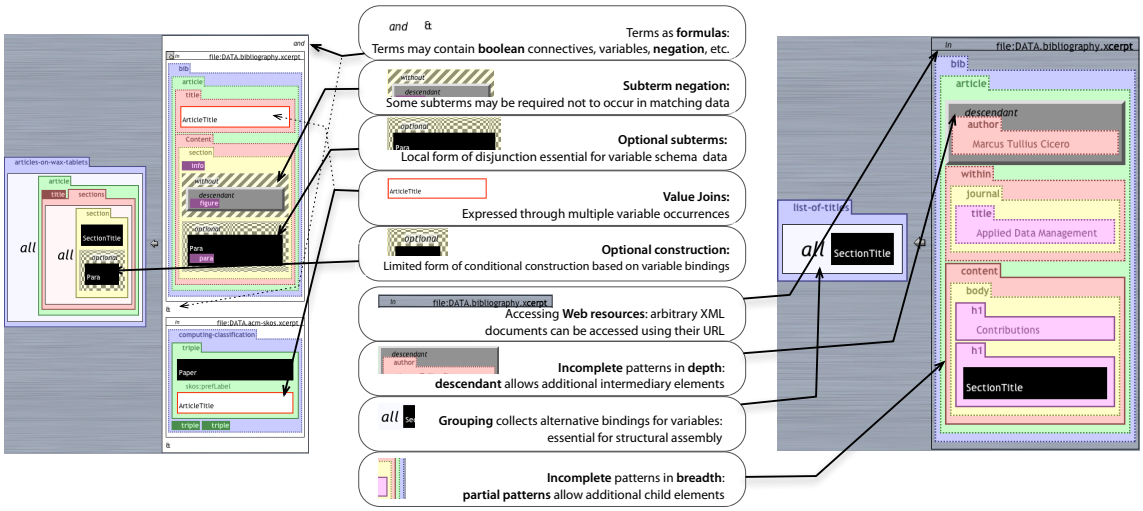

Figure 1: Exemplary visXcerpt Query Patterns

bined with free form, HTML-like information. A small Computer Science taxonomy has been built for the purpose of this demonstration. Very much in the spirit of SKOS, this is a lightweight ontology based on RDF and RDFS. Combining such an ontology as metadata with the XML data of DBLP is a foundation for applications such as community based classification and analysis of bibliographic information using interrelations between researchers and research fields. Realizing such applications is eased by using the integrated Web and semantic Web query language (vis)Xcerpt that also allows reasoning using rules.

3. REALIZING VERSATILITY

Query and construction *patterns* in (vis)Xcerpt are used, both for binding variables in query terms and for reassembling the variables in so-called construct terms. The variable binding paradigm is that of Datalog: the programmer specifies patterns including variables. Interactive behavior of variables in visXcerpt highlights the relation between variables in query and construct terms. Arguably, pattern based querying and constructing together with the variable binding paradigm make complex queries easier to specify and read.

To cope with the semistructured nature of Web data, (vis)Xcerpt query patterns use a notion of incomplete term specifications with optional or unordered content specification. This feature distinguishes (vis)Xcerpt from query languages like Datalog and query interfaces like QBE [8]. Simple, yet powerful textual and visual constructs of incompleteness are presented in the demonstrator application, cf. Figure 1 showing two exemplary visual query patterns and a breakdown of used language constructs.

An important characteristic of (vis)Xcerpt is its rule-based nature: (vis)Xcerpt provides rules very similar to SQL views. Arguably, rules or views are convenient for a logical structuring of complex queries. Thus, in specifying a complex query, it eases the programming and improves the program readability to specify (abstract) rules as intermediate steps—very much like procedures in conventional programming. Another aspect of rules is the ability to solve simple reasoning tasks.

Referential transparency and answer closedness are essential properties of Xcerpt and visXcerpt, surfacing in various parts of the demonstration. They are two precisely defined traits of the rather vague notion of "declarativity". Referential transparency means that within a definition scope all occurrences of an expression have the same value, i.e., denote the same data. Answer-closedness means that replacing a sub-query in a compound query by a possible single answer always yields a syntactically valid query. Referentially transparent and answer-closed programs are easy to understand (and therefore easy to develop and to maintain), as the unavoidable shift in syntax from the data sought for to the query specifying this data is minimized.

4. EFFECTIVENESS AND EFFICIENCY

Currently, two main threads are considered in the Xcerpt project: (1) A careful review of language constructs is underway that aims at an improved effectiveness for query authoring, cf. [6]. Related is a better support for RDF, including proper handling of b-nodes in results and incomplete data specifications. Furthermore, a type system [3] for Xcerpt is under development that eases error detection and recovery. (2) Novel evaluation methods for Xcerpt, enabled by high-level query constructs, are being investigated. Xcerpt's pattern matching is based on simulation unification. An efficient algorithm of simulation unification that is competitive with current main-stream Web query languages both in worst-case complexity and practical performance is described in [5]. Optimizations of the rule chaining algorithm are also investigated, partially based on dependency analysis provided by the above mentioned type system.

5. REFERENCES

[1] J. Bailey, F. Bry, T. Furche, and S. Schaffert. Web and Semantic Web Query Languages: A Survey. *Reasoning Web Summer School 2005*. Springer, 2005.

[2] S. Berger, F. Bry, S. Schaffert, and C. Wieser. Xcerpt and visXcerpt: From Pattern-Based to Visual Querying of XML and Semistructured Data. In *29th Intl. Conf. on Very Large Data Bases*, 2003.

[3] S. Berger, E. Coquery, W. Drabent, and A. Wilk. Descriptive Typing Rules for Xcerpt. In *Proc. of Workshop on Principles and Practice of Semantic Web Reasoning*. 2005.

[4] F. Bry, T. Furche, L. Badea, C. Koch, S. Schaffert, and S. Berger. Querying the Web Reconsidered: Design Principles for Versatile Web Query Languages. *Journal of Semantic Web and Information Systems*, 1(2), 2005.

[5] F. Bry, A. Schroeder, T. Furche, and B. Linse. Efficient Evaluation of n-ary Queries over Trees and Graphs. Submitted for publication, 2006.

[6] T. Furche, F. Bry, and S. Schaffert. Initial Draft of a Language Syntax. Deliverable I4-D6, REWERSE, 2006.

[7] S. Schaffert and F. Bry. Querying the Web Reconsidered: A Practical Introduction to Xcerpt. In *Extreme Markup Languages*, 2004.

[8] M. M. Zloof. Query-by-Example: A Data Base Language. *IBM Systems Journal*, 16(4):324–343, 1977.