

## 0.1 CouchDB

CouchDB[1] is a database that completely embraces the web. Store your data with JSON documents. Access your documents with your web browser, via HTTP. Query, combine, and transform your documents with JavaScript. CouchDB works well with modern web and mobile apps. You can distribute your data, efficiently using CouchDB's incremental replication. CouchDB supports master-master setups with automatic conflict detection. CouchDB comes with a suite of features, such as on-the-fly document transformation and real-time change notifications, that make web development a breeze. It even comes with an easy to use web administration console, served directly out of CouchDB! We care a lot about distributed scaling. CouchDB is highly available and partition tolerant, but is also eventually consistent. And we care a lot about your data. CouchDB has a fault-tolerant storage engine that puts the safety of your data first.

## 0.2 HBase

Apache HBase[3] is the Hadoop database, a distributed, scalable, big data store. Use Apache HBase when you need random, realtime read/write access to your Big Data. This project's goal is the hosting of very large tables -- billions of rows X millions of columns -- atop clusters of commodity hardware. Apache HBase is an open-source, distributed, versioned, non-relational database modeled after Google's Bigtable: A Distributed Storage System for Structured Data by Chang et al. Just as Bigtable leverages the distributed data storage provided by the Google File System, Apache HBase provides Bigtable-like capabilities on top of Hadoop and HDFS.

## 0.3 Apache Hadoop

The Apache Hadoop[2] project develops open-source software for reliable, scalable, distributed computing. The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures. The project includes these modules: Hadoop Common: The common utilities that support the other Hadoop modules. Hadoop Distributed File System (HDFS<sup>TM</sup>): A distributed file system that provides high-throughput access to application data. Hadoop YARN: A framework for job scheduling and cluster resource management. Hadoop MapReduce: A YARN-based system for parallel processing of large data sets.

## 0.4 Oracle NoSQL Database

Oracle NoSQL Database[5] is a scalable, distributed NoSQL database, designed to provide highly reliable, flexible and available data management across a configurable set of storage nodes. Data can be modeled as relational-database-style tables, JSON documents or key-value pairs. Oracle NoSQL Database is a sharded (shared-nothing) system which distributes the data uniformly across the multiple shards in the cluster, based on the hashed value of the primary key. Within each shard, storage nodes are replicated to ensure high availability, rapid failover in the event of a node failure and optimal load balancing of queries. NoSQL Database provides Java, C, Python and Node.js drivers and a REST API to simplify application development. Oracle NoSQL Database is

integrated with a wide variety of related Oracle and open source applications in order to simplify and streamline the development and deployment of modern big data applications. Oracle NoSQL Database is available in the following editions: Enterprise Edition - Oracle Commercial License Basic Edition \* - Oracle Database Enterprise Edition Commercial License Community Edition - Open source license

## 0.5 Ranger

Apache Ranger[4] is a framework to enable, monitor and manage comprehensive data security across the Hadoop platform. The vision with Ranger is to provide comprehensive security across the Apache Hadoop ecosystem. With the advent of Apache YARN, the Hadoop platform can now support a true data lake architecture. Enterprises can potentially run multiple workloads, in a multi tenant environment. Data security within Hadoop needs to evolve to support multiple use cases for data access, while also providing a framework for central administration of security policies and monitoring of user access. Please read the FAQs if you need to understand how it works over Apache Hadoop components.

## References

- [1] *Apache CouchDB*. Web. Accessed: 2018-02-18. URL: <http://couchdb.apache.org/> (cited on page 1).
- [2] *Apache Hadoop*. Web. Accessed: 2018-02-18. URL: <http://hadoop.apache.org/> (cited on page 1).
- [3] *Apache HBase*. Web. Accessed: 2018-02-18. URL: <https://hbase.apache.org/> (cited on page 1).
- [4] *Apache Ranger*. Web. Accessed: 2018-02-18. URL: <http://ranger.apache.org/> (cited on page 2).
- [5] *Oracle Nosql Database*. Web. Accessed: 2018-02-18. URL: <http://www.oracle.com/technetwork/database/database-technologies/nosqlldb/overview/index.html> (cited on page 1).