Report on the Seventh EDBT Summer School: XML & Databases

Riccardo Torlone, Paolo Atzeni
Università Roma Tre
Dipartimento di Informatica e Automazione
Via della Vasca Navale, 79
00146 Roma Italy
{torlone,atzeni}@dia.uniroma3.it

Introduction

The Seventh EDBT Summer School 2004 was held in Santa Margherita di Pula, Sardinia, Italy, September 6–10, 2004. It continued a successful series of international schools promoted by the EDBT Association and dedicated to the dissemination of latest advances in database management. Indeed, the EDBT Association (a Europe-based non-profit organization also known as the EDBT Endowment) has the purpose of promoting and supporting the progress and the study of the database technology and application. Its two major initiatives are the EDBT Conference (International Conference on Extending Data Base Technology), held every two years since 1988, and the EDBT Summer School, held almost with the same regularity, since 1991. Previous schools took place in Alghero (Italy) in 1991, Leysin (Switzerland) in 1993, Gubbio (Italy) in 1995, Capri (Italy) in 1997, La Baule (France) in 1999, and Cargèse (France) in 2002. Some information on the previous editions is available at the EDBT Association site (`www.edbt.org`), which also illustrates the goals and structure of the association.

The School topics and content

Whenever a Summer School is organized, there is always a discussion on how much it should be focused, and EDBT has tried both extremes, editions with a very broad program and editions with a very focused one. In this edition, there was consensus within the scientific committee on a topic that would characterize the school, at the same time being broad, with aspects that would range from theory to technology and applications, and with a clear interest in the research community as well as in industry.

This led to the choice of a very hot theme, "XML & Databases," motivated by the growing popularity and adoption of XML for data manipulation. As more and more organizations employ XML within their information management and data exchange strategies, classical database issues arise and new challenges appear. On the one hand, XML data need to be efficiently stored, queried, indexed and manipulated. On the other hand, methods and tools for effectively integrating heterogeneous data sources, for sharing information across different tasks, for exchanging data between multiple organizations, need to be defined.

The School was organized by Riccardo Torlone, supported by Paolo Atzeni who mainly acted as the interface with the EDBT Association and with the organizers of the previous editions. The program was set up by a scientific committee composed by Serge Abiteboul, Paolo Atzeni, Christine Collet, Torsten Grust, Riccardo Torlone, and Maurice van Keulen.

The school program included eight seminars of leading scientists of the international database systems community, addressing "XML & Database" topics from both a theoretical and a practical perspective. These seminars are briefly illustrated in the following sections. Abstracts and slides for the lectures are available for download from Web site of the

school: edbtss04.dia.uniroma3.it. The program also hosted a number of lively presentations given by Ph.D. students attending the school, who illustrated their current research activity.

The tutorials

General Techniques for XML Query processing

XQuery is used in a variety of different products. Examples to date include XML database systems, XML document repositories, XML data integation, workflow systems, and publish and subscribe systems. In addition, XPath of which XQuery is a superset is used in various products such as Web browsers. Although the W3C XQuery specification has not yet attained recommendation status, and the definition of the language has not entirely stabilized, a number of alternative proposals to implement and optimize XQuery have appeared both in industry and in the research community. Given the wide range of applications for which XQuery is applicable, a wide spectrum of alternative techniques have been proposed for XQuery processing.

The goal of this tutorial, presented by Dana Florescu was to give an overview of the existing approaches to process XQuery expressions and to give details of the most important techniques. Specifically, she discussed methods for internal XQuery representation, optimization strategies based on query rewriting, models for XML data storage, and efficient schemes for query execution.

XML & Data integration

The invention of XML injected additional fuel into the data integration fire. XML enabled researchers and practitioners of data integration to focus on issues of query processing and semantic heterogeneity, rather than low-level details involving data sharing formats and building wrappers.

This tutorial, given by Alon Halevy (University of Washington, USA), began by reviewing several architectures for data sharing and integration and the query processing issues that had to be solved in building them. Alon highlighted the novel challenges faced because of the need to process XML data in data integration systems and then discussed recent work on bridging semantic heterogeneity between data sources, and described current challenges for the field. These include mediator languages and techniques, query reformulation, peer data management, schema matching, and model management.

XML Query Processing: Storage and Query Model Interplay

XML data management has received a lot of attention from various perspectives. A variety of XML storage schemes have been proposed, many of which take advantage of existing query processing systems. The most notable family is that of relational stores for XML, in which XML query processing is delegated to a relational query processor. Most recently, complex native XML storage systems are being developed, and enhanced with indexing schemes, view mechanisms, and early attempts at cost-based query optimization. Any XML storage scheme, though, is only as good as the XML query processing performance it can achieve. Thus, after years of efforts on XML storage as a goal per se, new approaches start from the current winner in terms of XML querying — XQuery — and build from that point storage strategies that are likely to gracefully support XQuery query processing.

The tutorial, given by Ioana Manolescu (INRIA, France), provided a comparative, evolutionary view of these approaches, emphasizing the interplay between XML storage and query execution models. Ioana covered the fundamental elements of XML storage and execution models proposed in the last six years, systematizing them to encompass current and foreseeable future efforts. In particular, she illustrated the differences between the relational and native approach to XML data management.

Fully (or Purely) Relational XPath and XQuery Processors

Existing relational database technology has great potential for being able to manage the ever grow-

ing volumes of XML data. Query efficiency can be achieved by choosing an appropriate encoding. It has been shown that encodings based on assigning preorder and postorder ranks to each XML node have properties that can be exploited well in an ordinary RDBMS. Such relational encodings of XML are the basis of this tutorial.

In this tutorial, Maurice van Keulen (University of Twente, The Netherlands) and Torsten Grust (University of Konstanz, Germany) discussed how to exploit encoding properties and reuse DBMS functionality for various aspects of XML data management. They also showed how XPath queries can be evaluated using plain SQL. The performance of such queries can be significantly improved if special-purpose joins are employed. Efficient execution of full XQuery provides additional challenges. Maurice and Torsten presented a compilation technique that uses a rather standard relational algebra as an intermediary. Besides querying, they also touched upon XML document loading and maintenance.

This tutorial incorporated prototype demonstrations and assignments to make the theory more concrete and to strengthen understanding. This was an exciting experience for attendees.

Formal Methods for XML: Algorithms & Complexity

XML processing has to deal with at least with four kinds of tasks:

- validation: checking whether a document has the desired structure;
- navigation: selecting a set of positions given position via paths with specified properties;
- transformation: constructing a new document from a given one by applying a set of rules;
- querying: extracting information from a document.

For these tasks suitable languages have been designed, DTD & XML Schema, XPath, XSLT and XQuery, respectively. But they also have been studied from a more formal point of view by using well known concepts from database theory, logic and the theory of formal languages.

This tutorial, given by Thomas Schwentick (Philipps-Universität Marburg, Germany) illustrated the main aspects of these formal foundations of XML languages. Thomas particularly emphasized the expressive power of the underlying formalisms and the complexity of fundamental algorithmic tasks, like evaluation of XPath queries, checking containment of schemas or type checking transformations. It turns out that while schema languages and XPath are well understood in terms of complexity (many tasks can be solved efficiently for large fragments of these languages), a theory for XQuery still has to be developed.

Processing XQuery Now!

XQuery is a powerful language, which can be used not only to query XML databases, but also in new XML applications: to set up a Web-service, for data integration, as a component of the semantic Web infrastructure, or for plain file processing. Because it is a complex language, and because it can be used in so many different environments, building an XQuery processor is a challenging task.

The objective of this lecture, given by Jérôme Siméon (IBM Research, USA) was to explain the foundation of XQuery implementations which are both complete, and efficient. Jérôme started by giving an overview of relevant compilation techniques, for both query languages and programming languages. Then he described the architecture for a complete implementation of XQuery, walking also through the compilation of a query from parsing to evaluation.

Full-Text Querying in XML, Sihem Amer-Yahia

A key benefit of XML is its ability to represent a mix of structured and unstructured (text) data. Querying XML data is a well-explored topic with powerful database-style query languages such as XPath and XQuery set to become W3C standards. An equally compelling paradigm for querying XML documents

is full-text search. Although current XML query languages such as XPath and XQuery can express rich queries over structured data (e.g., navigate in document structure, construct new elements), they can only express very rudimentary queries over text.

In this tutorial, Sihem Amer-Yahia (AT&T Labs Research, USA) presented TeXQuery, a full-text extension to XQuery. TeXQuery provides a rich set of fully composable full-text search primitives, such as Boolean connectives, phrase matching, proximity distance, stemming and thesauri. This language enables users to seamlessly query over both structured and text data. It supports a flexible scoring construct that can be used to score query results based on full-text predicates. TeXQuery is the precursor of the full-text language extension to XPath 2.0 and XQuery 1.0 that is being developed at the W3C.

Management and Composition of Web services

The Web services paradigm promises to enable rich, flexible, and dynamic interoperation of highly distributed and heterogeneous web-hosted services. Substantial progress has already been made towards this goal and industrial technology. Several research efforts are already underway that build on or take advantage of the paradigm but there is still a long way to go, especially given the ostensible long-term goal of enabling the automated discovery, composition, enactment, and monitoring of collections of Web services working to achieve a specified objective.

In this tutorial, Richard Hull (Bell Laboratories, Lucent Technologies, USA) provided the groundwork to address the fundamental questions on the design and analysis of composite Web services. Specifically, are existing tools for design and analysis of software systems sufficient for Web services, or are new techniques needed to handle the novel aspects of the Web services paradigm? This raises a variety of questions, several of which are relevant for the database research community. These include: What is the right way to model Web services and their compositions? What is the right way to query them in order to support automated composition and analysis algorithms? And how can the data management aspects of composite Web services be incorporated into current Web services standards?

Conclusion

The call for attendance attracted a record of 164 applications from all over the world, out of which 76 participants from sixteen different countries were selected. With the help of sponsors, we were able to assign fellowships to thirty of them. In particular, great support was offered by INTAS (International Association for the promotion of cooperation with scientists from the New Independent States of the former Soviet Union) and, as in most of the previous editions, some fellowships were provided by the EDBT Association. Università Roma Tre, Dipartimento di Informatica e Automazione and IASI-CNR also provided some financial and organizational support.

The organization of the school required the efforts of many dedicated people. In particular, we would like to thank Torsten Grust and Maurice van Keulen, for playing a very active role in the scientific organization of this event, and the members of the scientific committee, for their precious suggestions in the composition of such a strong and exciting program.

Special thanks go to the speakers, for their very interesting tutorials and exciting presentations. Finally, all the participants, without whom the success of the event would not have been possible, deserve our deepest gratitude.