NoSQL & Real-Time Data Management in Research & Practice

Wolfram Wingerath[1] Felix Gessert[2] Norbert Ritter[3]

Abstract: Users have come to expect reactivity from mobile and web applications, i.e. they assume that changes made by other users become visible immediately. However, developers are challenged with building reactive applications on top of traditional pull-oriented databases, because they are ill-equipped to push new information to the client. Systems for data stream management and processing, on the other hand, are natively push-oriented and thus facilitate reactive behavior, but they do not follow the same collection-based semantics as traditional databases: Instead of database collections, stream-oriented systems are based on a notion of potentially unbounded sequences of data items. In this tutorial, we survey and categorize the system space between pull-oriented databases and push-oriented stream management systems, using their respectively facilitated means of data retrieval as a reference point. We start with an in-depth survey of the most relevant NoSQL databases to provide a comparative classification and highlight open challenges. To this end, we analyze the approach of each system to derive its scalability, availability, consistency, data modeling, and querying characteristics. We present how each system's design is governed by a central set of trade-offs over irreconcilable system properties. We then cover recent research results in distributed data management to illustrate that some shortcomings of NoSQL systems could already be solved in practice, whereas other NoSQL data management problems pose interesting and unsolved research challenges. A particular emphasis lies on the novel system class of real-time databases which combine the push-based access paradigm of stream-oriented systems with the collection-based query semantics of traditional databases. We explore why real-time databases deserve distinction in a separate system class and dissect their different architectures to highlight issues, derive open challenges, and discuss avenues for addressing them.

Keywords: Real-Time Databases, NoSQL, Scalability, Distributed Systems, High Availability, Polyglot Persistence, Stream Processing, Cloud Data Management, Big Data, Push-Based Data Access

1 Introduction

The design of any data management system reflects a bias towards either pull-based or push-based data access: A *pull-based* query assembles data from a bounded data repository and completes by returning data once, whereas a *push-based* query processes a conceptually unbounded stream of information to generate incremental output over time. For example, traditional databases are clearly geared towards efficiency for pull-based data retrieval, even though they do support push-based access to a certain degree (e.g. through triggers). Figure 1 illustrates how the different classes of data management systems can be classified by the way they facilitate access to data.

At the one extreme, there are **traditional databases** which represent snapshots of domain knowledge that are the basis of all queries. At the other extreme, there are general-purpose

[1] Baqend, Stresemannstraße 23, 22769 Hamburg, ww@baqend.com
[2] Baqend, Stresemannstraße 23, 22769 Hamburg, fg@baqend.com
[3] Universität Hamburg, DBIS, Vogt-Kölln-Straße 30, 22527 Hamburg, ritter@informatik.uni-hamburg.de

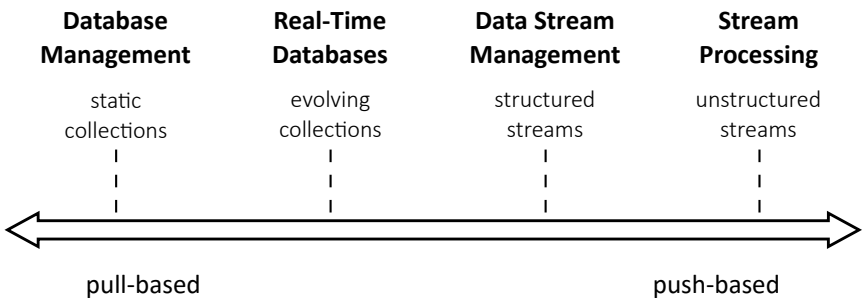

Fig. 1: Different classes of data management systems and the access patterns they support.

stream processing engines which are designed to generate output from conceptually unbounded and arbitrarily structured ephemeral data streams. Real-time databases and data stream management systems both stand in the middle, but adhere to different semantics: **Real-time databases** work on evolving collections that are distinguished from their static counterparts (i.e. from typical database collections) through continuous integration of updates over time, enabling push-based real-time queries. **Data stream management** systems provide APIs to query data streams, for example, by applying filters to incoming data or by computing rolling aggregations and joins over configurable time windows.

2 Tutorial Outline

With this tutorial, we intend to provide an overview over the entire system spectrum. Our tutorial is divided into three parts as follows.

In the first part, we cover **pull-based** data management systems. To this end, we first recall the basics of distributed data management (e.g. partitioning, replication, eventual consistency, NoSQL data models) and present the most important impossibility results (e.g. the CAP theorem). We then provide an in-depth survey of the NoSQL landscape by discussing the individual architectures and classifying each system according to functional and non-functional properties.

In the second part, we turn to **push-based** systems for real-time data management. After a short historical recap of push-based mechanisms in data management, we dissect the current state of the art in real-time databases and stream processing frameworks with respect to their capabilities in storing, querying, and analyzing data with low latency.

In the final part of our tutorial, we review polyglot persistence environments that bring together many of the discussed systems, unfold open practical and research challenges in the field of data management, and discuss possible venues for addressing them.

3 Intended Audience & Relationship to Prior Tutorials

We expect the tutorial to appeal to a large portion of the BTW community, specifically anybody interested in a comparative overview of the current data management landscape and a discussion of open challenges. Our target audience thus includes students who are

looking for novel research topics and orientation, experienced researchers in the fields of database systems, cloud computing, and distributed systems, as well as industry practitioners tackling data management problems who are looking for a survey and classification of existing systems and their respective sweet spots.

This tutorial includes revised content from earlier tutorials given at EDBT 2018 [WGW+18], BTW 2017 [GWR17], and ICDE 2016 [GR16].

4 Presenters

Wolfram Wingerath is a distributed systems engineer at the Backend-as-a-Service company Baqend[4] where he is responsible for all things related to real-time query processing. During his PhD studies at the University of Hamburg, Wolfram 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, Wolfram regularly speaks at developer and research conferences.

Felix Gessert is the CEO and co-founder of Baqend[4]. During his PhD studies at the University of Hamburg, he developed the core technology behind Baqend's web performance service. Felix is passionate about making the web faster by bringing research to practice. He frequently talks at conferences about exciting technology trends in data management and web performance.

Norbert Ritter is a full professor of computer science at the University of Hamburg, where he heads the databases and information systems 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 and Felix Gessert have been organizing the annual Scalable Cloud Data Management Workshop[5] to promote research in this area.

References

[GR16] Felix Gessert and Norbert Ritter. Scalable Data Management: NoSQL Data Stores in Research and Practice. In *32nd IEEE International Conference on Data Engineering, ICDE 2016*, 2016.

[GWR17] Felix Gessert, Wolfram Wingerath, and Norbert Ritter. Scalable Data Management: An In-Depth Tutorial on NoSQL Data Stores. In *Datenbanksysteme für Business, Technologie und Web (BTW 2017) - Workshopband, 2.-3. März 2017, Stuttgart, Germany*, volume P-266 of *LNI*, pages 399–402. GI, 2017.

[WGW+18] Wolfram Wingerath, Felix Gessert, Erik Witt, Steffen Friedrich, and Norbert Ritter. Real-Time Data Management for Big Data. In *Proceedings of the 21th International Conference on Extending Database Technology, EDBT 2018, Vienna, Austria, March 26-29, 2018*. OpenProceedings.org, 2018.

[4] Baqend: `https://www.baqend.com/`.

[5] Annual Scalable Cloud Data Management Workshop: `www.scdm.cloud`.