

Columnar-databases for Big Data

Apache HBase

Humberto Silva Galiza de Freitas - D20124066

TU060 - M.Sc in Computer Science / Advanced Software Development

Prof. Brendan Tierney

Agenda

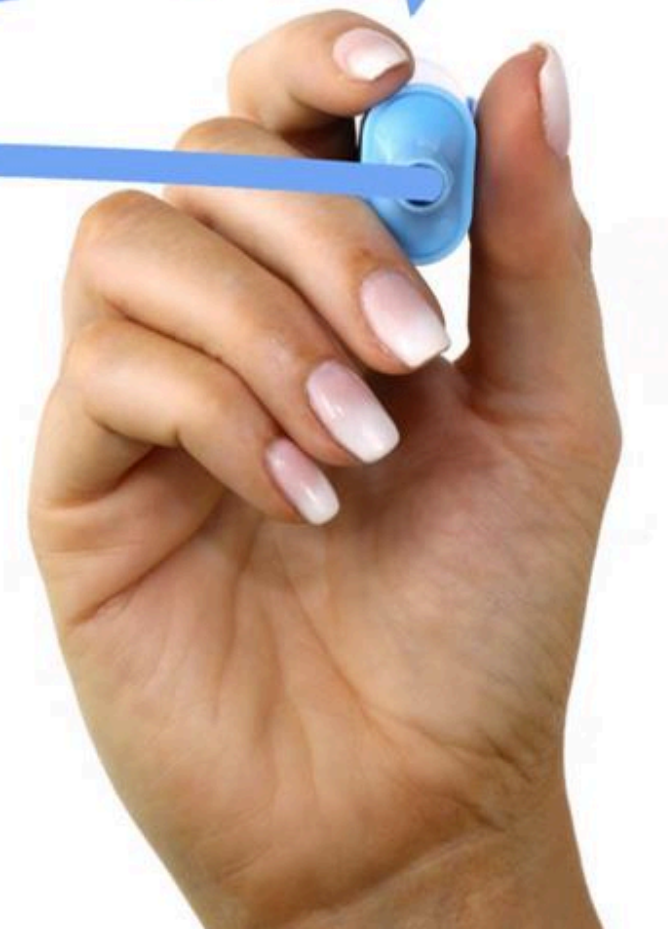
- Background
- Columnar-databases: Solutions Landscape
- HBase
 - Introduction
 - HBase vs RDBMS
 - HBase Data Model & Architecture
 - HBase Storage Model & Architecture
 - HBase Client API
 - HBase: quick demo
- Final Remarks

Background

What is Big Data and how columnar-databases help?

- Data explosion problem
 - Volume, Velocity, Variety:
 - A constant flow of massive volume of heterogeneous data being generated every second.
 - Data is **Unstructured & unpredictable**
 - RDBMs expects a well-defined / fixed structured (normalised) data
 - Optimised for Online Transaction Processing (OLTP) (e.g., single transactions, like, inserts, updates, etc)

BIG DATA



Background [2]

What is Big Data and how columnar-databases help?

- Columnar-databases solves the problem by storing data in 'columns' instead of 'rows'
 - Variable columns: columns that are not accessed in a query can be skipped
 - Allow for partial reads: much more efficient because a lower volume of data is loaded due to reading only the relevant data instead of the whole record
 - Because they store 'similar' columns together, the approach allows for better data compression for certain data types
- Use cases
 - Suitable for computing trends and averages for trillions of rows and petabytes for data (e.g., OLAP)
 - When the requirement is that a large data set needs to be distributed across multiple database nodes, e.g.,
 - Time series data, IoT sensor data, log data, etc

Logical table

	col1	col2	col3
row1	1	2	3
row2	4	5	6
row3	7	8	9
row4	10	11	12

Row-oriented layout (SequenceFile)

row1	row2	row3	row4
1 2 3	4 5 6	7 8 9	10 11 12

Column-oriented layout (RCFile)

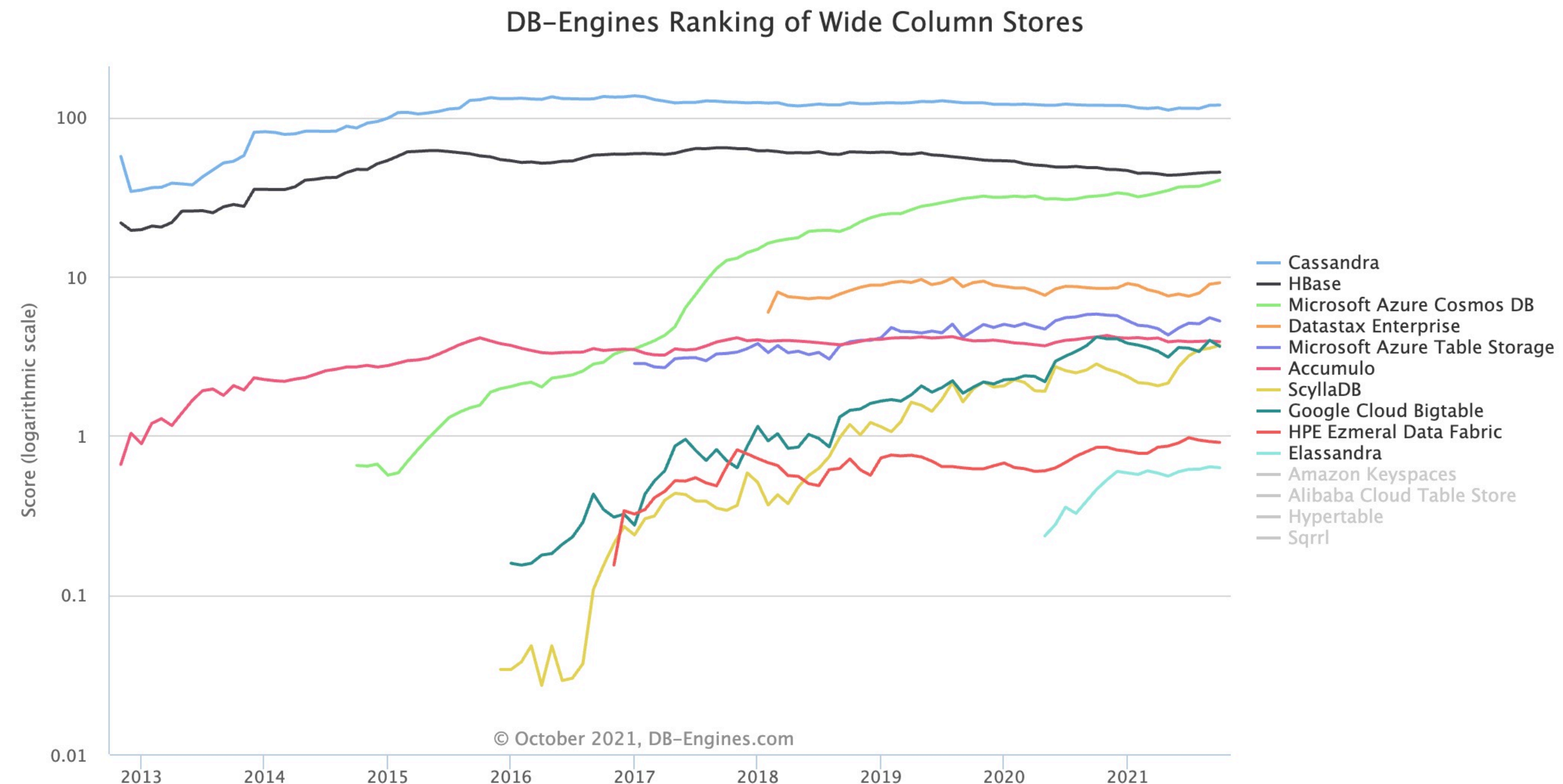
row split 1			row split 2		
col1	col2	col3	col1	col2	col3
1 4	2 5	3 6	7 10	8 11	9 12

PERSON TABLE					
row key	personal_data		demographic		...
PersonID	Name	Address	BirthDate	Gender	...
1	H. Houdini	Budapest, Hungary	1926-10-31	M	
2	D. Copper	New Jersey, USA	1956-09-16	M	
3	Merlin	Stonehenge, England	1136-12-03	F	
...	
500,000,000	F. Cadillac	Nevada, USA	1964-01-07	M	

Columnar-databases

Solutions landscape

- There are a multitude of solutions in the market. Top 2 (in interest over time) per category:
 - Free and Open-Source Software (FOSS)
 - Apache Cassandra & Apache HBase
 - Platform-as-a-Service (PaaS)
 - Google BigTable & Amazon RedShift
 - Proprietary
 - DataStax Enterprise (Cassandra) & Cloudera Enterprise (HBase)

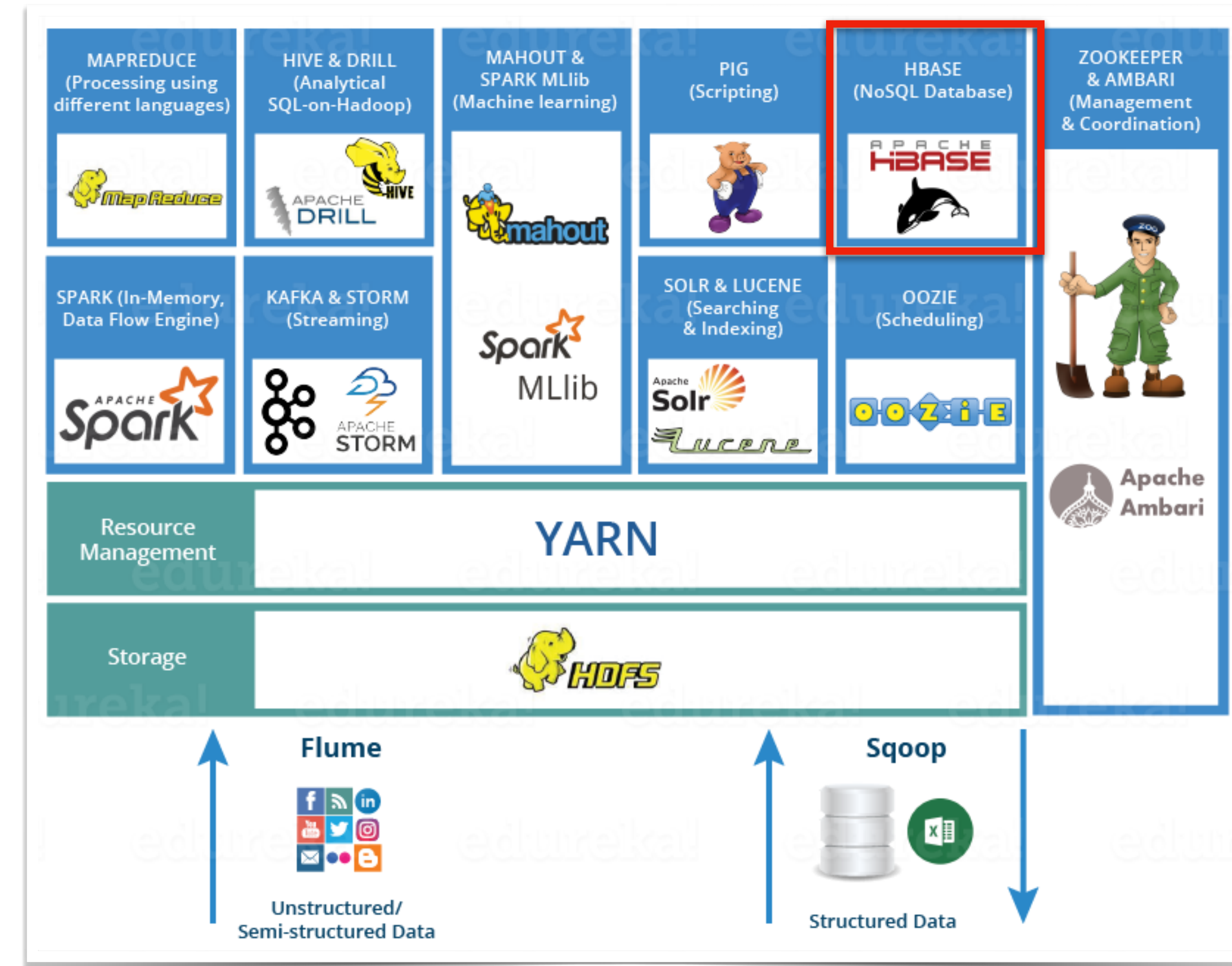


Source: https://db-engines.com/en/ranking_trend/wide+column+store

HBase

A distributed, consistent, and scalable columnar-database

- HBase is modeled on the ideas of Google's BigTable original paper
- HBase is programmed in Java and is part of Apache's Hadoop MapReduce framework
 - HBase replace Google File System with the robust Hadoop Distributed File System (HDFS)
 - **Unlike HDFS, HBase allows random access. It stores the data in indexed HDFS files for faster lookups by row keys. That's the gap HBase fills.**

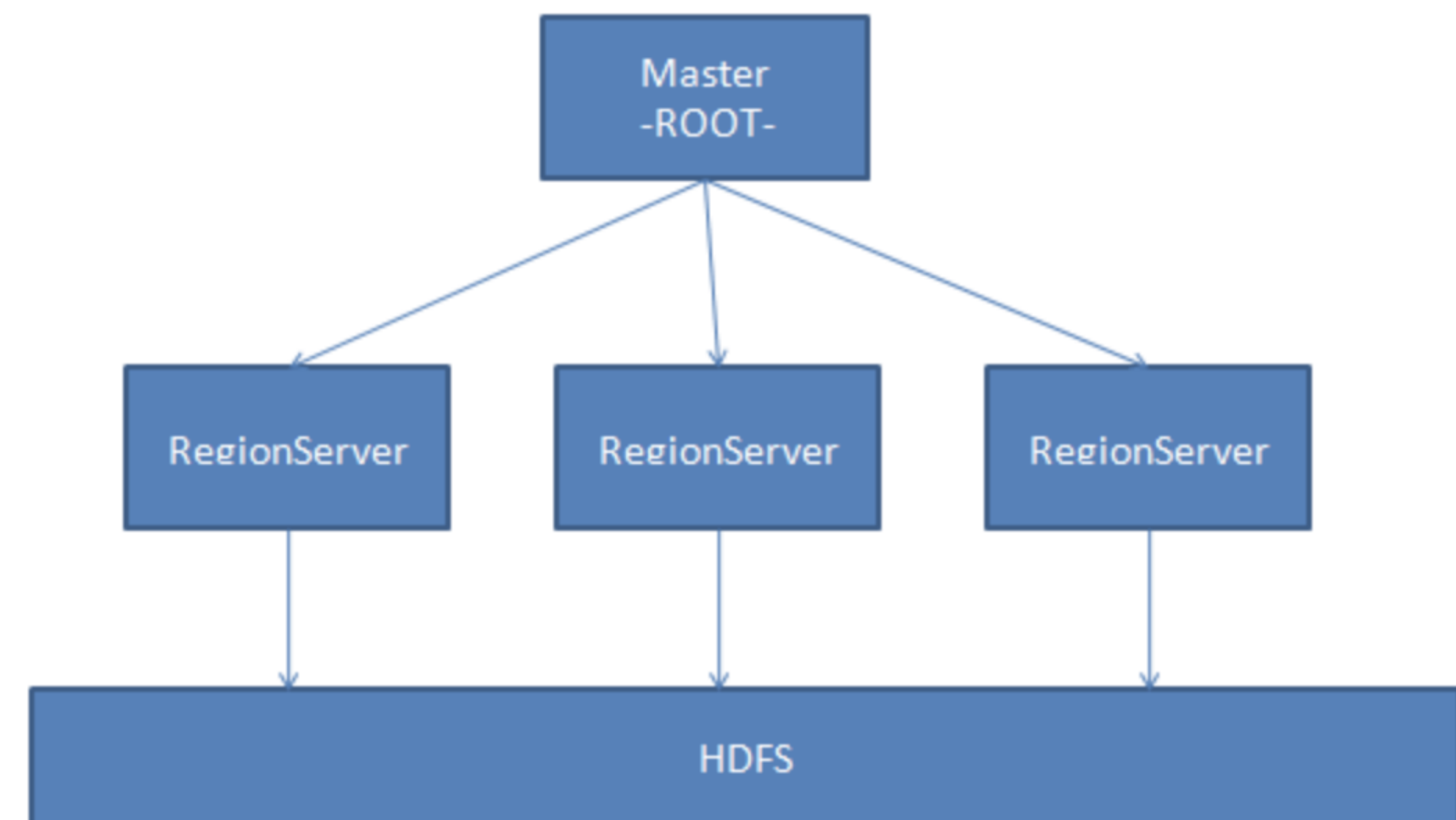
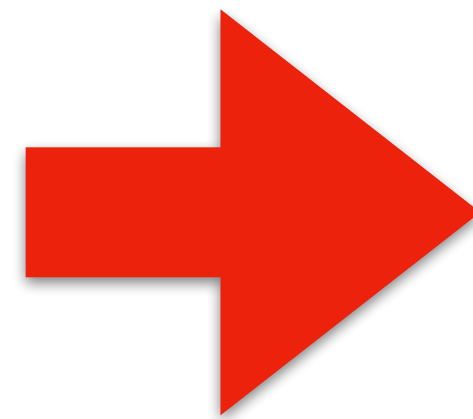
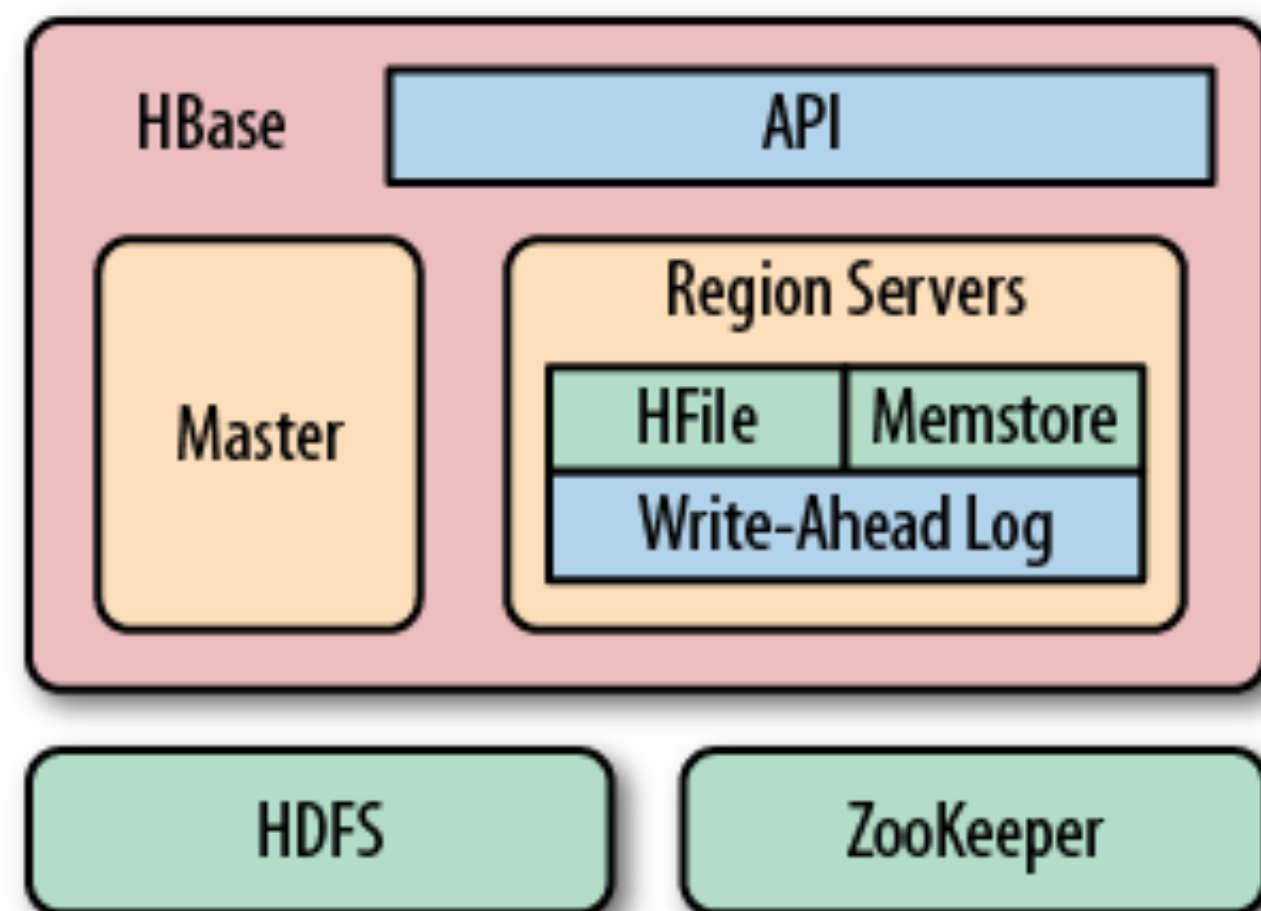


Credits: <https://www.edureka.co/blog/hadoop-tutorial/>

HBase

A distributed, consistent, and scalable columnar-database [2]

- Three major components: the client library, one master server, and many region servers.
- The master is responsible for assigning regions to region servers
- HBase uses Apache ZooKeeper, a reliable, highly available, persistent, and distributed coordination service, to facilitate this task



Credits (both): HBase: the definitive guide: random access to your planet-size data. "O'Reilly Media, Inc.", 2011

HBase vs RDBMS

A comparison of functionalities

- Although HBase stores data on disk in a column-oriented format, it is distinctly different from traditional columnar databases.

HBase	RDBMS
HBase is schema-less, it doesn't have the concept of fixed columns schema; defines only column families.	An RDBMS is governed by its schema, which describes the whole structure of tables.
It is built for wide tables. HBase is horizontally scalable.	It is thin and built for small tables. Hard to scale.
No transactions are there in HBase.	RDBMS is transactional.
It has de-normalized data.	It will have normalized data.
It is good for semi-structured as well as structured data.	It is good for structured data.

HBase: data model & architecture

Overview

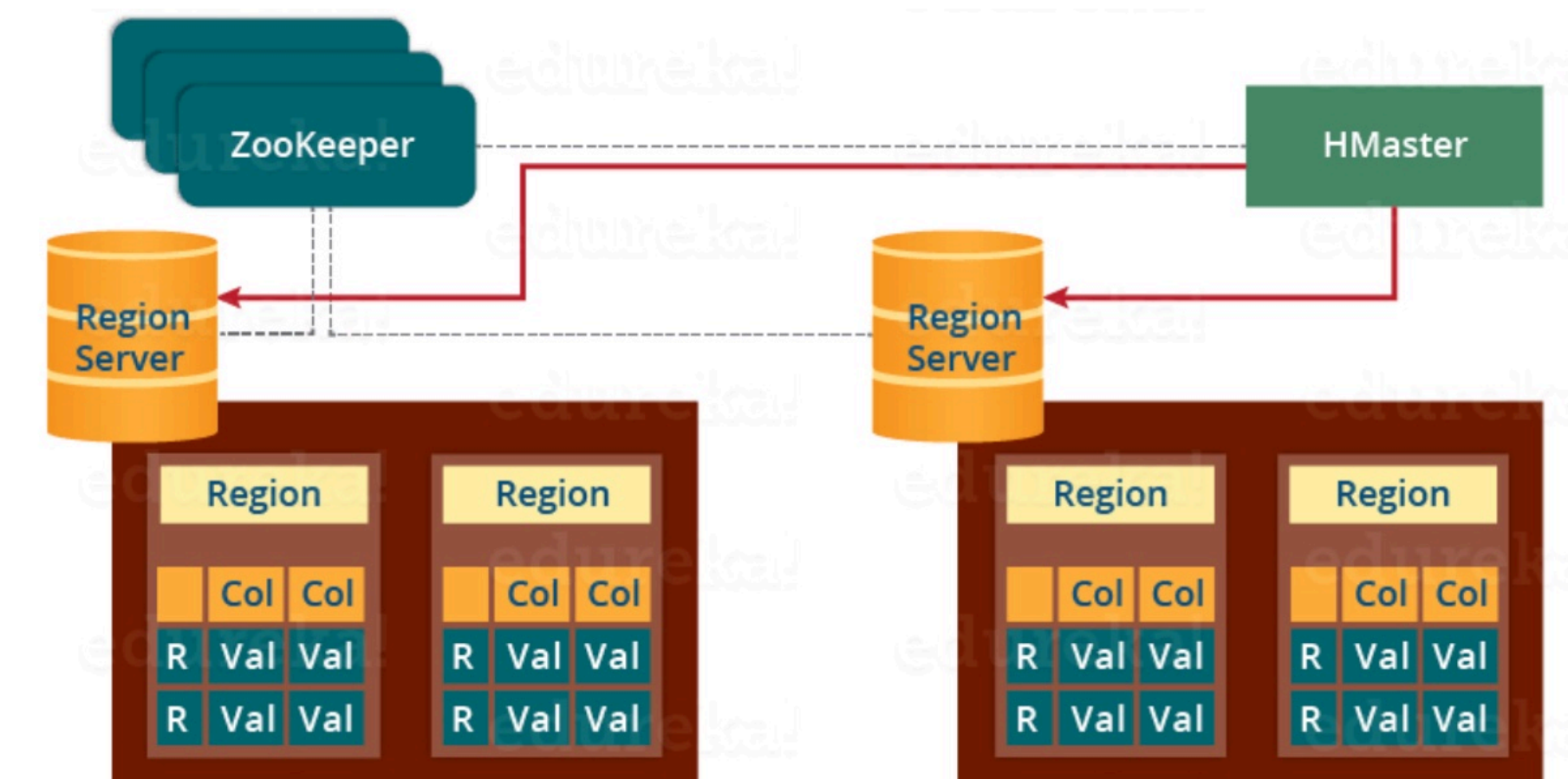
- HBase data model stores semi-structured data having different data types, varying column size, and field size.
- The layout of the HBase data model eases data partitioning and distribution across the cluster.
- A Region (HRegion) is the basic unit of horizontal scalability in the HBase ecosystem.
- HRegions are essentially contiguous ranges of rows stored together.
- A HRegion contains all the rows between the start key and the end key assigned to that region

Row Key		Column Family		
Row Key		Customers		Products
Customer ID	Customer Name	City & Country	Product Name	Price
1	Sam Smith	California, US	Mike	\$500
2	Arijit Singh	Goa, India	Speakers	\$1000
3	Ellie Goulding	London, UK	Headphones	\$800
4	Wiz Khalifa	North Dakota, US	Guitar	\$2500

Column Qualifiers

Cell

Credits: <https://www.edureka.co/blog/hadoop-tutorial/>

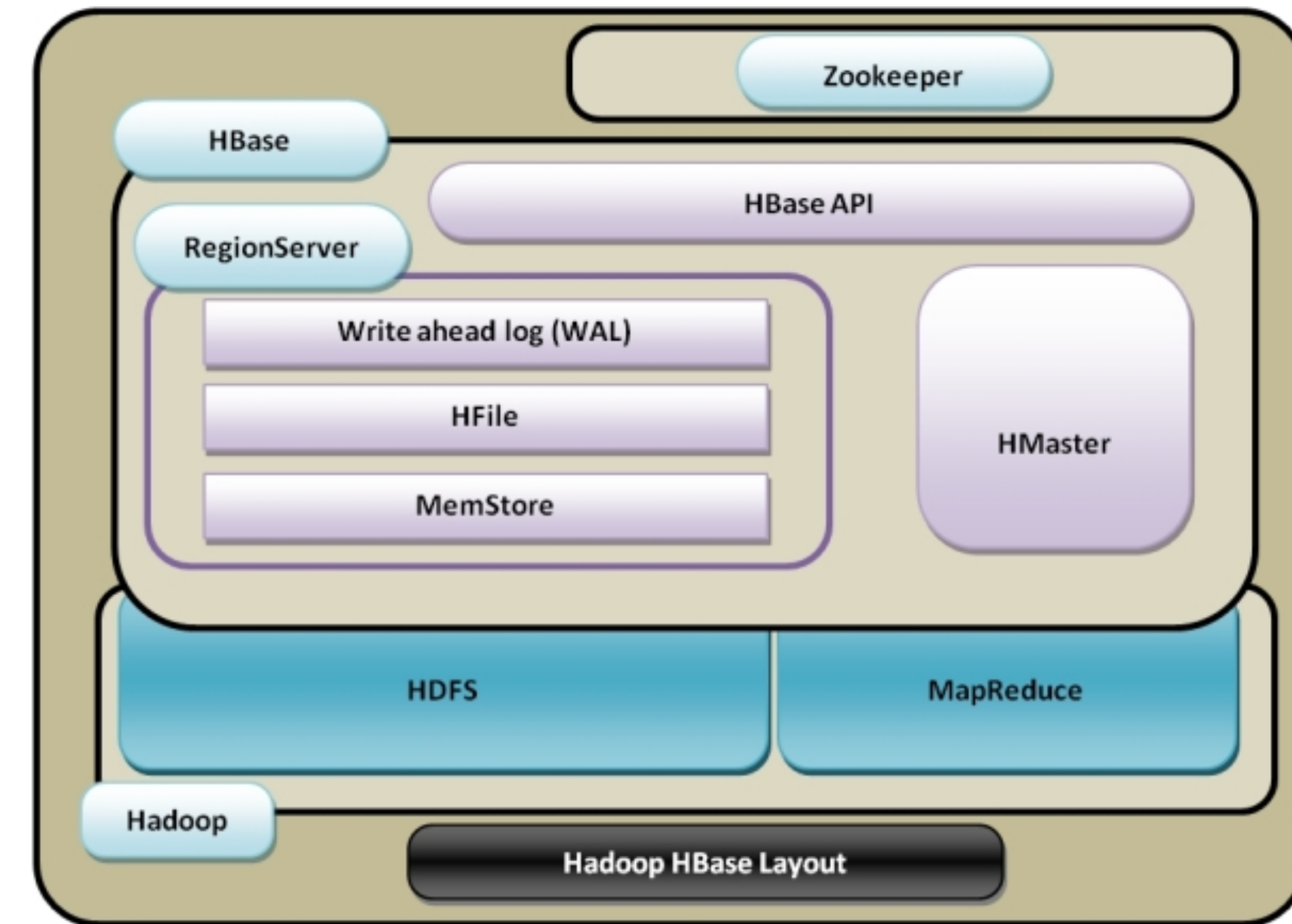


Credits: <https://www.edureka.co/blog/hadoop-tutorial/>

HBase: storage model & architecture

Overview

- HBase HMaster performs DDL operations (create and delete tables)
- All nodes in the cluster are coordinated by Zookeeper and handle various issues encountered during HBase operation
 - It also handles load balancing of the regions across region servers.
- Each Region Server maintains at minimum:
 - Block Cache, MemStore, Write-Ahead Log (WAL), and HFile.

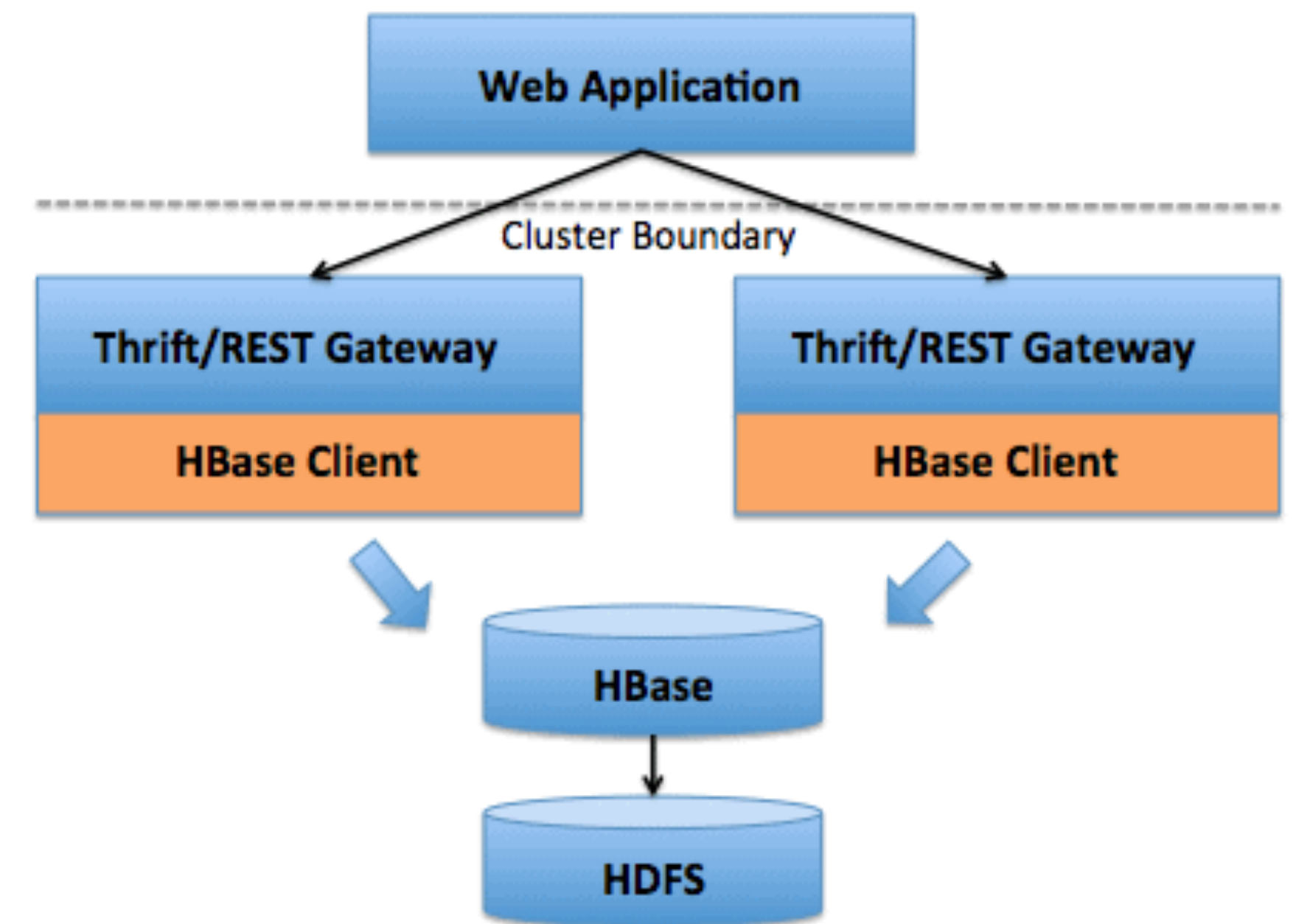


Credits: S. Shripav, *Learning HBase*. Packt Publishing Ltd, 2014.

HBase: client API

Overview

- HBase offers programmatic access through a Java API (native).
- There are also two other options to access HBase without Java:
 - Hbase Thrift interface
 - Lightweight and faster of the two options
 - REST interface (formally Stargate)
 - A RESTful web service front end for HBase, which uses HTTP verbs to perform an action



Credits: <https://blog.cloudera.com/how-to-use-the-hbase-thrift-interface-part-1/>

HBase: hands on

Quick demo

- Source code & docker image for testing: https://github.com/humbertogaliza/hbase_tutorial

127.0.0.1:55018/master-status

APACHEHBASE

Home

Table Details

Procedures & Locks

HBCK Report

Process Metrics

Local Logs

Log Level

Debug Dump

Metrics Dump

Procedures

Tables

User Tables

System Tables

Snapshots

3 table(s) in set. [Details]

Namespace	Name	State	Regions								Description
			OPEN	OPENING	CLOSED	CLOSING	OFFLINE	FAILED	SPLIT	Other	
default	table-name	ENABLED	1	0	0	0	0	0	0	0	'table-name', {NAME => 'family', VERSIONS => '3', BLO
default	table2-tudublin	ENABLED	1	0	0	0	0	0	0	0	'table2-tudublin', {NAME => 'family', VERSIONS => '3', I 'false'}
default	wiki	ENABLED	1	0	0	0	0	0	0	0	'wiki', {NAME => 'text'}

Final remarks

- This work highlighted the side effects of the data explosion problem and how the approaches to database systems had to evolve to cope with these challenges' scale and shift in data patterns.
- Columnar databases are a robust and so far well-tested candidate to deal with massive volumes of data, providing fast writing and retrieval while supporting heterogeneous data types.
- Furthermore, Hadoop deployment is constantly rising, both for big techies and enterprises, and HBase is the perfect platform for working on top of the powerful and robust HDFS to meet all the big data storage and processing requirements. Whenever the enterprise use case requires random, real-time read/write access to Big Data, Apache HBase is a good fit solution.

References

- [1] Merriam-Webster, Big data, in Merriam-Webster.com Dictionary. [Online]. Available: <https://www.merriam-webster.com/dictionary/big%5C%20data> (visited on 09/29/2021), (accessed: 29.09.2021).
- [2] J. Hurwitz, A. Nugent, F. Halper, and M. Kaufman, "Big data," New York, 2013. [Online]. Available: <http://www.dummies.com/how-to/content/big-data-for-dummies-cheat-sheet.html> (visited on 10/01/2021), (accessed: 23.10.2021).
- [3] Sisense, Oltp, in Sisense glossary. [Online]. Available: <https://www.sisense.com/glossary/oltp/> (visited on 09/29/2021), (accessed: 29.09.2021).
- [4] M. Stonebraker and U. Çetintemel, "'one size fits all' an idea whose time has come and gone," in Making Databases Work: the Pragmatic Wisdom of Michael Stonebraker, 2018, pp. 441–462.
- [5] A. Griffith. "Deciding between row- and columnar-stores | why we chose both." (), [Online]. Available: <https://medium.com/bluecore-engineering/deciding-between-row-and-columnar-stores-why-we-chose-both-3a675dab4087> (visited on 08/10/2020). (accessed: 18.10.2021).
- [6] T. White, Hadoop: The definitive guide. "O'Reilly Media, Inc.", 2012.
- [7] L. George, HBase: the definitive guide: random access to your planet-size data. "O'Reilly Media, Inc.", 2011.
- [8] Timepasstechies. "Timepasstechies blog - big data tutorial." (), [Online]. Available: <https://timepasstechies.com/row-oriented-column-oriented-file-formats-hadoop/> (visited on 10/23/2021). (accessed: 23.10.2021).
- [9] D. Ocean. "Understanding database sharding." (), [Online]. Available: <https://www.digitalocean.com/community/tutorials/understanding-database-sharding> (visited on 10/23/2021). (accessed: 23.10.2021).
- [10] A. Petrov, Database internals: A deep dive into how distributed data systems work, 2019.
- [11] Scylla, Wide-column database, in ScyllaDB glossary. [Online]. Available: <https://www.scylladb.com/glossary/>

References [2]

- [12] Rhodes. “Hbase - cs 305 syllabus.” (), [Online]. Available: <http://jcsites.juniata.edu/faculty/rhodes/smui/hbase.htm> (visited on 10/23/2021). (accessed: 23.10.2021).
- [13] DB-Engines. “Db-engines ranking - trend of wide column stores popularity.” (), [Online]. Available: https://db-engines.com/en/ranking_trend/wide+column+store (visited on 10/23/2021). (accessed: 23.10.2021).
- [14] F. Chang, J. Dean, S. Ghemawat, et al., “Bigtable: A distributed storage system for structured data,” ACM Transactions on Computer Systems (TOCS), vol. 26, no. 2, pp. 1–26, 2008.
- [15] L. Perkins, E. Redmond, and J. Wilson, Seven databases in seven weeks: a guide to modern databases and the NoSQL movement. Pragmatic Bookshelf, 2018.
- [16] Edureka. “Hadoop tutorial.” (), [Online]. Available: <https://www.edureka.co/blog/hadoop-tutorial/> (visited on 10/23/2021). (accessed: 23.10.2021).
- [17] M. Chen, S. Mao, Y. Zhang, V. C. Leung, et al., “Big data: Related technologies, challenges and future prospects,” 2014.
- [18] Edureka. “Hbase tutorial: Hadoop database.” (), [Online]. Available: https://www.tutorialspoint.com/hbase/hbase_overview.htm (visited on 10/04/2021). (accessed: 04.10.2021).
- [19] DataScienceCentral.com. “Hdfs vs. hbase : All you need to know - data science central.” (), [Online]. Available: <https://www.datasciencecentral.com/xn/detail/6448529:BlogPost:610315> (visited on 09/29/2021). (accessed: 29.09.2021).
- [20] A. HBase. “Apache hbase reference guide.” (2021), [Online]. Available: <https://hbase.apache.org/book.html> (visited on 11/08/2021). (accessed: 08.11.2021).
- [21] N. Garg, HBase Essentials. Packt Publishing Ltd, 2014.
- [22] OpenTSDB. “Opentsdb 2.4 documentation - user guide - storage - hbase schema.” (2021), [Online]. Available: http://opentsdb.net/docs/build/html/user_guide/backends/hbase.html (visited on 11/09/2021). (accessed: 09.11.2021).
- [23] S. Shriparv, Learning HBase. Packt Publishing Ltd, 2014.

References [3]

- [24] towardsdatascience.com. “Hbase working principle: A part hadoop architecture.” (), [Online]. Available: <https://towardsdatascience.com/hbase-working-principle-a-part-of-hadoop-architecture-fbe0453a031b> (visited on 09/29/2021). (accessed: 29.09.2021).
- [25] C. Blog. “How-to: Use the hbase thrift interface, part 1.” (), [Online]. Available: <https://blog.cloudera.com/how-to-use-the-hbase-thrift-interface-part-1/> (visited on 10/07/2021). (accessed: 07.10.2021).
- [26] “Exploring database trends using python pytrends (google trends).” (), [Online]. Available: <https://oralytics.com/2020/11/09/exploring-database-trends-using-python-pytrends-google-trends/>. (accessed: 20.10.2021).