Advanced Applications and Structures in XML Processing: Label Streams, Semantics Utilization and Data Query Technologies

Changqing Li
*Duke University, USA*

Tok Wang Ling
*National University of Singapore, Singapore*
Detailed Table of Contents

Foreword .............................................................................................................................................. xvi
Preface ...................................................................................................................................................... xviii
Acknowledgment ..................................................................................................................................... xxii

Section 1
XML Data Management

This section discusses different XML data management techniques, including XML native storage, management in object relational database systems, compression, and benchmark.

Chapter 1
XML Native Storage and Query Processing ...................................................................................... 1
    Ning Zhang, Facebook, USA
    M. Tamer Ozsu, University of Waterloo, Canada

This chapter reviews different native storage formats and query processing techniques that have been developed in both academia and industry. Among the XML data management issues, storage and query processing are the most critical ones with respect to system performance. Different storage schemes have their own pros and cons. Therefore, based on their own requirements, different systems adopt different storage schemes to tradeoff one set of features over the others. Various XML indexing techniques are also presented since they can be treated as specialized storage and query processing tools.

Chapter 2
XML Data Management in Object Relational Database Systems .................................................. 18
    Zhen Hua Liu, Oracle Corporation, USA
    Anguel Novoselsky, Oracle Corporation, USA
    Vikas Arora, Oracle Corporation, USA

This chapter describes the XML data management capabilities in Object Relational DBMS (ORDBMS), various design approaches and implementation techniques to support these capabilities, as well as the pros and cons of each design and implementation approach. Key topics such as XML storage, XML
Indexing, XQuery and SQL/XML processing, are discussed in depth presenting both academic and industrial research work in these areas.

Chapter 3
XML Compression

Chin-Wan Chung, Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea
Myung-Joe Park, Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea
Jihyun Lee, Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea

This chapter provides a better understanding on relevant theoretical frameworks and an up-to-date research trend of the XML compression. Existing XML compression techniques are classified and examined based on their characteristics and experimental evaluations. Also, according to the comprehensive analysis, appropriate XML compression techniques for different environments are recommended. Furthermore, some future research directions on the XML compression are presented.

Chapter 4
XML Benchmark

Ke Geng, University of Auckland, New Zealand
Gillian Dobbie, University of Auckland, New Zealand

Benchmarks are widely used in database-related research, helping users choose suitable database management systems and helping researchers evaluate their new methods. Recently benchmarks for XML have been designed to support the development of XML tools and systems. In this chapter, XML benchmarks are categorized into four groups: application benchmark, micro benchmark, XML generator and real dataset. Characteristics of each benchmark are discussed and compared. Finally, the future direction of XML benchmarks are discussed.

Section 2
XML Index and Query

This section presents the XML index and query techniques, including XML index structures, labeling, keyword search, and query optimization.

Chapter 5
Index Structures for XML Databases

Samir Mohammad, Queen’s University, Canada
Patrick Martin, Queen’s University, Canada

This chapter gives a brief history of the creation and the development of the XML data model. Then it discusses the three main categories of indexes proposed in the literature to handle the XML semi-structured data model. Finally, it discusses limitations and open problems related to the major existing indexing schemes.
Chapter 6
Labeling XML Documents ................................................................. 125
Jiaheng Lu, School of Information and DEKE, MOE, Renmin University of China, China
Liang Xu, School of Computing, National University of Singapore, Singapore
Tok Wang Ling, School of Computing, National University of Singapore, Singapore
Changqing Li, Duke University, USA

This chapter shows how to extend the traditional prefix labeling scheme to speedup XML query processing. In addition, for XML documents that are updated frequently, many labeling schemes require relabeling which can be very expensive. A lot of research interest has been generated on designing dynamic XML labeling schemes. Making labeling schemes dynamic turns out to be a challenging problem and many of the approaches proposed only partially avoid relabeling. This chapter describes some recently emerged dynamic labeling schemes that can completely avoid relabeling, making efficient update processing in XML database management systems possible.

Chapter 7
Keyword Search on XML Data .......................................................... 143
Ziyang Liu, Arizona State University, USA
Yi Chen, Arizona State University, USA

This chapter describes the importance, challenges and future directions for supporting XML keyword search. It presents and compares representative state-of-the-art techniques from multiple aspects, including identifying relevant keyword matches and an axiomatic framework to evaluate different strategies, identifying other relevant data nodes, ranking schemes, indexes and materialized views, and result snippet generation. These studies enable casual users to easily access XML data without the need to learn XPath/XQuery and data schemas, and yet to obtain high-quality results. It also summarizes the possible future research directions of XML keyword search.

Chapter 8
A Framework for Cost-Based Query Optimization in Native XML Database Management Systems ................................................. 160
Andreas M. Weiner, University of Kaiserslautern, Germany
Theo Harder, University of Kaiserslautern, Germany

This chapter introduces an extensible and rule-based framework for cost-based native XML query optimization, which supports a large fragment of XQuery 1.0—the predominant query language in XML databases. For the evaluation of XQuery expressions, the framework can exploit around 50 physical operators. It can be configured in such a way, that different types of query optimization techniques can be compared with respect to their sufficiency for cost-based XQuery optimization under equal and fair conditions. Therefore, it relies on the native XML database management system XTC (XML Transaction Coordinator). In combination with a cost model for constraining the search space, the framework can be turned into a full-fledged XQuery optimizer in the future.
Section 3
XML Stream Processing, Publish/Subscribe, and P2P

This section is about some advanced topics of XML, including XML stream processing, publish/subscribe system, and P2P system.

Chapter 9
XML Stream Processing: Stack-Based Algorithms ................................................................. 184
Junichi Tatemura, NEC Laboratories America, USA

Description: This chapter reviews recent advances on stream XML query evaluation algorithms with stack-based encoding of intermediary data. Originally proposed for disk-resident XML, the stack-based architecture has been extended for streaming algorithms for both single and multiple query processing, ranging from XPath filtering to more complex XQuery. The key benefit of this architecture is its succinct encoding of partial query results to avoid exponential enumeration. In addition, the chapter discusses opportunities to integrate benefits demonstrated in the reviewed work.

Chapter 10
Content-Based Publish/Subscribe System for XML Data .................................................... 207
Yuan Ni, IBM China Research Lab, China
Chee-Yong Chan, National University of Singapore, Singapore

This chapter focuses on the content-based publish/subscribe system for XML data. Firstly, the fundamental concepts, i.e. publisher, subscriber and XML routers, in the content-based publish/subscribe system for XML data is introduced. After that, the chapter presents two important issues, i.e. the efficiency of the system and the functionalities that are supported by this system, to consider in content-based publish/subscribe for XML data, and discussed the approaches that address these problems. Finally, the chapter pointed out some potential directions in the content-based publish/subscribe for XML data.

Chapter 11
Content-Based XML Data Dissemination .............................................................................. 227
Guoli Li, University of Toronto, Canada
Shuang Hou, University of Toronto, Canada
Hans-Arno Jacobsen, University of Toronto, Canada

This chapter describes the XML-based data dissemination networks. In these networks XML content is routed from data producers to data consumers throughout an overlay network of content-based routers. Routing decisions are based on XPath expressions (XPEs) stored at each router. To enable efficient routing, while keeping the routing state small, this chapter introduces advertisement-based routing algorithms for XML content, presents a novel data structure for managing XPEs, especially apt for the hierarchical nature of XPEs and XML, and develops several optimizations for reducing the number of XPEs required to manage the routing state. The experimental evaluation shows that the algorithms and optimizations reduce the routing table size by up to 90%, improve the routing time by roughly 85%, and reduce overall network traffic by about 35%. Experiments running on PlanetLab show the scalability of this approach.
This chapter presents XP2P, a framework for fragmenting and managing XML data over structured peer-to-peer networks. XP2P is characterized by an innovative mechanism for fragmenting XML documents based on meaningful XPath queries, and novel fingerprinting techniques for indexing and looking-up distributed fragments based on Chord's DHT. Efficient algorithms for querying distributed fragments over peer-to-peer networks are also presented and experimentally assessed against both synthetic and real XML data sets. A comprehensive analysis of future research directions on XML data management over peer-to-peer networks completes the contribution of the chapter.

**Section 4**

**XML Query Translation and Data Integration**

This section describes how to normalize and translate XML queries and how to do XML data integration.

**Chapter 13**

Normalization and Translation of XQuery

This chapter argues for an algebraic optimization and evaluation technique for XQuery as it allows people to benefit from experience gained with relational databases. An algebraic XQuery processing method requires a translation into an algebra representation. While many publications already exist on algebraic optimizations and evaluation techniques for XQuery, an assessment of translation techniques is required. Consequently, the chapter gives a comprehensive survey for translating XQuery into various query representations. The chapter relates these approaches to the way normalization and translation is implemented in Natix and discusses these two steps in detail.

**Chapter 14**

XML Data Integration: Schema Extraction and Mapping

This chapter discusses the challenges and techniques in XML Data Integration. It first presents a four step outline, illustrating the steps involved in the integration of XML data. This chapter, then, focuses on the first two of these steps: schema extraction and data/schema mapping; the next chapter focuses on the remaining steps: merging, query processing and conflict resolution.
Chapter 15
XML Data Integration: Merging, Query Processing and Conflict Resolution ........................................................................... 333
Yan Qi, Arizona State University, USA
Huiping Cao, Arizona State University, USA
K. Selcuk Candan, Arizona State University, USA
Maria Luisa Sapino, University of Torino, Italy

This chapter continues from the previous chapter to discuss the merging, query processing and conflict resolution steps in XML data integration. Specifically, merging integrates multiple disparate (heterogeneous and autonomous) input data sources together for further usage, while query processing is one main reason why the data need to be integrated in the first place. Besides, when supported with appropriate user feedback techniques, queries can also provide contexts in which conflicts among the input sources can be interpreted and resolved. This chapter also discusses two alternative ways XML data/schema can be integrated: conflict-eliminating (where the result is cleaned from any conflicts that the different sources might have with each other) and conflict-preserving (where the resulting XML data or XML schema captures the alternative interpretations of the data).

Section 5
XML Semantics Utilization and Advanced Application

This section includes how to utilize semantics to process XML update and query, as well as XML application on web service.

Chapter 16
Document and Schema XML Updates.............................................................................................................................................. 361
Dario Colazzo, Laboratoire de Recherche en Informatique (LRI-CNRS), Université de Paris-Sud, France
Giovanna Guerrini, DISI – Università degli Studi di Genova, Italy
Marco Mesiti, DICO – Università degli Studi di Milano, Italy
Barbara Olboni, DI – Università degli Studi di Verona, Italy
Emmanuel Waller, Laboratoire de Recherche en Informatique (LRI-CNRS), Universite de Paris-Sud, France

This chapter starts from the update primitives supported by current language proposals, and deals with the data management issues that arise when documents and schema are updated and new versions created. Specifically, the chapter will provide a review of various proposals for XML document updates, their different semantics and their handling of update sequences, with a focus on the XQuery Update proposal. Approaches and specific issues concerned with schema updates are then reviewed. Document and schema versioning is considered and a review of the degree and limitations of update support in existing DBMSs is discussed.
Chapter 17
Integration of Relational and Native Approaches to XML Query Processing.......................... 385
Huayu Wu, National University of Singapore, Singapore
Tok Wang Ling, National University of Singapore, Singapore

This chapter proposes a hybrid approach which integrates the two classes of exiting XML query processing approaches, i.e. the relational approach and the native approach (inverted lists), and it wants to inherit the advantages and solve the problems in the two approaches. By performing content search using relational tables before structural pattern matching, this approach not only properly solves the value constraints, but also simplifies the pattern matching process, thus improves the query processing efficiency. The chapter also proposes three optimizations based on semantic information of object. Once more object information is known in a given XML document, the approach can use such semantics to improve relational tables, and to get a better performance.

Chapter 18
XML Query Evaluation in Validation and Monitoring of Web Service Interface Contracts .......... 406
Sylvain Hallé, University of California, Santa Barbara, USA
Roger Villemaire, Université du Québec à Montréal, Canada

This chapter describes that web applications communicate with web services through the exchange of sequences of XML messages representing requests and responses for specific operations. The available documentation for a service constitutes what is called an interface contract, which specifies the acceptable messages and sequences of messages that can be exchanged with this service. By capturing incoming and outgoing messages during an actual execution of an application and aligning them into an XML document, it is possible to determine whether a specific execution trace satisfies an interface contract. In particular, the chapter shows how sequential constraints expressed in an extension of Linear Temporal Logic with first-order quantification on data can be verified by translating them into equivalent XQuery expressions on XML trace documents.

Compilation of References ............................................................................................................. 425

About the Contributors ..................................................................................................................... 459

Index .................................................................................................................................................. 467