

Polyglot Persistence on Oracle Cloud using Hadoop Map Reduce

Ms. Namrata Rawal¹, Ms. Vatika Sharma²

¹Research Scholar, Network Security, GTU PG School, Ahmedabad, Gujarat, India

²Developer, Hadoop Technology, I-verve Infoweb Company Ahmedabad, Gujarat, India

Abstract - Handling Big Data means to handle huge databases. In other words, handling of multiple data stores on multiple platforms cannot be done at a time. So, polyglot persistence came into place to handle data. It is the term that used to describe different data storage technologies to handle multiple data stores at a same time. This paper focuses on Polyglot Persistence with the map reduce on oracle cloud as it is limited in the research framework, that can not apply multiple technologies on cloud; So, I am going to apply polyglot persistence (handle multiple databases at a multiple platforms on same time) and analysis on Hadoop which is a technology that used on a distributed framework to handle bigdata. We can work on multiple nodes at time using Hadoop by creating masters and slave. In this we also analyze the performance metrics (response time and complexity).

Key Words: Datastores, polyglot persistence, map reduce, KVstore, Cygwin.

1. INTRODUCTION

Big data is not handled by traditional data management system. So technology like Hadoop is used for managing large and distributed data. It is the open source framework which offers features like Hadoop Distributed File Structure (HDFS) and Mapreduce[1] that are reliable for bigdata. But Hadoop is not able to handle multiple datastores like (SQL and NoSQL datastores) at a onetime [3]. So term Polyglot persistence is used. Polyglot persistence means simultaneous use of SQL and NoSQL(key-value datastore, column oriented datastore, document oriented datastore, graph based datastore) datastores which is gradually becoming popular in future applications development[4].

In this research we are applying Hadoop technology on polyglot persistence to get distributed environment and handling multiple datastores (SQL and NoSQL) at a one time on oracle cloud. Based on that creating bank simulator and analyzing performance

2. BACKGROUND AND RELATED WORK

2.1 Motivation to Hadoop

For handling bigdata using Hadoop technology great research work is done, it provides distributed data

storage and processing of huge data using Hadoop Distributed file system and mapreduce[1]. Hadoop addressing various challenge of bigdata like scalability, unstructured data, accessibility, real time analytics, fault tolerance and many more [2]. Hadoop contains Hadoop cluster which is used for storage of huge amount of unstructured data in distributed environment [5].

2.2. Polyglot persistence

Modern application development are mostly using data which is distributed [6] and volume of data which is going to be send is also day by day increasing so managing data is not easy task. In this data is Heterogeneous, and also the requirements like Horizontal Scalability, schematic flexibility and failure safety has become indispensable for application development. But using only single database like SQL or NoSQL database for data managing distributed data is not possible so to overcome this problem there is a term Polyglot persistence which allows handling of multiple datastores simultaneously. It is one-fits-all database development productivity and may increase performance.

2.3. Hadoop with Polyglot Persistence

2.3.1 Oracle Nosql Database

Oracle Nosql database is a type of NoSQL distributed key value database from Oracle Corporation [7]. It provides transactional semantics, horizontally scalability. It is very simple model no specific query is required to manage data in database like SQL database. NoSQL using CAP theorem and BASE property which is very reliable for transaction process [7].

2.3.2. Advantages

- Distributed environment
- Handling big Data
- Speed Processing
- Expenses are very less
- Scalable and flexible

2.3.3. Disadvantages

- Cluster management is difficult task
- Required more management as data is increasing

3. IMPLEMENTATION SCENARIO

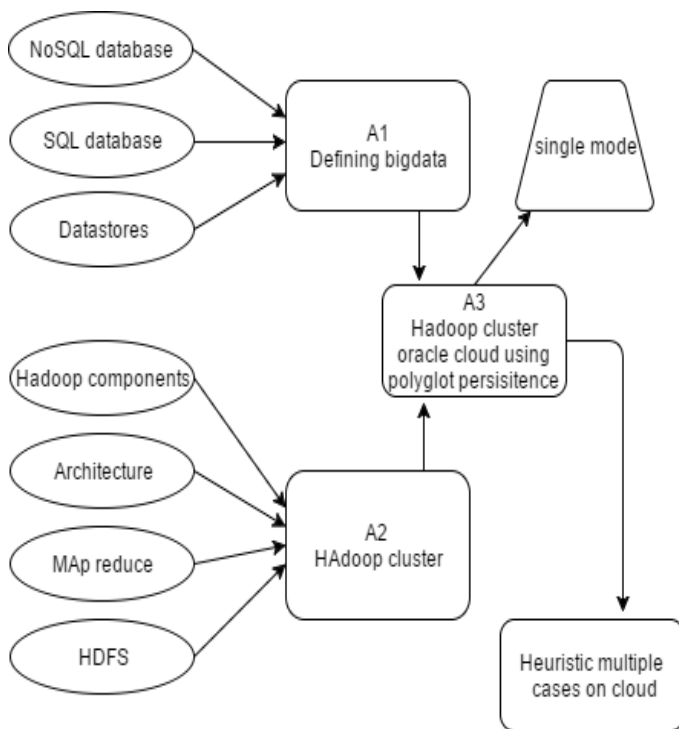


Figure 1: Implementation Scenario

Figure 1 shows implementation scenario of proposed research work. In A1 bigdata is defined which can be combination of any SQL, NoSQL data or Types of NoSQL datastores or multiple datastores. Along with that we are creating Hadoop Cluster using Cygwin. Cygwin provides linux platform on windows. Cluster is created with the help of Hadoop components, its architecture, and its features like HDFS and map reduce. After that A3- Hadoop cluster and bigdata is combined to implement polyglot persistence using MR framework and KV datastore. This is done on single mode cluster we can also do this on multi mode cluster. After successful implementation this approach is applied on case study. Here we had taken bank simulator on which concept of Hadoop technology with polyglot persistence is applied.

4. PROPOSED RESEARCH WORK

Step: 1 Create Hadoop cluster with Cygwin

Step: 2 Create KV store with Oracle

Step: 3 Applying Proposed approach on bank simulator which performs the CRUD (Create, Read, Update, Delete) operations.

Below Pseudo Code is given for Bank simulator

- **Creating Records**

Creating a new record in the store are usually identical operations. So, simply write a record to the store that uses

the appropriate key. If a record with that key does not currently exist in the store, then the record is created for us.

Mj: major component

Mi: minor component

Pseudo code for **creating new account**

```

Create (name, balance)
Acc_no ← 1
Set <Mj1,Mi1>
Acc_no ← <Mj1>
Name ← <Mi1>
// minor component can be more then and one
major will be one as it act as primary key
Create_key (Mj1,Mi1)
//while this key is already present in kvstore
increase value (accno) ← 1
set (Mj1) ← new accno value
create key (Mj1 and new Mi1)
create value with name
put this key and value to kvstore
set Mi2 ← balance
create key (Mj1 and new Mi2)
create value ← balance
put this key and value to kvstore
    
```

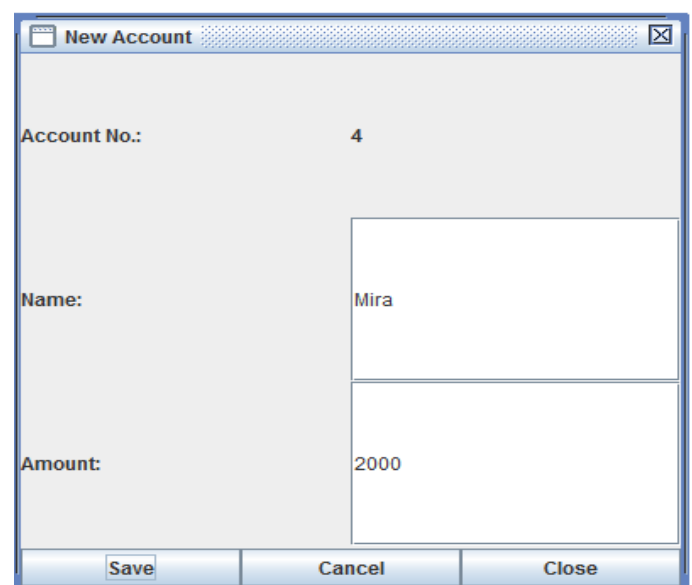


Figure: 2 creating new account

• **Reading Records**

Retrieving/Reading a record from the store, we use the KVStore.get() method. This method returns a ValueVersion object. Use ValueVersion.getValue() to return the Value object associated with the key. There are several ways to reading records from the store

1. Retrieve a single record at a time using KVStore.get()
2. Retrieve records that share a complete set of major components using either KVStore.multiGet() or KVStore.multiGetIterator().
3. Retrieve records that share a partial set of major components using KVStore.storeIterator()

Pseudo code for **Viewing account details**

```

Create(acc_no)
set (Mj1,Mi1,Mi2)
Mj1 ← accno value
Mi1 ← name
Mi2 ← balance
Create key (Mj1,Mi1)
get key ← <kvstore>
if key exists
    display value into name label
    create key ← (Mj1, Mi2)
    get value ← <kvstore> // value of key
    display value into balance label
else
    display error message
    
```

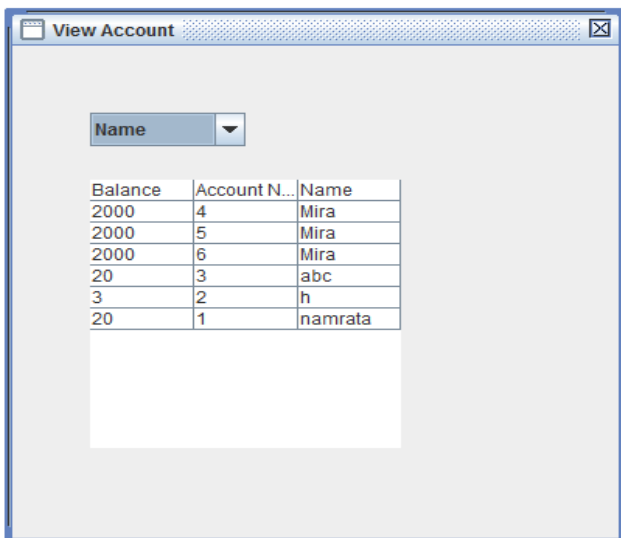


Figure: 3 viewing account details

• **Updating Records**

Updating a existing records in the store is done by simply spe with the key record exists that does use the specified key, then that record is updated with the information that we are writing to the store

Pseudo code for **Deposit amount to account**

```

Create(amount)
set (acc_no) ← value from label
set(balance) ← value from label
set(new balance) ← balance+amount
if (amt < 0)
    display error message
else
    set (Mj1,Mi2)
    Mj1 ← accno
    Mi2 ← balance
    createkey(Mj1,Mi2)
    create value(newbalance)
    put (key,value) ← kvstore // if already present
    
```

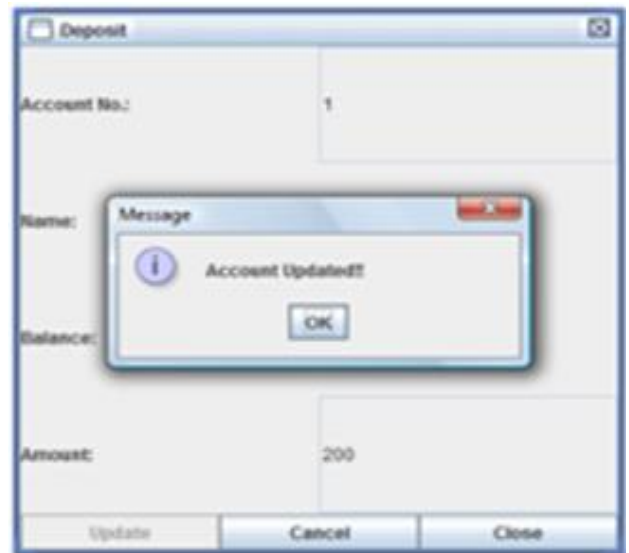


Figure 4: updating account after deposit

Pseudo code for **Withdraw amount from account**

```

create(amount)
set (acc_no) ← value from label
set(balance) ← value from label
set(new balance) ← balance+amount
    
```

```

if (amt < 0)
    display error message
else
    set (Mj1,Mi2)
    Mj ← accno
    Mi ← balance
    createkey(Mj1,Mi2)
    create value(newbalance)
    put (key,value) ← kvstore // if already present
    
```

• Delete Records

Single record can be deleted from the store using the KVStore.delete() method. Records are deleted based on a key. And multi record can be delete from the store using KVStore.multiDelete() method.

Pseudo code for deleting account

```

create(accno)
set (Mj1, Mi1,Mi2)
Mj1 ← accno
Mi1 ← name
Mi2 ← balance
createkey(Mj1,Mi1)
//delete this key form kvstore
createkey(Mj1,Mi2)
//delete this from kvstore
    
```

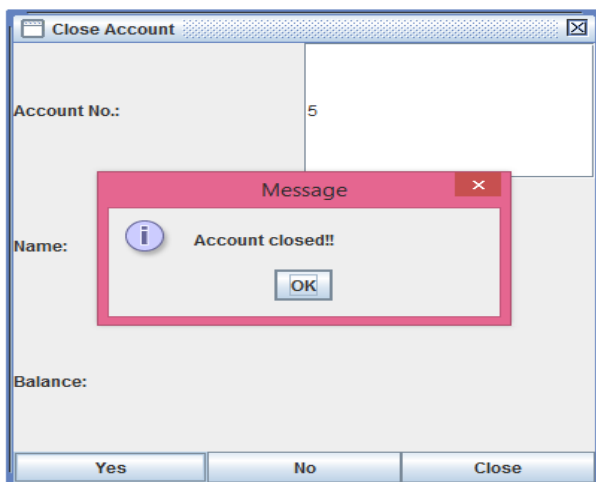


Figure 5: Deleting account

5. CONCLUSION AND FUTURE WORK

This research paper is concluded with proposing polyglot persistence with Hadoop technology using MR framework and kv store. It also gives introduction to Hadoop technology, Polyglot persistence, etc. Also Implementation scenario is given to handle bigdata using Hadoop and polyglot persistence which handles multiple datastore at multiple platforms at a one time

Everyone can handle big or huge data using Hadoop technology by creating a Hadoop cluster on cloud and also can perform CRUD operations on data. On cloud, we can store any type of data whether it is relational or non-relational data. This provides access of data to all users as it is open sourced and is implemented in java.

Further research in future is going on in this field to achieve polyglot persistence on multi node cluster and its performance analysis.

REFERENCES

- [1] Konstantin Shvachko, Hairong Kuang, Sanjay Radia, Robert Chansler, "The Hadoop Distributed File System". IEEE 2010
- [2] Aditya B. Patel, Manashvi Birla, Ushma Nair, "Addressing Big Data Problem Using Hadoop and Map Reduce". IEEE 2012
- [3] Yishan Li and Sathiamoorthy Manoharan, "A performance comparison of SQL and NoSQL databases". IEEE 2013
- [4] Juan Castrejon, Genoveva Vargas-Solar, Christine Collet, Rafael Lozano, "ExSchema: Discovering and Maintaining Schemas from Polyglot Persistence Applications". IEEE 2013
- [5] Ms. Namrata Rawal, Ms. Vatika Sharma, "Polyglot Persistence: Handling Multiple Datastores". IJARIIIE 2016
- [6] Felix Gessert - Norbert Ritter, "Polyglot persistence". Springer-Verlag Berlin Heidelberg 2015
- [7] An Oracle White Paper Febuary2011, "Hadoop and NoSQL Technologies and the Oracle database", <http://www.oracle.com/technetwork/database/hadoop-nosql-oracle-twp-398488.pdf>
- [8] Andrew J. Brust, Blue Badge Insights, Inc., "NoSQL and the Windows Azure platform", April 25, 2011
- [9] Srikrishna Prasad and NunifarSha M.S, "NextGen Data persistence Pattern in Healthcare: Polyglot persistence" in 4th ICCCNT – 2013 of IEEE