Big Data Analytics Tool: HBase

Lecture 13

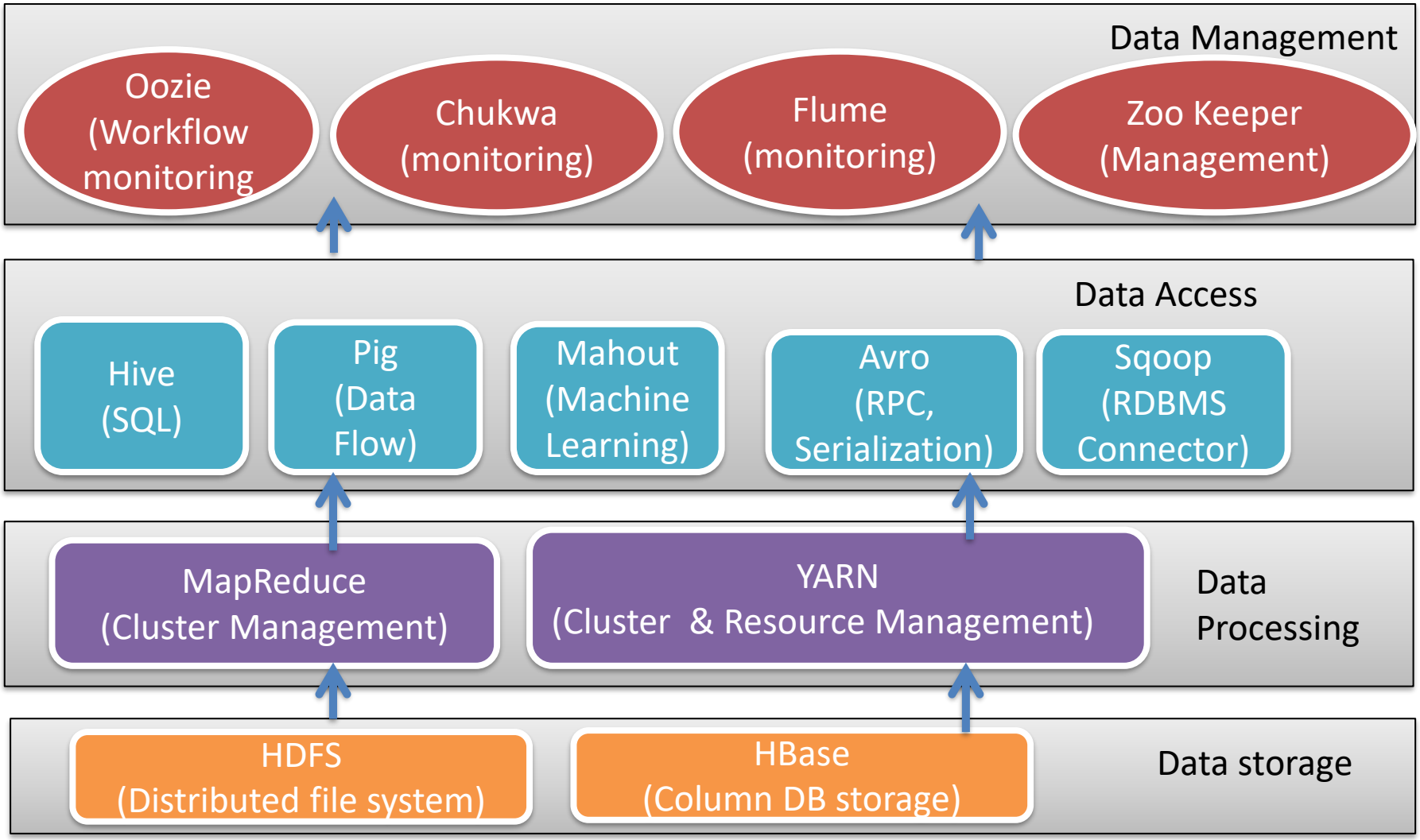
Dr. Kashish Ara Shakil
Assistant Professor
Department of Computer Science & Engineering,
Jamia Hamdard, New Delhi

HBase: Overview

- HBase is a column-oriented database management system that runs on top of Hadoop Distributed File System **(HDFS)**.
- It is well suited for sparse data sets, which are common in many big data use cases.
- Unlike relational database systems, HBase does not support a structured query language like SQL; in fact, HBase isn't a relational data store at all.
- HBase applications are written in Java much like a typical **Apache MapReduce** application.
- An HBase system comprises a set of tables.
- Each table contains rows and columns, much like a traditional database.
- Each table must have an element defined as a Primary Key, and all access attempts to HBase tables must use this Primary Key.

- HBase allows for many attributes to be grouped together into what are known as column families, such that the elements of a column family are all stored together.
- This is different from a row-oriented relational database, where all the columns of a given row are stored together. With HBase you must predefine the table schema and specify the column families.
- However, it's very flexible in that new columns can be added to families at any time, making the schema flexible and therefore able to adapt to changing application requirements.

HBase: Part of Hadoop's Ecosystem



Ref[14]

HBase Use Cases

- HBase has a market share of about 7.5% for example, approximately 6190 companies are using HBase. The industries are using HBase for time series analysis or for click stream data storage and analysis.
- Original HBase was used at Google which wanted to store massive databases for the internet and its users.

Ref[7]

Continued

- Facebook is using HBase for real-time analytics, counting Facebook likes and for messaging.
- FINRA Financial Industry Regulatory Authority using HBase to store all the trading graphs.
- Pinterest is using HBase to store the graph data.
- Flipboard is using HBase to personalize the content feed for its users.

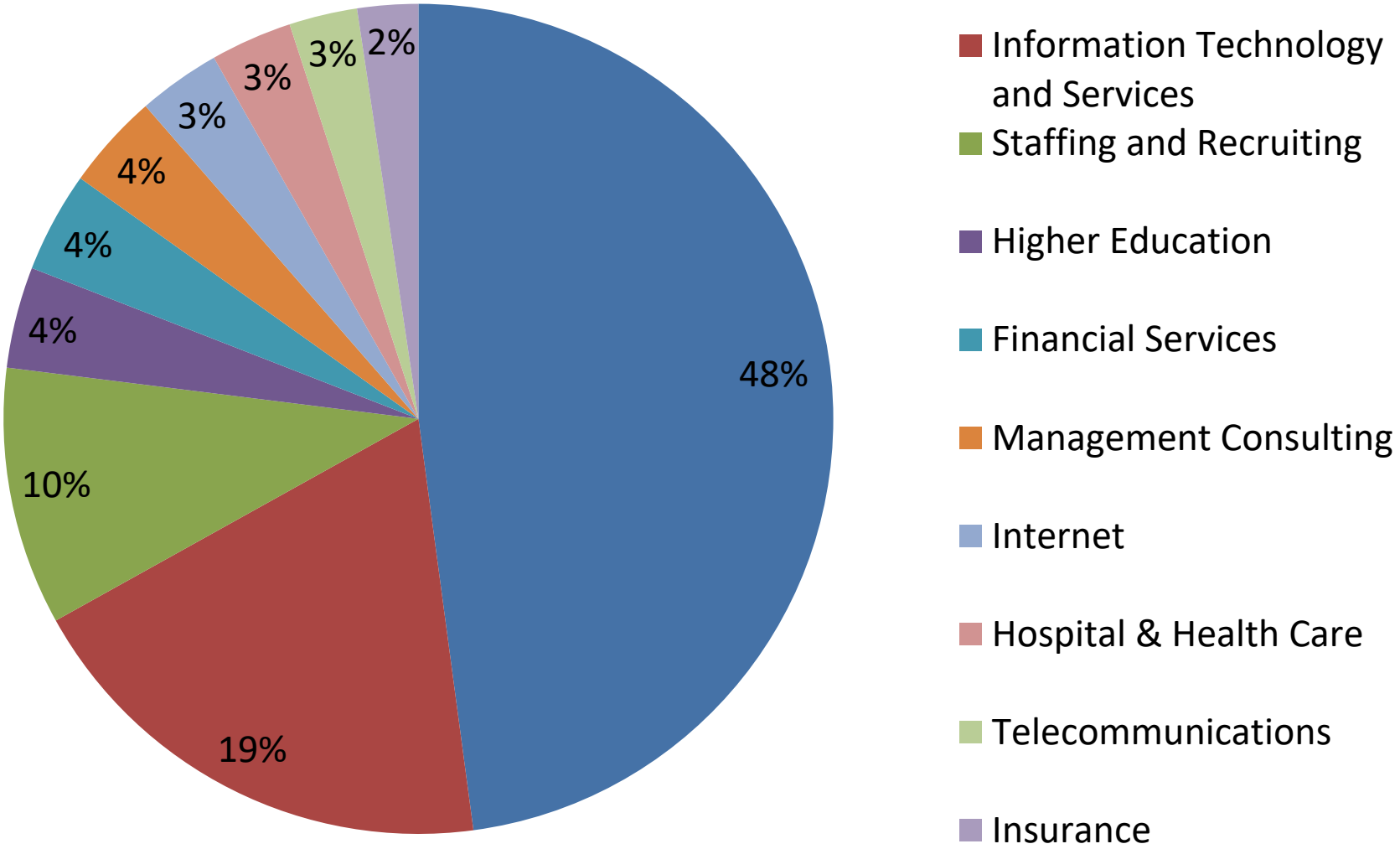
Top Companies use Apache HBase

Industry	Number of companies
Computer Software	2306
Information Technology and Services	917
Staffing and Recruiting	484
Higher Education	190
Financial Services	189
Management Consulting	180
Internet	155
Hospital & Health Care	153
Telecommunications	128
Insurance	114

Source of Data: https://idatalabs.com/tech/products/apache-hbase

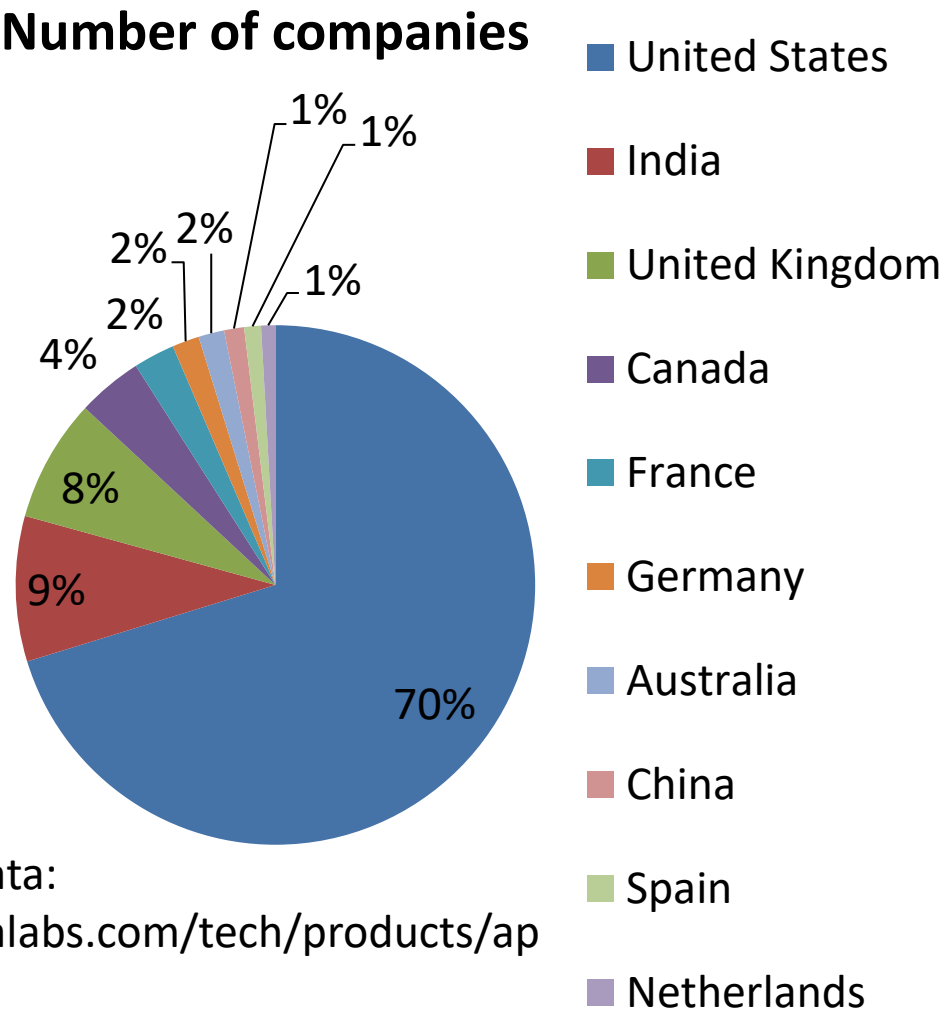
Number of companies

The percentage sharing of companies who is using HBase



Country Wise Companies use Apache HBase

Country	Number of companies
United States	4296
India	553
United Kingdom	467
Canada	248
France	157
Germany	103
Australia	98
China	76
Spain	64
Netherlands	54



Source of Data: https://idatalabs.com/tech/products/apache-hbase

Difference between Hive and HBase

Hive	HBase
Hive is query engine	HBase is a data storage particularly for unstructured data.
Apache Hive is mainly used for batch processing i.e. OLAP	HBase is extensively used for transactional processing wherein the response time of the query is not highly interactive i.e. OLTP.
Hive is to analytical queries.	HBase is to real-time querying

Hive and HBase –Better Together

- There is some limitations with Hive of high latency and HBase does not have analytical capabilities
- Integrating Hive and Hbase Technologies together is the preeminent solution.
- People working with big data have this question in their mind on –“How to use HBase from Hive?
- How well does hive and HBase work together and what is the preeminent way to use them?

Ref[7]

Continued

- Usually HBase and Hive are used together on the same Hadoop cluster.
- Hive can be used as an ETL tool for batch inserts into HBase or to execute queries that join data existing in HBase tables with the data existing in HDFS files or in external data stores.

Why use the HBase technology?

HBase is one of the core Hadoop components of the ecosystem and other two HDFS and MapReduce. It is part of the Hortonworks Data Platform Apache Hadoop ecosystem which is available as a highly secure, enterprise ready big data framework. It is being frequently deployed by some of the biggest industries like Facebook messaging system and as many.

Ref[8]

Continued

There are some of the salient features of HBase is as follows:

- It has a entirely distributed architecture and can work on extremely huge scale data.
- It works for extremely random R/W operations
- It has high security and easy Data management.
- It provides an unprecedented high write throughput
- Scaling to meet additional requirements is seamless and quick

- It can be used for both structured and semi-structured data.
- It is good when you don’t requirement of full RDBMS capabilities.
- Hbase has a perfectly modular and linear scalability feature.
- The data R/W are strictly consistent.
- The table sharding can be easily configured and automatized.
- The several servers are provided automatic failover support.
- The MapReduce tasks can be backed with HBase Tables.

What is the scope of Apache HBase?

- One of the most important HBase, it can handle data sets which number in billions of rows and millions of columns.
- It can extremely well combine the many data sources that are coming from a extensive diversity of types, structures and schemas.
- It can be integrated natively with Hadoop in order to provide a seamless fit is the best part. It also works very well with YARN. HBase gives very low latency access over fast-changing and huge amounts of data.

Ref[8]

Why need this technology and what is the problem that it is solving?

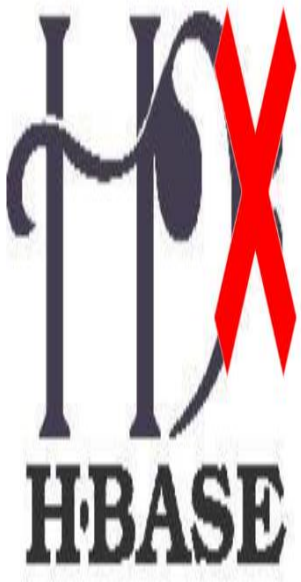
- HBase is a very progressive NoSQL database that is sighted increased usage in present today which is overwhelmed with Big Data.
- It has a very simple Java programming roots that can be deployed for scaling HBase on a big scale.

Ref[8]

- There are a lot of commercial scenarios wherein you are exclusively working with sparse data that is to look for a handful of data fields matching a certain criteria within data fields which are in the billions.
- It is extremely fault-tolerant and resilient and can work on multiple data type making it useful for varied commercial scenarios.

When not Use HBase

- When you have only few thousand/ million rows
- Lacks Command of RDBMS
- When you have hardware less than 5 data Nodes when replication factor is 3



➢ But HBase can run quit well standalone on a laptop-but this should be considered a development configuration

HBase Implementation

A number of applications including people search rely on HBase internally for data generation

Uses of HBase to power their Messages infrastructure

We use HBase as a real time data storage and analytics platform

Referred - http://wiki.apache.org/hadoop/Hbase/PoweredBy

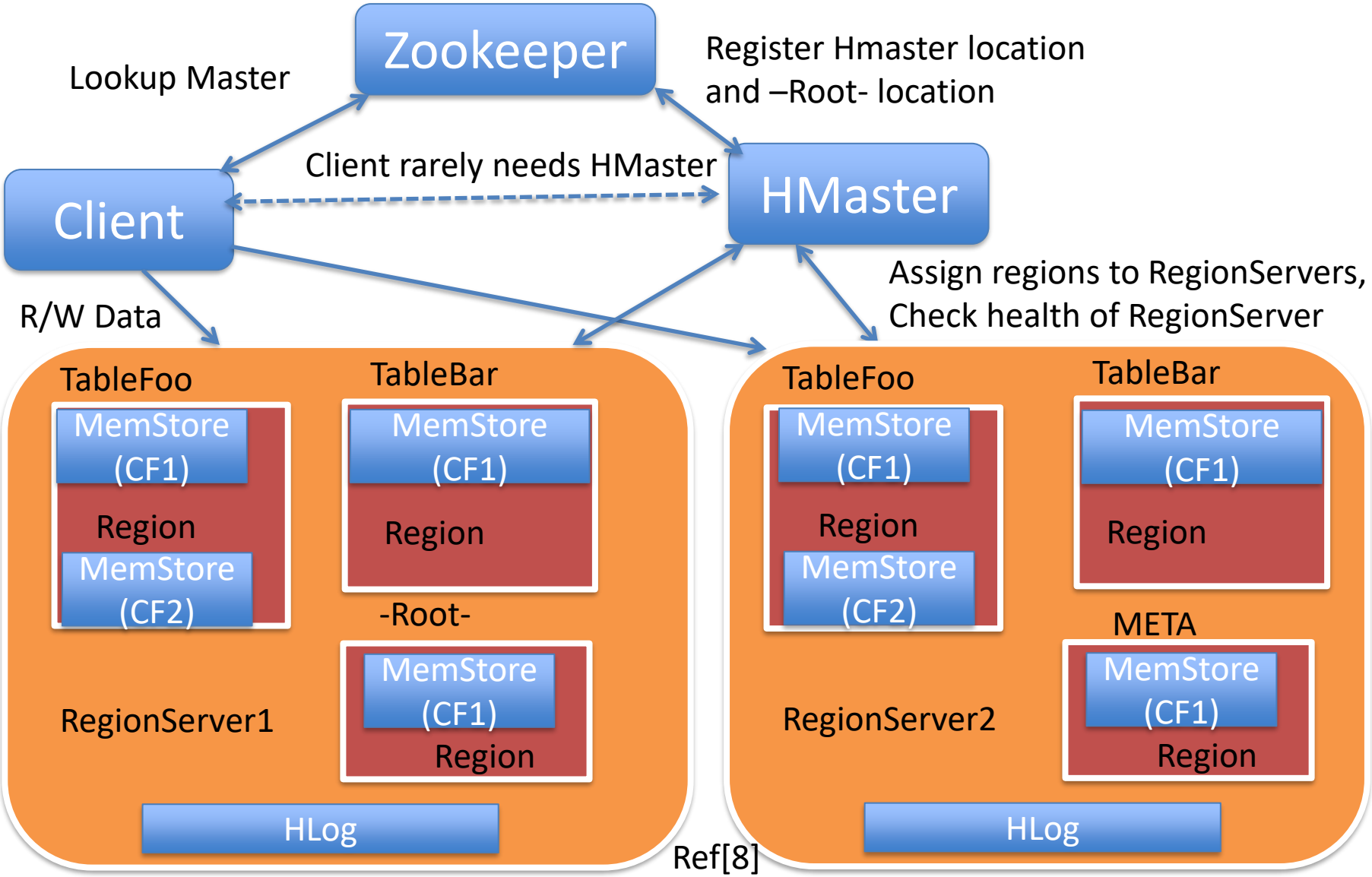
Continued

Uses HBase to store document fingerprint for detecting near-duplications. We have a cluster of few nodes that run HDFS, MapReduce, and HBase

Uses an HBase cluster containing over a billion anonymized clinical records

Uses HBase as a foundation for cloud scale storage for a variety of applications

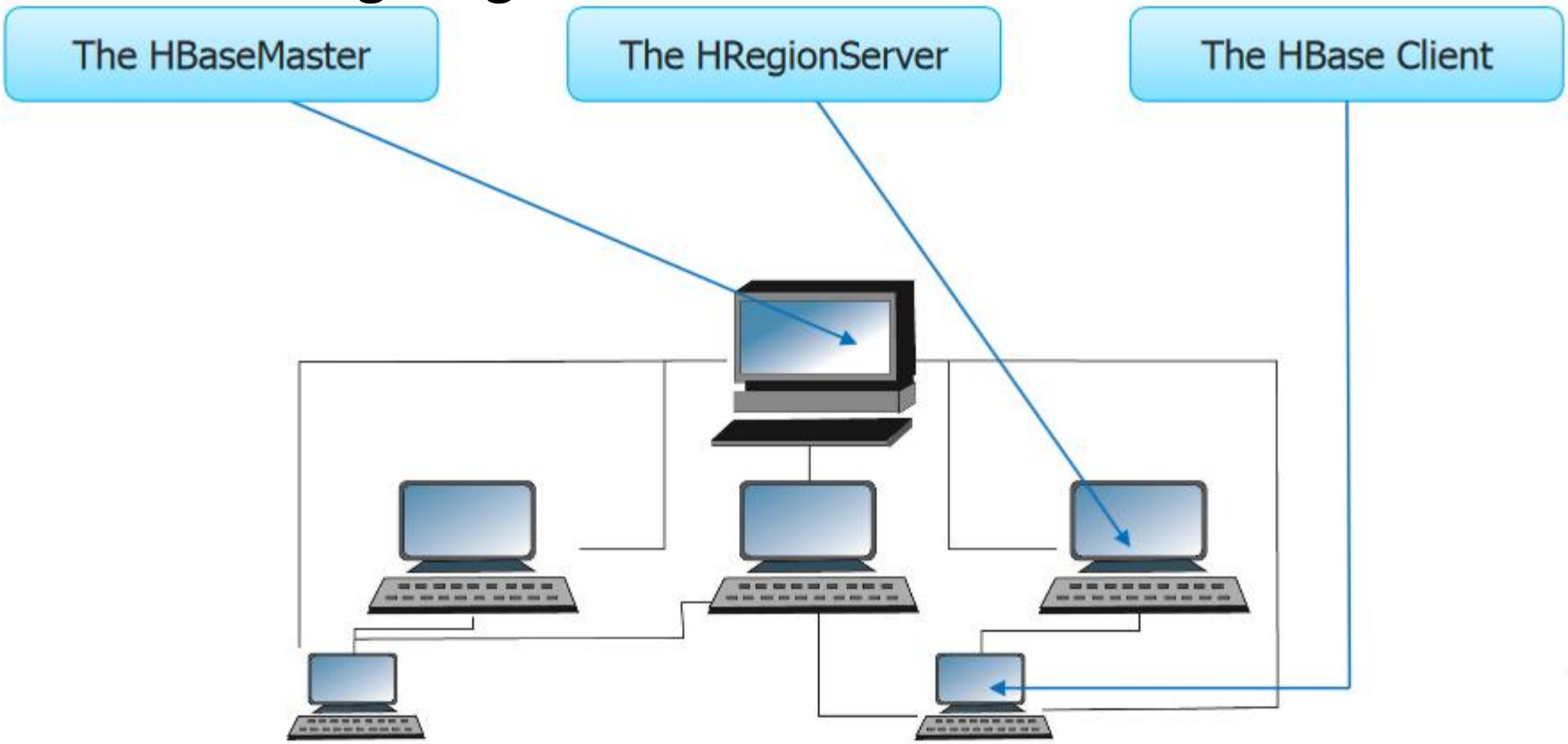
HBase Architecture



Ref[8]

Major Components of HBase

There are three major components of HBase are given in the following diagram



HBaseMaster

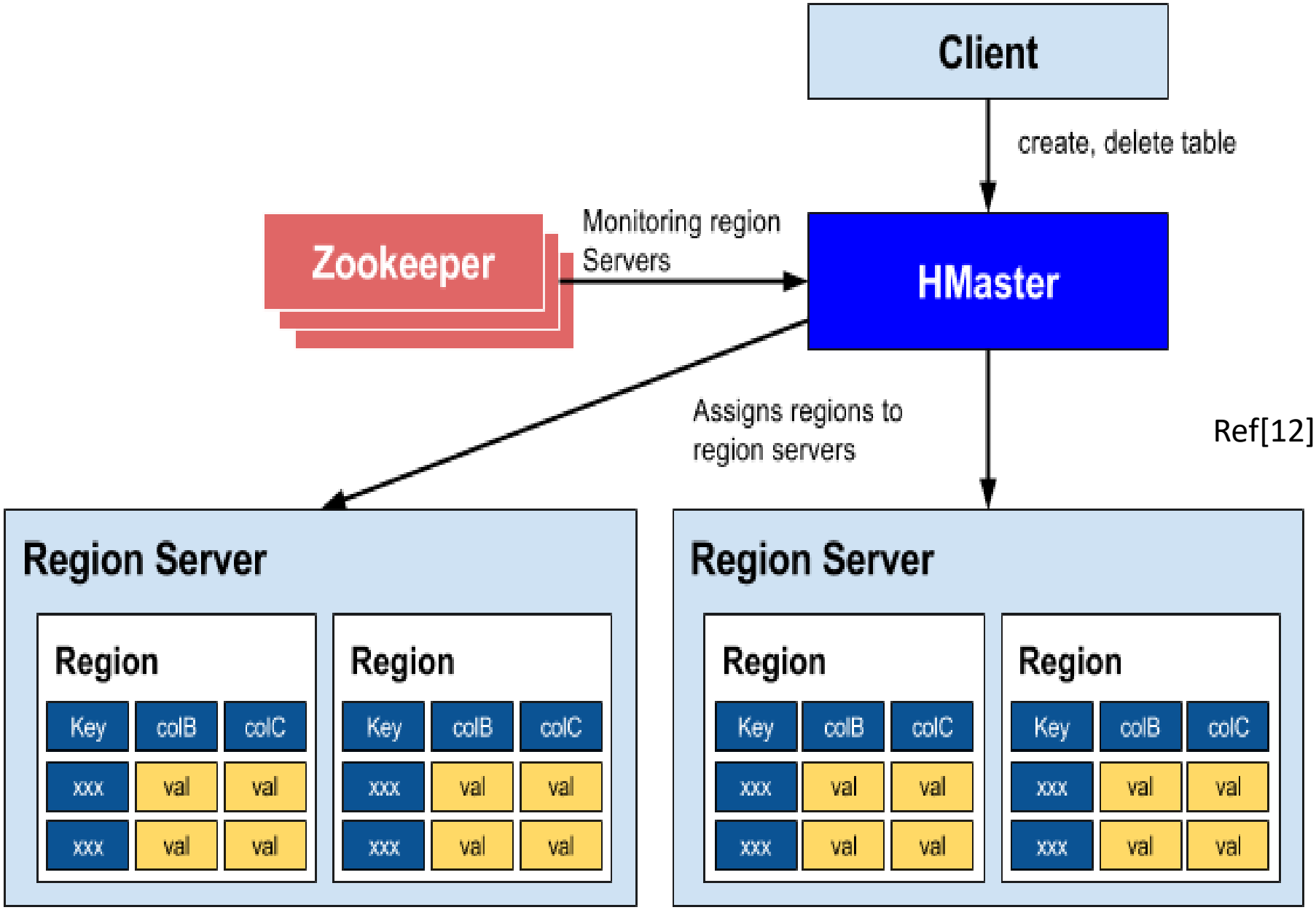
- HBase Master coordinates the HBase Cluster and is responsible for administrative operations.
- A Region Server can serve one or more Regions.
- Each Region is assigned to a Region Server on startup and the master can decide to move a Region from one Region Server to another as the result of a load balance operation.
- The Master also handles Region Server failures by assigning the region to another Region Server.

Ref[10]

HRegionServer

- HRegionServer makes a set of HRegions available to clients.
- It checks in with the HMaster. There are many HRegionServers in a single HBase deployment.
- HMaster- Region Assignment and Balancing

Ref[11]


Client
create, delete table
Zookeeper
Monitoring region Servers
HMaster
Assigns regions to region servers
Region Server
Region
Key
colB
colC
xxx
val
Region
Key
colB
colC
xxx
val
xxx
val
Region Server
Region
Key
colB
colC
xxx
val
xxx
val
Region
Key
colB
colC
xxx
val
xxx
val

Ref[12]

Architectural differences between RDBMS and HBase

Relational Databases	HBase
Uses tables as databases	Uses regions as databases
File system supported are FAT, NTFS and EXT	File System supported is HDFS
The Technique used to store logs is commit logs	The technique used to store logs is Write-Ahead Logs(WAL) Ref[13]

Continued

Relational Databases	HBase
The reference System used is coordinate system	The reference system used is Zookeeper
Uses the primary key	Uses the row key
Partitioning is supported	Sharding is supported
Use of rows, column, and cell	Use of rows, column families, column, and cells

Comparing HBase with Hadoop

HADOOP/HDFS	HBase
This provide file system for distributed storage	This provide tabular column-oriented data storage
This is optimized for storage of huge-sized files with no random r/w of these files	This is optimized for tabular data with random r/w facility
This uses flat files	This uses key-value pairs of data

Ref[13]

Continued

HADOOP/HDFS	HBase
The data model is not flexible	Provides flexible data model
This uses file system and processing framework	This uses tabular storage with built-in Hadoop MapReduce support
This is mostly optimized for write-once read-many	This optimized for both R/W many

Features of HBase

- **Automatic failover and load balancing:** The failover is assisted using RegionServer replication
- **Automatic sharding:** An HBase table is made up of regions which are accommodated by RegionServers and these regions are distributed through RegionServers on dissimilar DataNodes. HBase make available automatic and manual splitting of these regions to smaller subregions, once it reaches a threshold size to reduce I/O time and overhead.

Ref[13]

Continued

- **Linear scalability (scale out):** HBase Scaling is not scale up but scale out, that means , we do not need to make servers more powerful but we add additional machines to its cluster. We can add additional nodes to the cluster on the fly. As soon as a new RegionServer node is up, the cluster can begin re-balancing, start the RegionServer on the new node, and it is scaled up, it is as simple as that.

Continued

- **Column-oriented:** HBase stores individual column separately in contrast with most of the relational databases that uses stores or are row-based storage. Therefore in HBase, columns are stored contiguously and not the rows.
- **HBase shell support:** HBase make available a command-line tool to interact with HBase and perform operations.

Continued

- **Sparse, multidimensional, sorted map database**
- **Snapshot support:** HBase supports captivating snapshots of metadata for getting the previous or data correct state form.

References

1. https://www.tutorialspoint.com/hive/hive_introduction.htm
2. https://data-flair.training/blogs/apache-hive-architecture/
3. https://mapr.com/products/product-overview/apache-hive/
4. https://www.edureka.co/blog/hive-data-models/
5. https://www.dezyre.com/hadoop-course/hive
6. https://www.quora.com/What-are-the-advantages-of-Apache-Hive
7. https://www.dezyre.com/article/hive-vs-hbase-different-technologies-that-work-better-together/322
8. https://intellipaat.com/blog/what-is-apache-hbase/
9. https://www.slideshare.net/martyhall/hadoop-tutorial-hbase-part-1-overview
10. https://blogs.apache.org/hbase/entry/hbase_who_needs_a_master
11. https://www.cloudera.com/documentation/other/shared/CDH5-Beta-2-RNs/hbase_jdiff_report-p-cdh4.5-c-cdh5b2/cdh5b2/org/apache/hadoop/hbase/regionserver/HRegionServer.html
12. https://yeopoong.github.io/dev/2018/03/06/hbase/
13. http://www.hadoopinsight.com/blog/hbase/understanding-hbase-ecosystem/
14. https://savvycomsoftware.com/what-you-need-to-know-about-hadoop-and-its-ecosystem/

Thank you