Probabilistic CAP implementation for Latency and Consistency SLAs

DESIGN FOR PERFORMANCE EVALUATION OF PCAP MODEL USING YCSB

Version 0.1

By

Nikhila Galala (nikhila.galala@sjsu.edu)

07/26/2018

Advisor: Prof. Rakesh Ranjan

Version History

Version	Changes
0.1	Initial Draft

Introduction

The project is about implementing the Probabilistic CAP model to achieve trade-off between latency and consistency. This model is evaluated using Yahoo Cloud Serving Benchmarking (YCSB). YCSB is a framework to evaluate the performance of NoSQL stores. YCSB has predefined workload scenarios for read, write and update operations. It is supports implementation of new workloads to user defines scenarios. The developed algorithm is evaluated and compared with the existing CAP model in terms of latency, consistency and throughput.

References

https://github.com/brianfrankcooper/YCSB/wiki/Getting-Started

https://github.com/brianfrankcooper/YCSB/wiki/Core-Workloads

https://github.com/brianfrankcooper/YCSB/wiki/Implementing-New-Workloads

Requirements

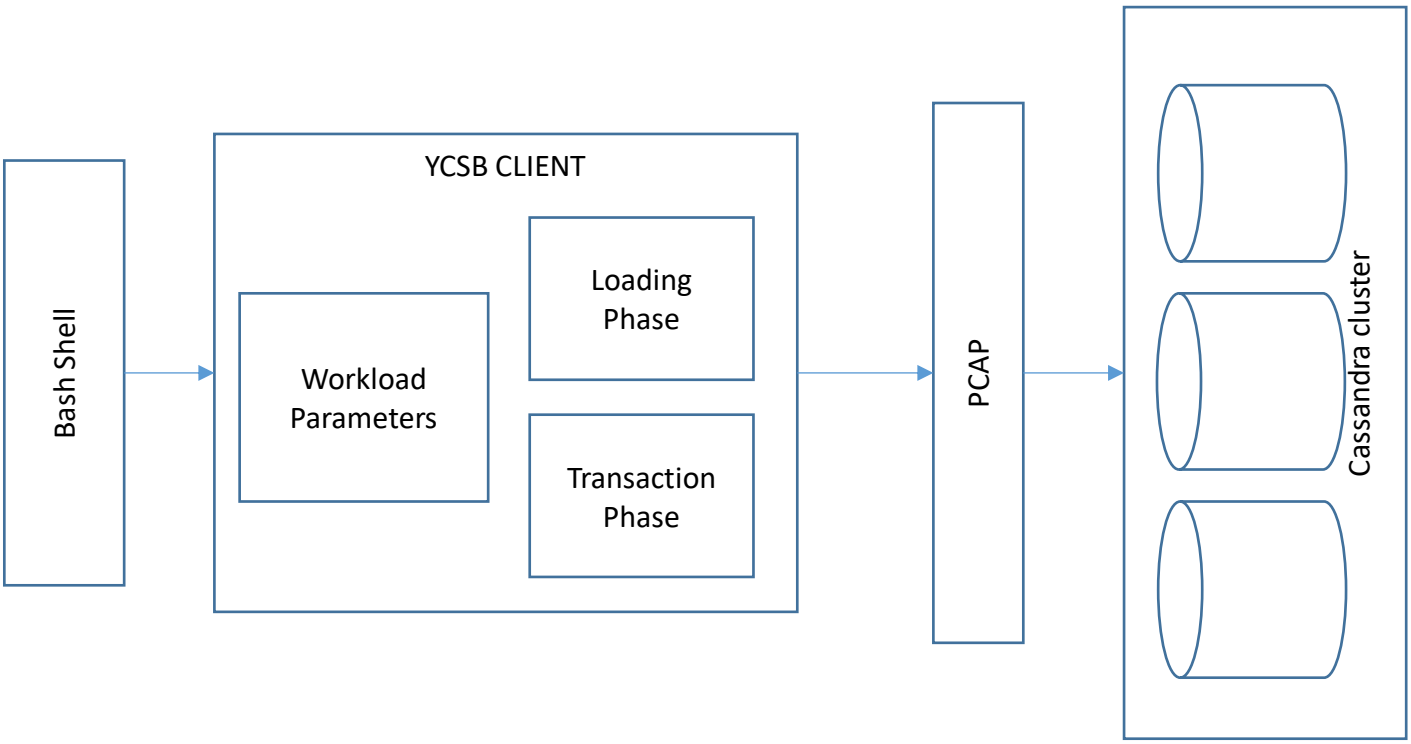
1. The initial step is to set up a Cassandra cluster. The implementation of the PCAP algorithm is executed on top of the cluster.
2. Based on the workload, the initial key spaces, tables are created to execute the workload. A sample user table is created for performing workload operation as shown in the below figure.

```
drop keyspace if exists ycsb;
CREATE KEYSPACE ycsb
WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' : 3 };

use ycsb;
create table usertable (
y_id varchar primary key,
field0 varchar,
field1 varchar,
field2 varchar,
field3 varchar,
field4 varchar,
field5 varchar,
field6 varchar,
field7 varchar,
field8 varchar,
field9 varchar);
```

3. The performance has to be tested for read, write and update operation to achieve consistency and latency trade-off

Functional Overview



The YCSB core provides workloads to evaluate the system performance. The YCSB client helps in defining new workloads that helps in benchmarking the system in different scenarios. Different parameters are configured in the configuration file which is used while loading the data and while performing operations.

a) Threads: This parameter helps in determining the number of threads using which YCSB client has to load and perform operation on the data.
b) Target: The target parameter helps to determine the number of operation that have to be performed per second.
c) Status: The status parameter provides the periodic time that YCSB client should provide

the status report of the operation performed to the user. This parameter is useful for workloads that run for long time.

These parameters are configured before running the workload. They are used in both Loading phase and the transaction phase. The IP address of the master node in the cluster is also configured in the parameter file.

There are two phases for execution of the workload

1) Loading phase
2) Transaction phase

Loading Phase:

The loading phase defines the data to be inserted before performing operation on it. The load parameter tells the YCSB client that workloadb should be loaded. -P parameter tells that parameters in config file should use while loading the data to Cassandra database

```
$ ./bin/ycsb load basic -P workloads/workloadb
```

Execution Phase:

The client executes the transaction during this phase after loading the data. Run command is used to tell the YCSB client to execute the operation defined in the workload. -P parameter tells that parameters have to loaded from config file while executing the commands. The result could be routed to a log file while can be used to analyze and compare the performance with other approches

```
$ ./bin/ycsb run basic -P workloads/workloadb > log.txt
```

Configuration/ External Interfaces

The package com.yahoo.ycsb package includes many core workloads which are used to

benchmark the NoSQL stores. These works are combination of different read, write and update operations. This package consists of 6 workloads.

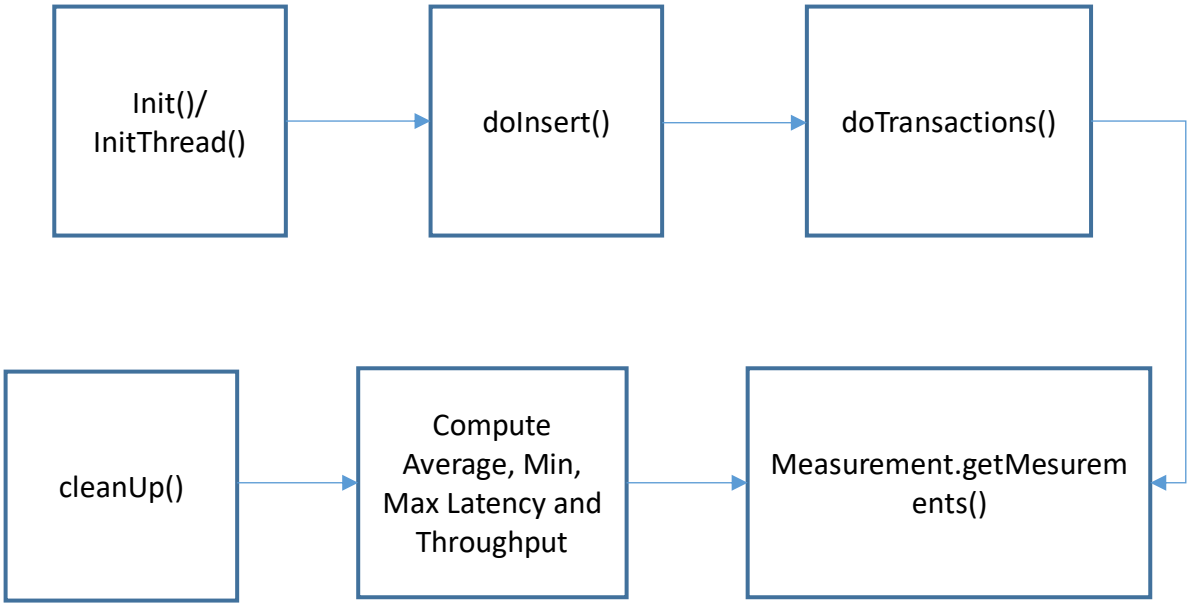
1) WorkloadA: This workload has 50 read and 50 write operations. It is a workload used to test heavy read and write operations.
2) WorkloadB: This workload is a mixture of 95 reads and 5 writes. It is mainly used to test heavy read operation application
3) WorkloadC: This workload had all read operations. It is basically used to test application that mostly read the data from the database
4) WorkloadD: This workload is used to test the latest reads. New records are inserted and the latest insertions and read for testing
5) WorkloadE: In this workload range of values are queried instead of each record. It helps to check retrieval of multiple records each time.
6) WorkloadF: This workload has read the data, modifies it and write back to the database again. It has many read, modify and write operations.

   For logging we use org.slf4j.impl.StaticLoggerBinder package. This helps us to debug the error and configure any dependencies if needed while loading and executing any operations.

Implementation

The core workloads provided by the YCSB client are simple and have small number of data operations. So, we need to define our own workloads to benchmark the PCAP application. While making our own workload, we need to configure our own data set to load the data and also define our own transaction set to perform operation on that data.

YCSB User Defined Workflow diagram



1) A parameter file in needed to define the configuration and tune the specific workload. All the configurations needed for execution of the workloads needs to be define in this parameter file.
2) com.yahoo.ycsb.Workload package is needed to create our own workload. The workload object define in the java file is created by the YCSB client and assigned to the worker object. So, if there are multiple worker threads then multiple objects are created by the YCSB client.
3) In the init() or initThread() functions of the java class the parameters are loaded and any other objects that are necessary for execution of the workload are loaded. This method is called once before the execution of the object for initialization.
4) The cleanup() method of the java class is used to destroy any of objects or for disconnection. This is called after the execution of the workload.
5) The doInsert() method of the java class contains all the necessary data that has to be loaded. So the initial data that has to be loaded here is defined in this function. The YCSB client calls this function during loading phase and loads all the data that is defined in this

function.

6) The doTransaction() method contains all the necessary transactions that have to be performed by the YCSB client. If the operation count is 1000 then the transaction defined in doTransaction() is executed 1000 times. All the operations that related to insert, update, delete data are defined in this method.
7) Later, the measurements such as Latency and throughput are calculated on the operation that are performed. The YCSB client provided the statistics of individual operations using the method Measurement.getMesurements(). We can get measurements of all operation using and calculate the overall latency and throughput of operation according the system time. We can calculate the max, min and average of the latency of operations and plots the statistics of the operation

Testing

General Approach

The PCAP implementation is tested with YCSB core workloads and also the user defined workloads increasing the count of operation to check the robustness of the implementation. Below are some of the results shown of the core workloads. The throught, min,max and average latencies of the operation performed are shown by YCSB client as part of results

Results

Below are some of the screenshots of the terminal to show the results of the core workloads on the Cassandra without PCAP implementation. We would test the results after PCAP implementation and compare the results.

Resulst of WorkloadA – Heavy Worklaod:

The average latency, throughput, the number of read and write operation is shown in the result. The min and max latency is also shown in the result.

```
[OVERALL], RunTime(ms), 5015
[OVERALL], Throughput(ops/sec), 199.40179461615153
[TOTAL_GCS_Copy], Count, 9
[TOTAL_GC_TIME_Copy], Time(ms), 28
[TOTAL_GC_TIME_%_Copy], Time(%), 0.5583250249252244
[TOTAL_GCS_MarkSweepCompact], Count, 0
[TOTAL_GC_TIME_MarkSweepCompact], Time(ms), 0
[TOTAL_GC_TIME_%_MarkSweepCompact], Time(%), 0.0
[TOTAL_GCs], Count, 9
[TOTAL_GC_TIME], Time(ms), 28
[TOTAL_GC_TIME_%], Time(%), 0.5583250249252244
[READ], Operations, 483
[READ], AverageLatency(us), 1450.8592132505175
[READ], MinLatency(us), 523
[READ], MaxLatency(us), 33791
[READ], 95thPercentileLatency(us), 2485
[READ], 99thPercentileLatency(us), 4911
[READ], Return=OK, 483
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 2233344.0
[CLEANUP], MinLatency(us), 2232320
[CLEANUP], MaxLatency(us), 2234367
[CLEANUP], 95thPercentileLatency(us), 2234367
[CLEANUP], 99thPercentileLatency(us), 2234367
[UPDATE], Operations, 517
[UPDATE], AverageLatency(us), 1359.4390715667312
[UPDATE], MinLatency(us), 522
[UPDATE], MaxLatency(us), 14423
[UPDATE], 95thPercentileLatency(us), 2503
[UPDATE], 99thPercentileLatency(us), 5067
[UPDATE], Return=OK, 517
ubuntu@ip-172-31-22-139:~/ycsb-0.14.0$
```

Results of WorkloadB – Read mostly load

```
[OVERALL], RunTime(ms), 5036
[OVERALL], Throughput(ops/sec), 198.57029388403495
[TOTAL_GCS_Copy], Count, 10
[TOTAL_GC_TIME_Copy], Time(ms), 31
[TOTAL_GC_TIME_%_Copy], Time(%), 0.6155679110405083
[TOTAL_GCS_MarkSweepCompact], Count, 0
[TOTAL_GC_TIME_MarkSweepCompact], Time(ms), 0
[TOTAL_GC_TIME_%_MarkSweepCompact], Time(%), 0.0
[TOTAL_GCs], Count, 10
[TOTAL_GC_TIME], Time(ms), 31
[TOTAL_GC_TIME_%], Time(%), 0.6155679110405083
[READ], Operations, 935
[READ], AverageLatency(us), 1378.2192513368984
[READ], MinLatency(us), 505
[READ], MaxLatency(us), 37279
[READ], 95thPercentileLatency(us), 2579
[READ], 99thPercentileLatency(us), 5143
[READ], Return=OK, 935
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 2231296.0
[CLEANUP], MinLatency(us), 2230272
[CLEANUP], MaxLatency(us), 2232319
[CLEANUP], 95thPercentileLatency(us), 2232319
[CLEANUP], 99thPercentileLatency(us), 2232319
[UPDATE], Operations, 65
[UPDATE], AverageLatency(us), 1588.6923076923076
[UPDATE], MinLatency(us), 645
[UPDATE], MaxLatency(us), 9655
[UPDATE], 95thPercentileLatency(us), 3503
[UPDATE], 99thPercentileLatency(us), 4459
[UPDATE], Return=OK, 65
ubuntu@ip-172-31-22-139:~/ycsb-0.14.0$
```

Results of WorkloadC – Read-only Workload

```
[OVERALL], RunTime(ms), 5243
[OVERALL], Throughput(ops/sec), 190.7304978065993
[TOTAL_GCS_Copy], Count, 10
[TOTAL_GC_TIME_Copy], Time(ms), 29
[TOTAL_GC_TIME_%_Copy], Time(%), 0.553118443639138
[TOTAL_GCS_MarkSweepCompact], Count, 0
[TOTAL_GC_TIME_MarkSweepCompact], Time(ms), 0
[TOTAL_GC_TIME_%_MarkSweepCompact], Time(%), 0.0
[TOTAL_GCs], Count, 10
[TOTAL_GC_TIME], Time(ms), 29
[TOTAL_GC_TIME_%], Time(%), 0.553118443639138
[READ], Operations, 1000
[READ], AverageLatency(us), 1568.721
[READ], MinLatency(us), 521
[READ], MaxLatency(us), 53087
[READ], 95thPercentileLatency(us), 3789
[READ], 99thPercentileLatency(us), 8123
[READ], Return=OK, 1000
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 2237440.0
[CLEANUP], MinLatency(us), 2236416
[CLEANUP], MaxLatency(us), 2238463
[CLEANUP], 95thPercentileLatency(us), 2238463
[CLEANUP], 99thPercentileLatency(us), 2238463
```

Results of WorkloadD- Read latest workload

```
[OVERALL], RunTime(ms), 5243
[OVERALL], Throughput(ops/sec), 190.7304978065993
[TOTAL_GCS_Copy], Count, 10
[TOTAL_GC_TIME_Copy], Time(ms), 30
[TOTAL_GC_TIME_%_Copy], Time(%), 0.5721914934197978
[TOTAL_GCS_MarkSweepCompact], Count, 0
[TOTAL_GC_TIME_MarkSweepCompact], Time(ms), 0
[TOTAL_GC_TIME_%_MarkSweepCompact], Time(%), 0.0
[TOTAL_GCs], Count, 10
[TOTAL_GC_TIME], Time(ms), 30
[TOTAL_GC_TIME_%], Time(%), 0.5721914934197978
[READ], Operations, 959
[READ], AverageLatency(us), 1570.7267987486966
[READ], MinLatency(us), 533
[READ], MaxLatency(us), 36383
[READ], 95thPercentileLatency(us), 3249
[READ], 99thPercentileLatency(us), 8727
[READ], Return=OK, 959
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 2247680.0
[CLEANUP], MinLatency(us), 2246656
[CLEANUP], MaxLatency(us), 2248703
[CLEANUP], 95thPercentileLatency(us), 2248703
[CLEANUP], 99thPercentileLatency(us), 2248703
[INSERT], Operations, 41
[INSERT], AverageLatency(us), 2385.5853658536585
[INSERT], MinLatency(us), 785
[INSERT], MaxLatency(us), 9023
[INSERT], 95thPercentileLatency(us), 5491
[INSERT], 99thPercentileLatency(us), 9023
[INSERT], Return=OK, 41
ubuntu@ip-172-31-22-139:~/ycsb-0.14.0$
```

Results of WorkloadE – Test Range of Queries

```
[OVERALL], RunTime(ms), 7610
[OVERALL], Throughput(ops/sec), 131.4060446780552
[TOTAL_GCS_Copy], Count, 43
[TOTAL_GC_TIME_Copy], Time(ms), 74
[TOTAL_GC_TIME_%_Copy], Time(%), 0.9724047306176085
[TOTAL_GCS_MarkSweepCompact], Count, 0
[TOTAL_GC_TIME_MarkSweepCompact], Time(ms), 0
[TOTAL_GC_TIME_%_MarkSweepCompact], Time(%), 0.0
[TOTAL_GCs], Count, 43
[TOTAL_GC_TIME], Time(ms), 74
[TOTAL_GC_TIME_%], Time(%), 0.9724047306176085
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 2245632.0
[CLEANUP], MinLatency(us), 2244608
[CLEANUP], MaxLatency(us), 2246655
[CLEANUP], 95thPercentileLatency(us), 2246655
[CLEANUP], 99thPercentileLatency(us), 2246655
[INSERT], Operations, 59
[INSERT], AverageLatency(us), 4334.610169491525
[INSERT], MinLatency(us), 612
[INSERT], MaxLatency(us), 35871
[INSERT], 95thPercentileLatency(us), 10671
[INSERT], 99thPercentileLatency(us), 15959
[INSERT], Return=OK, 59
[SCAN], Operations, 941
[SCAN], AverageLatency(us), 3905.0255047821465
[SCAN], MinLatency(us), 513
[SCAN], MaxLatency(us), 49663
[SCAN], 95thPercentileLatency(us), 11703
[SCAN], 99thPercentileLatency(us), 17391
[SCAN], Return=OK, 941
ubuntu@ip-172-31-22-139:~/ycsb-0.14.0$
```

Results of WorkloadF – Test Read-modify-write operations

```
[OVERALL], RunTime(ms), 5558
[OVERALL], Throughput(ops/sec), 179.92083483267362
[TOTAL_GCS_Copy], Count, 11
[TOTAL_GC_TIME_Copy], Time(ms), 32
[TOTAL_GC_TIME_%_Copy], Time(%), 0.5757466714645556
[TOTAL_GCS_MarkSweepCompact], Count, 0
[TOTAL_GC_TIME_MarkSweepCompact], Time(ms), 0
[TOTAL_GC_TIME_%_MarkSweepCompact], Time(%), 0.0
[TOTAL_GCs], Count, 11
[TOTAL_GC_TIME], Time(ms), 32
[TOTAL_GC_TIME_%], Time(%), 0.5757466714645556
[READ], Operations, 1000
[READ], AverageLatency(us), 1293.352
[READ], MinLatency(us), 552
[READ], MaxLatency(us), 35647
[READ], 95thPercentileLatency(us), 2501
[READ], 99thPercentileLatency(us), 4935
[READ], Return=OK, 1000
[READ-MODIFY-WRITE], Operations, 507
[READ-MODIFY-WRITE], AverageLatency(us), 2561.2031558185404
[READ-MODIFY-WRITE], MinLatency(us), 1131
[READ-MODIFY-WRITE], MaxLatency(us), 15295
[READ-MODIFY-WRITE], 95thPercentileLatency(us), 4951
[READ-MODIFY-WRITE], 99thPercentileLatency(us), 6827
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 2233344.0
[CLEANUP], MinLatency(us), 2232320
[CLEANUP], MaxLatency(us), 2234367
[CLEANUP], 95thPercentileLatency(us), 2234367
[CLEANUP], 99thPercentileLatency(us), 2234367
[UPDATE], Operations, 507
[UPDATE], AverageLatency(us), 1284.5207100591715
[UPDATE], MinLatency(us), 538
[UPDATE], MaxLatency(us), 13031
[UPDATE], 95thPercentileLatency(us), 2407
[UPDATE], 99thPercentileLatency(us), 4631
[UPDATE], Return=OK, 507
ubuntu@ip-172-31-22-139:~/ycsb-0.14.0$
```