

Searchable Storage in Cloud Computing

Yu Hua · Xue Liu

Searchable Storage in Cloud Computing

Yu Hua
Huazhong University of Science
and Technology
Wuhan, Hubei, China

Xue Liu
McGill University
Montreal, QC, Canada

ISBN 978-981-13-2720-9 ISBN 978-981-13-2721-6 (eBook)
<https://doi.org/10.1007/978-981-13-2721-6>

Library of Congress Control Number: 2019930156

© Springer Nature Singapore Pte Ltd. 2019

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Foreword

Exabyte- or Zettabyte-scale storage systems are one of the major challenges facing cloud computing because it is nontrivial to store and manage huge amount of data stemming from many data-intensive applications such as business transactions, scientific computing, social network webs, mobile applications, and information visualization, in which cloud computing serves as an important infrastructure. As a matter of fact, we are generating and storing much more data than ever before and this trend continues at an accelerated pace. What is more, the data in storage systems need to be searchable, which becomes an important function and the foundation of many operations in cloud computing. Therefore, researchers and engineers must find ways to quickly and accurately find the right information, query the data, extract the knowledge, and at the same time, reduce computational and transmission overheads.

Yu Hua and Xue Liu have been the leading contributors to searchable storage research. This book provides the backgrounds of searchable storage in cloud computing systems. It covers the basic concepts and implementation techniques such as hash functions, semantic namespace, exact and approximate queries, data analytics, multicore computation, and data cube in the networks in a chapter-by-chapter presentation. Furthermore, these topics are presented with examples and real-world applications, and they are readable and informative.

Researchers and engineers in the fields of cloud computing and storage systems will find it interesting to learn about the searchable techniques due to the ease of use and simplicity with the aid of many open-source codes. Readers will also obtain the needed backgrounds and gain new insights and implementation experiences in data structures and computer systems.

Personally, I found this book very informative and stimulating, and believe that readers will gain lots of new insights from this book.

Hong Kong
December 2018

John C. S. Lui

Preface

In the era of cloud computing and big data, storage systems are very important and have become the infrastructure of many real-world applications. However, existing storage systems fail to offer cost-efficient searching service due to overlooking the semantics behind massive data. This book introduces and presents new schemes for exploring and exploiting the searchable storage via cost-efficient semantic hashing computation. The contents in this book include basic hashing structures (Bloom Filters, Locality Sensitive Hashing, Cuckoo Hashing), semantic storage systems, and searchable namespace, which support multiple applications, such as cloud backups, exact and approximate queries, and image sharing. More importantly, all these mentioned structures and techniques have been really implemented to support real-world applications, some of which offer open-source codes for public use.

This book consists of eight chapters to present the core ideas and methodology for searchable storage in cloud computing.

Chapter 1 introduces the backgrounds of searchable storage, which requires multiple queries, including point query, range query, cover query, and bound query. The proposed Bloom filter based R-tree can efficiently support the above queries in a cost-efficient manner.

Chapter 2 describes the scalable metadata management in the context of cloud computing via efficient hash computation. A real system prototype, i.e., the Group-Hierarchical Bloom Filter Array, is used as a case in point to comprehensively demonstrate the significant performance improvements and scalability.

Chapter 3 presents an important component, i.e., the semantic-aware namespace, which serves as the infrastructure of large-scale cloud computing platforms. This namespace exploits the semantic correlation behind the multi-dimensional attributes of files to efficiently support not only namespace construction but also dynamic evolution.

Chapter 4 summarizes the exact-matching queries for metadata via efficient semantic grouping from the views of both users and systems. This scheme is able to provide fast and exact-matching query services. In the meantime, the consistency can be guaranteed by using versioning techniques.

Chapter 5 provides the approximate queries by using locality sensitive Bloom filters. Unlike conventional Bloom filters, the proposed space-efficient filters offer approximate query services with the constant-scale complexity.

Chapter 6 introduces the near real-time searchable analytics for images. This technique well explores and exploits the features of images to configure semantic-aware methodology. The analytics are accurate and in the meantime, the memory systems deliver high performance.

Chapter 7 shows computing infrastructure of searchable storage in cloud computing. By leveraging the data similarity, we can accurately place suitable data within the hierarchical caches and memory for multicore processors, thus obtaining significant performance improvements.

Chapter 8 presents semantic-aware data cube to provide fast online queries for the cloud. Correlation-based representation is used to support partial materialization and efficient incremental updates.

This book is for computer scientists, computer engineers, and others who are interested in the cloud computing from the perspective of searchable storage.

Acknowledgement

This book is supported in part by National Natural Science Foundation of China (NSFC) under Grant No. 61772212.

Wuhan, China
Montreal, Canada
November 2018

Yu Hua
Xue Liu

Contents

1	Backgrounds of Searchable Storage	1
1.1	Introduction	1
1.2	Related Work	3
1.3	BR-Tree Structure	4
1.3.1	Proposed Architecture	4
1.3.2	Load Balance	5
1.3.3	Example for Multiple Queries	6
1.4	Local Operations on a BR-Tree	8
1.4.1	Item Insertion	8
1.4.2	Item Deletion	9
1.4.3	Point Query	9
1.4.4	Range Query	11
1.4.5	Cover Query	11
1.4.6	Bound Query	12
1.4.7	Simple Summary of Multiple Queries	13
1.4.8	Comparisons of BR-Tree and Other State-of-the-Art Structures	14
1.5	Distributed BR-Trees	14
1.5.1	Distributed BR-Tree Structure	15
1.5.2	Operations on Distributed BR-Tree	16
1.5.3	Update Stale Replicas	16
1.6	Prototype Implementation	17
1.6.1	Constructing Experiment Environment	17
1.6.2	Performance Evaluation	18
1.7	Conclusion	20
	References	21

2	Hashing Computation for Scalable Metadata	23
2.1	Introduction	23
2.1.1	Motivations	24
2.1.2	Contributions	26
2.2	G-HBA Design	27
2.2.1	Dynamic and Adaptive Metadata Management	27
2.2.2	Group-Based HBA Scheme	28
2.2.3	Critical Path for G-HBA Multi-level Query Service	30
2.2.4	Updating Replicas	31
2.3	Dynamic and Adaptive Group Reconfigurations	32
2.3.1	Lightweight Migration for Group Reconfiguration	32
2.3.2	Group Splitting and Merging	32
2.3.3	Optimal Group Configuration	33
2.4	Performance Evaluation	34
2.5	Prototype Implementation and Analysis	37
2.6	Related Work	40
2.7	Conclusion	41
	References	42
3	The Component of Searchable Storage: Semantic-Aware Namespace	45
3.1	Introduction	45
3.2	Backgrounds and Problem Statement	48
3.2.1	Multi-dimensional Attributes	48
3.2.2	Problem Statement	50
3.3	Design and Implementation	51
3.3.1	An Architectural Overview	51
3.3.2	Semantic Correlation Identification	52
3.3.3	Data Storage Structure	53
3.3.4	Namespace Construction	54
3.3.5	Dynamic Evolution	56
3.4	Performance Evaluation	57
3.4.1	Experimental Setup	57
3.4.2	Results and Discussions	59
3.5	Related Work	62
3.6	Conclusion	63
	References	63
4	Semantic-Aware Metadata Organization for Exact-Matching Queries	67
4.1	Introduction	67
4.1.1	Semantic Correlation	70
4.1.2	SmartStore's Contributions	71

4.2	SmartStore System	72
4.2.1	Overview	72
4.2.2	User View	73
4.2.3	System View	74
4.2.4	Configuration to Match Query Patterns	75
4.3	Semantic-Aware Design and Analysis	75
4.3.1	The LSI Tool	76
4.3.2	Semantic Grouping	77
4.3.3	PLSA for Complexity Reduction	78
4.3.4	Complexity Analysis	79
4.4	System Configurations	80
4.4.1	Grouping in a Semantic R-tree	80
4.4.2	Insertion	80
4.4.3	Deletion	81
4.4.4	Query Approaches	81
4.5	Key Design Issues	83
4.5.1	Node Split and Merge	83
4.5.2	Mapping of Index Units	83
4.5.3	Multi-mapping of Root Node	84
4.5.4	Selection of Multi-dimensional Attributes	85
4.5.5	Consistency Guarantee via Versioning	85
4.6	Performance Evaluation	88
4.6.1	Prototype Implementation	88
4.6.2	Performance Comparisons Between SmartStore and Baseline Systems	91
4.6.3	Overhead and Efficiency of Versioning	92
4.7	Related Work	94
4.7.1	Content-Based Search	94
4.7.2	Directory-Based Subtree Partitioning	94
4.8	Conclusion	95
	References	95
5	Locality-Sensitive Bloom Filter for Approximate Membership Query	99
5.1	Introduction	99
5.1.1	Motivations	100
5.1.2	Our Contributions	101
5.2	AMQ Problem and Backgrounds	102
5.2.1	Problem Description	102
5.2.2	Key Components	103
5.3	Locality-Sensitive Bloom Filter	105
5.3.1	Basic LSBF Structure	106
5.3.2	False Positive and False Negative	108

5.3.3	Bit-Based Verification Scheme for Decreasing FP	109
5.3.4	Active Overflowed Scheme for Decreasing FN	110
5.4	Theoretical Analysis	111
5.4.1	False Positive Probability	112
5.4.2	False Negative Probability	114
5.4.3	Query Time and Space Overheads	116
5.5	Performance Evaluation	117
5.5.1	Implementation Details	117
5.5.2	Performance Analysis	119
5.6	Real-World Application Scenarios of LSBF	122
5.7	Related Work	124
5.8	Conclusion	125
	References	126
6	Near Real-Time Searchable Analytics for Images	129
6.1	Introduction	129
6.2	FAST Methodology	132
6.2.1	Insights and Motivations	132
6.2.2	The Methodology	133
6.3	Design and Implementations	136
6.3.1	A Use Case and Its Problem Statement	136
6.3.2	The Architecture of Use Case	138
6.3.3	Features of Images	140
6.3.4	Semantic-Aware Grouping	140
6.3.5	Semantic-Aware Namespace to Improve System Performance	142
6.4	Performance Evaluation	143
6.4.1	Experiment Setup	144
6.4.2	Results and Analysis	145
6.5	Related Work	147
6.6	Conclusion	149
	References	150
7	Data Similarity-Aware Computation Infrastructure for the Cloud	153
7.1	Introduction	153
7.2	Datasets Analysis and Problem Statement	156
7.2.1	Analysis of Real-world Traces	156
7.2.2	Problem Statement and Basic Idea	158
7.3	MERCURY Architecture	160
7.3.1	Caches in a Multicore Processor	161
7.3.2	Operating System	162

7.4	Cached Data Management in <i>MERCURY</i>	163
7.4.1	The MC-LSH Scheme	163
7.4.2	Updating Data	166
7.5	Performance Evaluation	167
7.5.1	Experiment Configuration	167
7.5.2	Results	169
7.6	System Implementation Study	169
7.7	Related Work	171
7.8	Conclusion	175
	References	175
8	Semantic-Aware Data Cube for Cloud Networks	179
8.1	Introduction	179
8.2	Backgrounds	182
8.2.1	Locality-Aware Analysis	182
8.2.2	Data Cube for Online Query	183
8.2.3	An Example	186
8.3	Design of ANTELOPE	188
8.3.1	Partial Materialization	188
8.3.2	Correlation-Based Semantic Representation	189
8.3.3	Identification of Locality-Aware Data	191
8.4	System Implementations	192
8.4.1	Structure Mapping	192
8.4.2	Grouping Procedures	193
8.4.3	Insertion	194
8.4.4	Deletion	195
8.4.5	Aggregate Queries	195
8.4.6	Incremental Updates	196
8.5	Performance Evaluation	196
8.5.1	Experimental Setup	197
8.5.2	Experimental Results	198
8.6	Related Work	198
8.7	Conclusion	201
	References	201