

Fast and Scalable Cloud Data Management

Felix Gessert • Wolfram Wingerath • Norbert Ritter

Fast and Scalable Cloud Data Management



Felix Gessert
Baqend GmbH
Hamburg, Germany

Wolfram Wingerath
Baqend GmbH
Hamburg, Germany

Norbert Ritter
Department of Informatics
University of Hamburg
Hamburg, Germany

ISBN 978-3-030-43505-9 ISBN 978-3-030-43506-6 (eBook)
<https://doi.org/10.1007/978-3-030-43506-6>

© Springer Nature Switzerland AG 2020

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, expressed 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 Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Our research for this book goes back a long way. It all started as early as 2010 with a bachelor's thesis on how to make use of the purely expiration-based caching mechanisms of the web in a database application with rigorous consistency requirements. Strong encouragement by fellow researchers and growing developer interest eventually made us realize that the task of providing low latency for users in globally distributed applications does not only pose an interesting research challenge but also an actual real-world problem that was still mostly unsolved.

This revelation eventually led to the creation of Baqend, a Backend-as-a-Service platform designed for developing fast web applications. We built Baqend on knowledge gathered in many bachelor's and master's and a number of PhD theses. Technically, Baqend is rooted in our research systems, Orestes for web caching with tunable consistency, its extension Quaestor for query result caching, and InvalidDB for scalable push-based real-time queries for end-users. We are telling you all this for a reason: Because given its origin, this book does not only condense our knowledge after years of research done in a practical context but it also encapsulates our view on the concepts and systems that are currently out there. While we try to provide a balanced overview of the current state of affairs in data management and web technology, we are clearly opinionated with regard to certain best practices and architectural patterns. We would like to consider this a positive trait of this book—and we hope you agree;-)

Hamburg, Germany
Hamburg, Germany
Hamburg, Germany
December 2019

Felix Gessert
Wolfram Wingerath
Norbert Ritter

Contents

- 1 Introduction** 1
 - 1.1 Modern Data Management and the Web 2
 - 1.2 Latency vs. Throughput..... 3
 - 1.3 Challenges in Modern Data Management 3
 - 1.4 Outline of the Book 6
 - References 8
- 2 Latency in Cloud-Based Applications** 13
 - 2.1 Three-Tier Architectures..... 14
 - 2.1.1 Request Flow 15
 - 2.1.2 Implementation 15
 - 2.1.3 Problems of Server-Rendered Architectures 16
 - 2.2 Two-Tier Architectures 17
 - 2.2.1 Request Flow 19
 - 2.2.2 Implementation 19
 - 2.2.3 Problems of Client-Rendered Architectures 21
 - 2.3 Latency and Round-Trip Time 21
 - 2.4 Cloud Computing as a Source of Latency 23
 - 2.4.1 Characteristics 23
 - 2.4.2 Service Models..... 24
 - 2.4.3 Deployment Models 25
 - 2.4.4 Latency in Cloud Architectures 26
 - References 27
- 3 HTTP for Globally Distributed Applications** 33
 - 3.1 HTTP and the REST Architectural Style 33
 - 3.2 Latency on the Web: TCP, TLS and Network Optimizations 35
 - 3.3 Web Caching for Scalability and Low Latency..... 38
 - 3.3.1 Types of Web Caches 39
 - 3.3.2 Scalability of Web Caching 40
 - 3.3.3 Expiration-Based Web Caching 41

3.3.4	Invalidation-Based Web Caching	43
3.3.5	Challenges of Web Caching for Data Management	44
3.4	The Client Perspective: Processing, Rendering, and Caching for Mobile and Web Applications	45
3.4.1	Client-Side Rendering and Processing	46
3.4.2	Client-Side Caching and Storage	48
3.5	Challenges and Opportunities: Using Web Caching for Cloud Data Management	50
	References	50
4	Systems for Scalable Data Management	57
4.1	NoSQL Database Systems	57
4.2	Data Models: Key-Value, Wide-Column and Document Stores	59
4.2.1	Key-Value Stores	59
4.2.2	Document Stores	60
4.2.3	Wide-Column Stores	60
4.3	Pivotal Trade-Offs: Latency vs. Consistency vs. Availability	61
4.3.1	CAP	62
4.3.2	PACELC	63
4.4	Relaxed Consistency Models	63
4.4.1	Strong Consistency Models	65
4.4.2	Staleness-Based Consistency Models	66
4.4.3	Session-Based Consistency Models	68
4.5	Offloading Complexity to the Cloud: Database- and Backend-as-a-Service	70
4.5.1	Database-as-a-Service	70
4.5.2	Backend-as-a-Service	72
4.5.3	Multi-Tenancy	73
4.6	Summary	75
	References	75
5	Caching in Research and Industry	85
5.1	Reducing Latency: Replication, Caching and Edge Computing	87
5.1.1	Eager and Lazy Geo-Replication	87
5.1.2	Caching	88
5.1.3	Edge Computing	88
5.1.4	Challenges	89
5.2	Server-Side, Client-Side and Web Caching	89
5.2.1	Server-Side Caching	89
5.2.2	Client-Side Database Caching	90
5.2.3	Caching in Object-Relational and Object-Document Mappers	92
5.2.4	Web Caching	93
5.3	Cache Coherence: Expiration-Based vs. Invalidation-Based Caching	94
5.3.1	Expiration-Based Caching	94

5.3.2	Leases	94
5.3.3	Piggybacking	95
5.3.4	Time-to-Live (TTL)	96
5.3.5	Invalidation-Based Caching	96
5.3.6	Browser Caching	99
5.3.7	Web Performance	100
5.4	Query Caching	100
5.4.1	Peer-to-Peer Query Caching	101
5.4.2	Mediators	101
5.4.3	Query Caching Proxies and Middlewares	101
5.4.4	Search Result Caching	102
5.4.5	Summary Data Structures for Caching	103
5.5	Eager vs. Lazy Geo Replication	104
5.5.1	Replication and Caching	104
5.5.2	Eager Geo-Replication	105
5.5.3	Lazy Geo-Replication	108
5.6	Summary	113
	References	114
6	Transactional Semantics for Globally Distributed Applications	131
6.1	Latency vs. Distributed Transaction Processing	131
6.1.1	Distributed Transaction Architectures	132
6.2	Entity Group Transactions	137
6.3	Multi-Shard Transactions	138
6.4	Client-Coordinated Transactions	139
6.5	Middleware-Coordinated Transactions	140
6.6	Deterministic Transactions	141
6.7	Summary: Consistency vs. Latency in Distributed Applications	142
	References	143
7	Polyglot Persistence in Data Management	149
7.1	Functional and Non-functional Requirements	150
7.1.1	Implementation of Polyglot Persistence	152
7.2	Multi-Tenancy and Virtualization in Cloud-Based Deployments	154
7.2.1	Database Privacy and Encryption	156
7.2.2	Service Level Agreements	157
7.3	Auto-Scaling and Elasticity	158
7.4	Database Benchmarking	159
7.4.1	Consistency Benchmarking	159
7.5	REST APIs, Multi-Model Databases and Backend-as-a-Service	160
7.5.1	Backend-as-a-Service	161
7.5.2	Polyglot Persistence	163
7.6	Summary	164
	References	164

8	The NoSQL Toolbox: The NoSQL Landscape in a Nutshell	175
8.1	Sharding	176
8.1.1	Range Partitioning	176
8.1.2	Hash Partitioning	177
8.1.3	Entity-Group Sharding	177
8.2	Replication	178
8.3	Storage Management.....	179
8.4	Query Processing.....	182
8.5	Summary: System Studies	183
	References	187
9	Summary and Future Trends	191
9.1	From Abstract Requirements to Concrete Systems.....	192
9.2	Future Prospects.....	193

About the Authors

Felix Gessert is the CEO and co-founder of the Backend-as-a-Service company Baqend.¹ During his PhD studies at the University of Hamburg, he developed the core technology behind Baqend’s web performance service. He is passionate about making the web faster by turning research results into real-world applications. He frequently talks at conferences about exciting technology trends in data management and web performance. As a Junior Fellow of the German Informatics Society (GI), he is working on new ideas to facilitate the research transfer of academic computer science innovation into practice.

Wolfram “Wolle” Wingerath is the leading data engineer at Baqend¹ where he is responsible for data analytics and all things related to real-time query processing. During his PhD studies at the University of Hamburg, he conceived the scalable design behind Baqend’s real-time query engine and thereby also developed a strong background in real-time databases and related technology such as scalable stream processing, NoSQL database systems, cloud computing, and Big Data analytics. Eager to connect with others and share his experiences, he regularly speaks at developer and research conferences.

Norbert Ritter is a full professor of computer science at the University of Hamburg, where he heads the Databases and Information Systems (DBIS) group. He received his PhD from the University of Kaiserslautern in 1997. His research interests include distributed and federated database systems, transaction processing, caching, cloud data management, information integration, and autonomous database systems. He has been teaching NoSQL topics in various courses for several years. Seeing the many open challenges for NoSQL systems, he, Wolle, and Felix have been organizing the annual Scalable Cloud Data Management Workshop² to promote research in this area.

¹Baqend: <https://www.baqend.com/>.

²Scalable Cloud Data Management Workshop: <https://scdm.cloud>.