

Evaluation of Different Key/Value Storage Systems on Amazon EC2.

Aditya Kale (A20355807)

Department of Computer Science, Illinois Institute Of Technology, Chicago, Illinois.

Abstract

The purpose of this assignment is to compare three high end key value storage systems with a Distributed hash Table storage system which I developed in programming assignment 2 PA#2. In this experiment 10,000 Put, Get and Delete operations are performed on these storage systems. These experiment are performed on Amazon EC2 instances by increasing number of concurrent requests, by increasing number of concurrent clients. Results for Latency and Throughput of different operations are compared for different systems on different number of concurrent clients and their graphs are plotted. Also Average Latency and Throughput for all different key-value storage systems are compared. After comparing all systems it is found that Redis and PA#2 has very low latency as compared to MongoDB and CouchDB. CouchDB has highest latency and thus has lowest throughput.

Experiment Setup

Commercial Cloud:

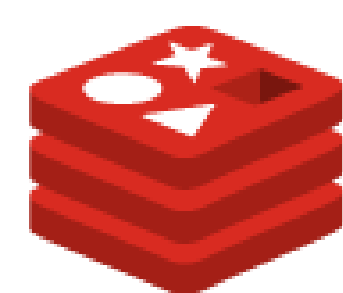
- Amazon EC2
- m3.Medium instance
 - 1 Compute Unit
 - 3.75 GB Ram
- 64-bit Linux machine

Commodity Cluster:

- Upto 16 instances.

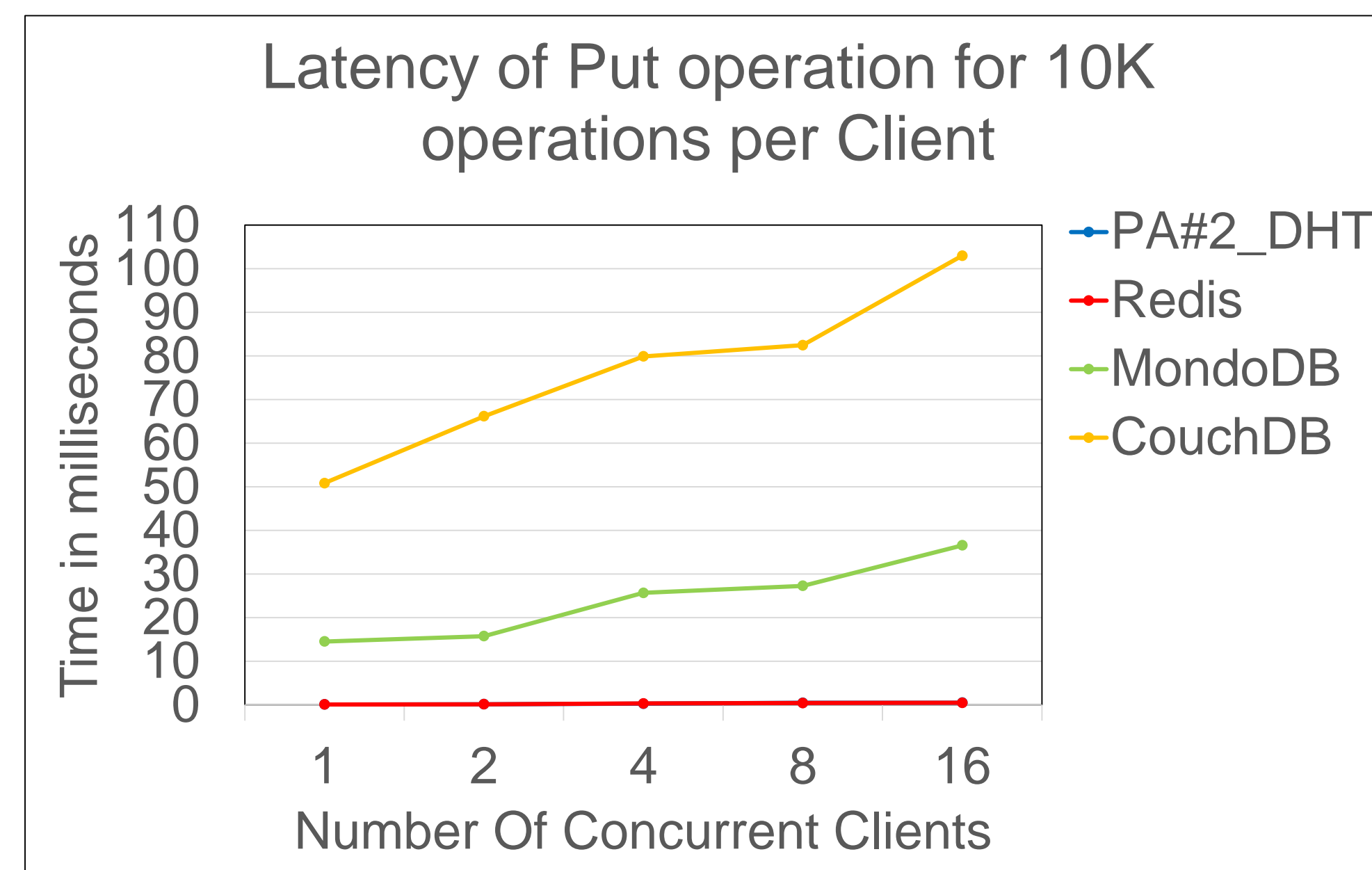
Benchmark settings:

- N clients to N servers: All-to-All communication pattern.
- Random generated key-value pairs: 10 bytes key, 90 bytes value.
- All 16 EC2 environment are setup with all different key value storage systems and perform 10K operations to measure latency and throughput of systems.

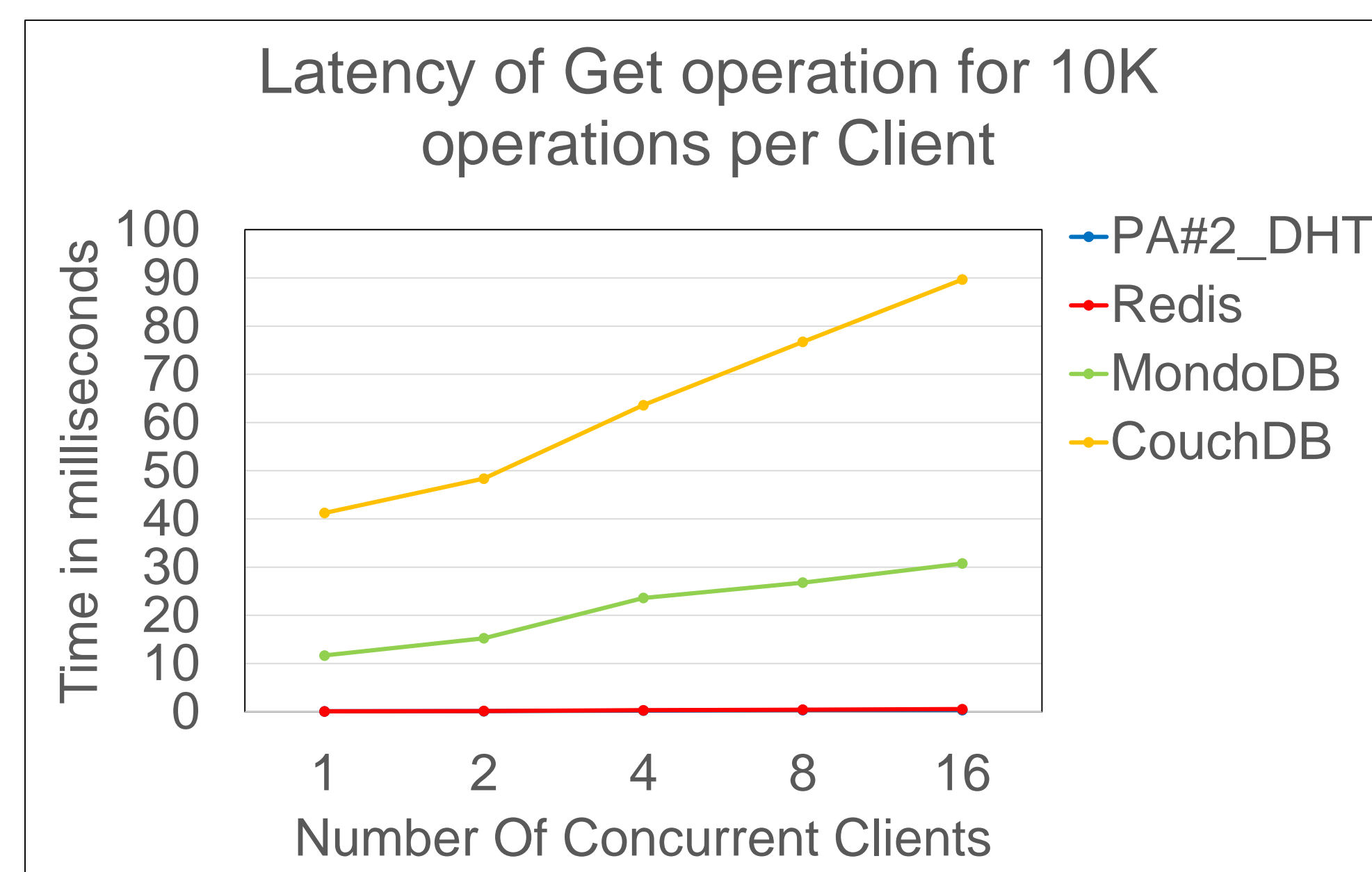


Latency of a different operations on different storage systems for 10K operations.

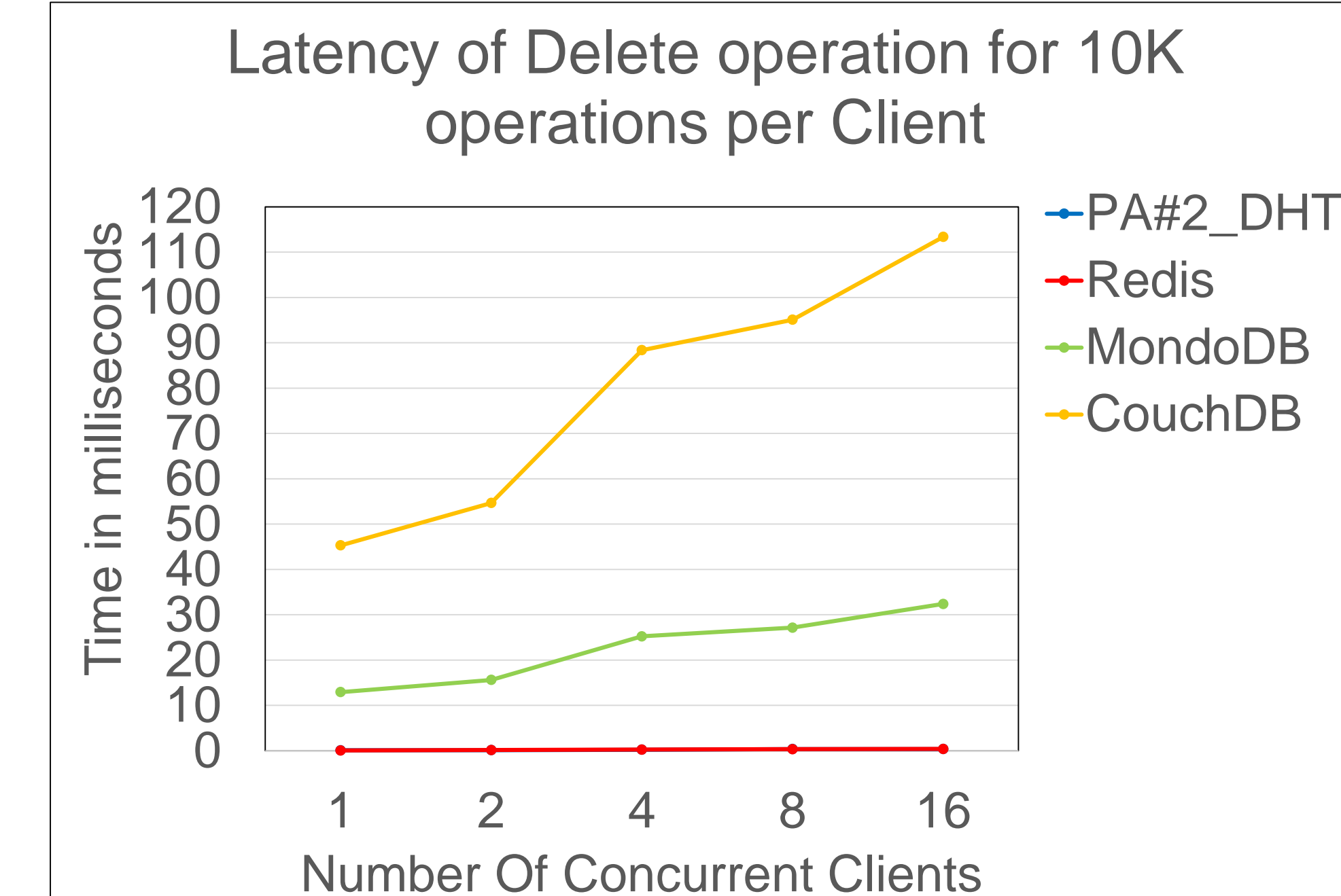
Latency for Put Operations



Latency for Get Operations

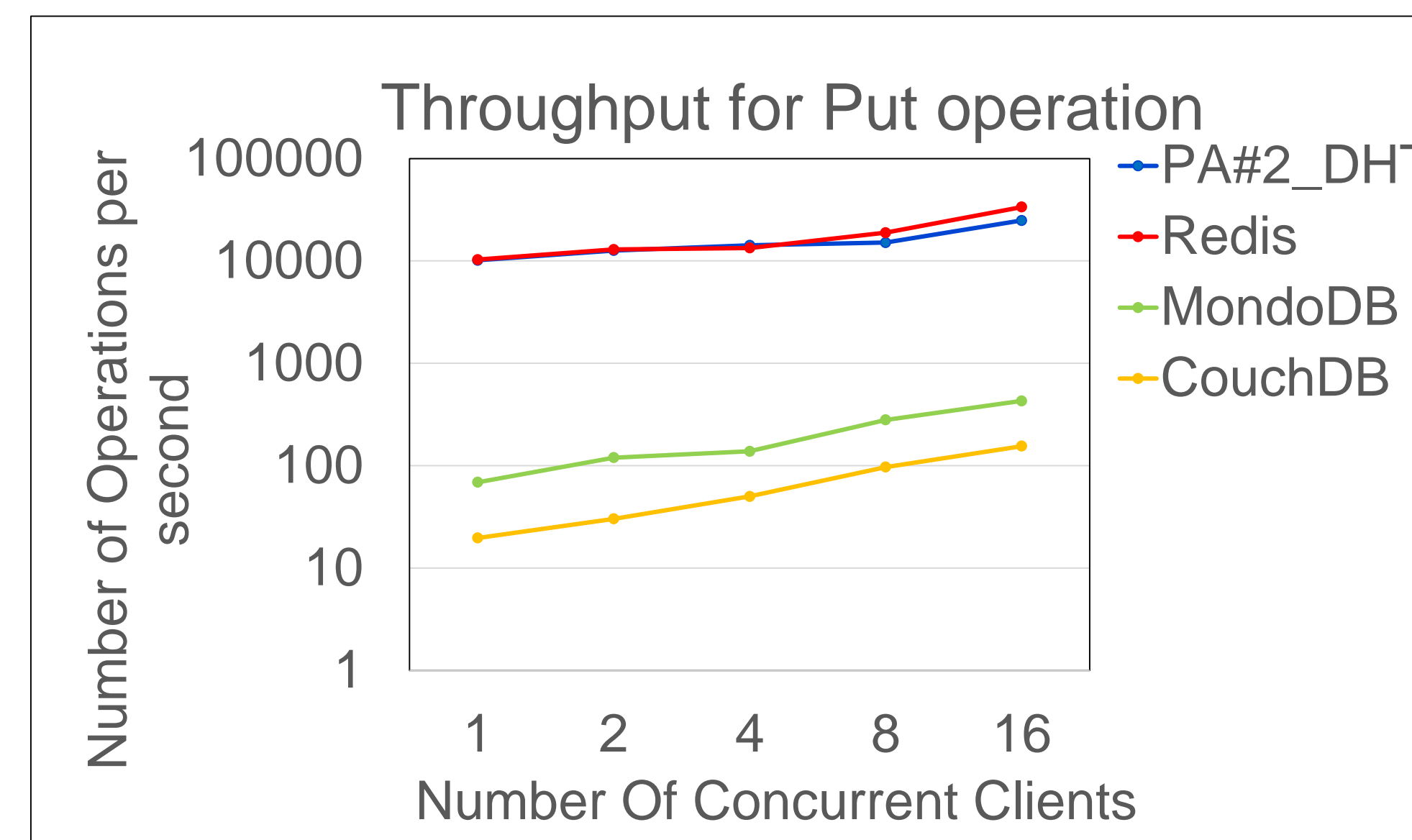


Latency for Delete Operations

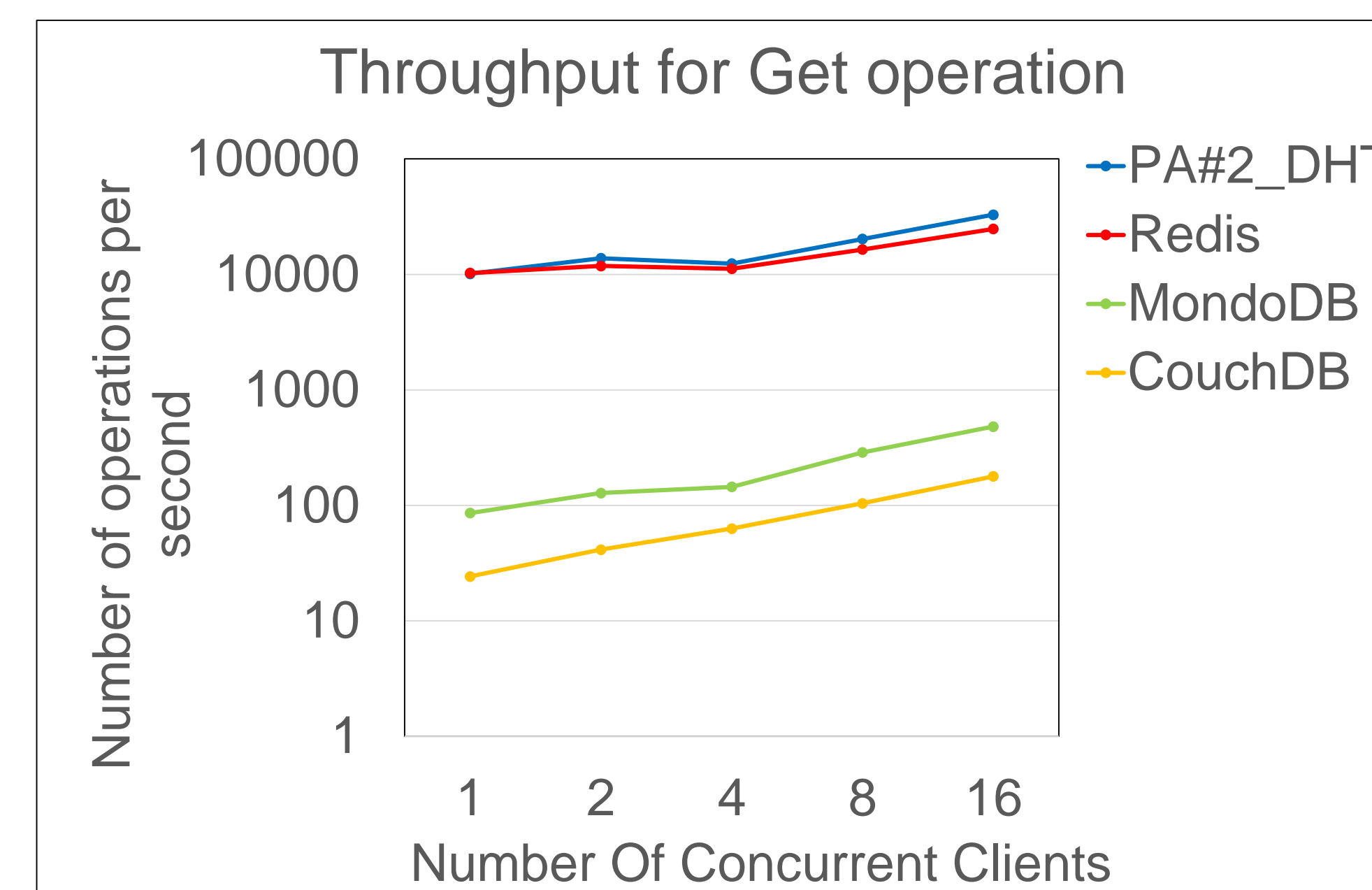


Throughput of a different operations on different storage systems for 10K operations.

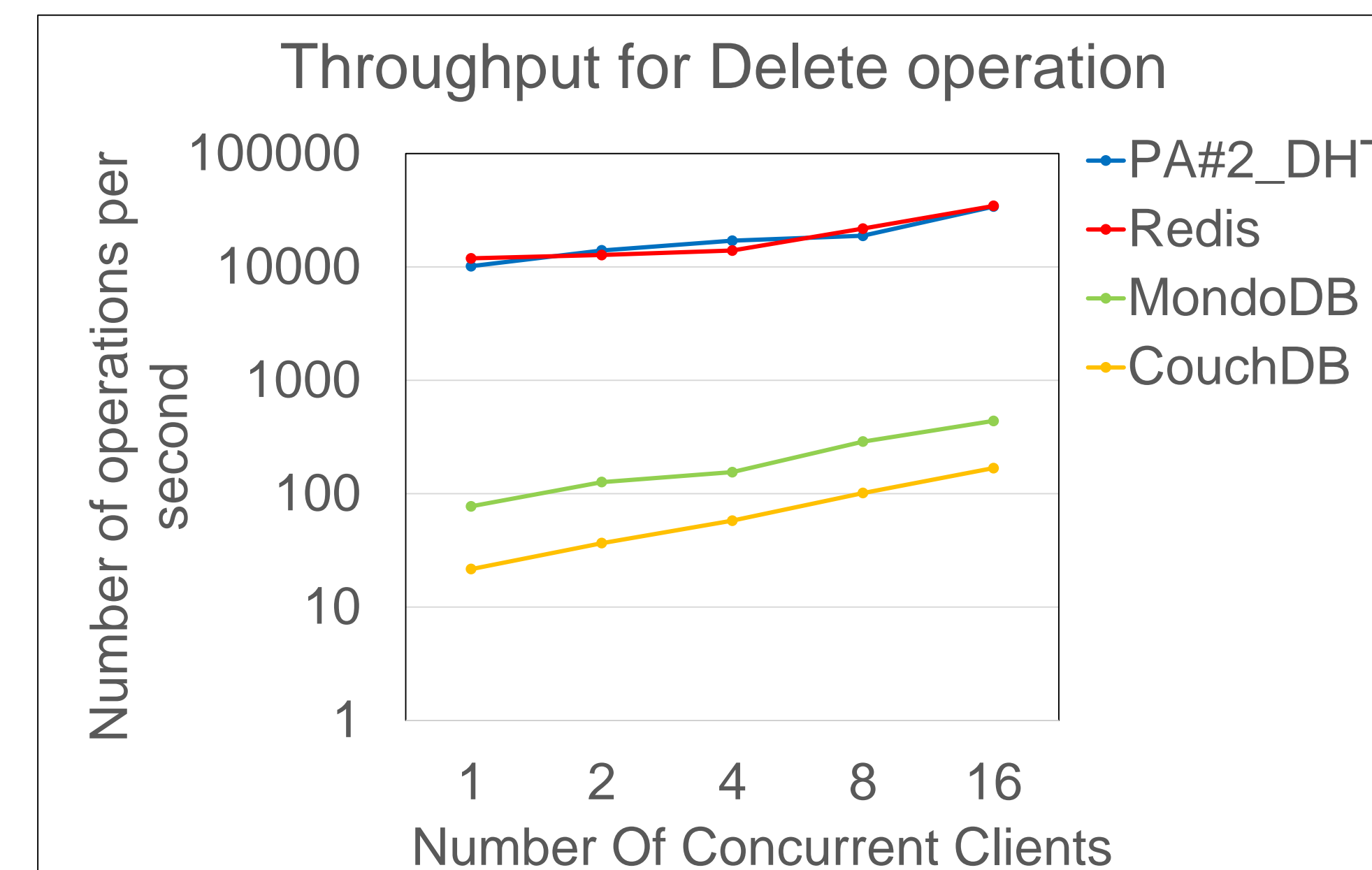
Throughput for Put Operations



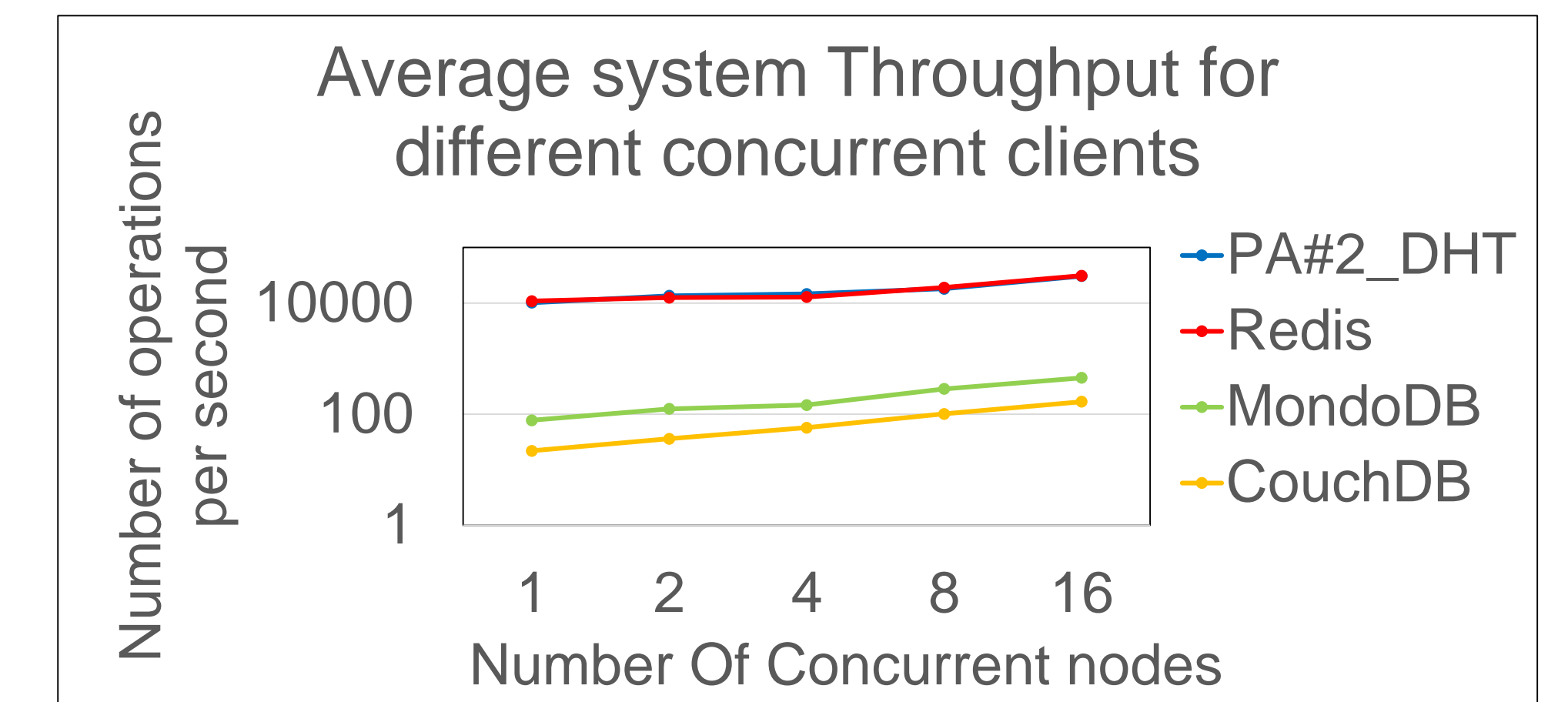
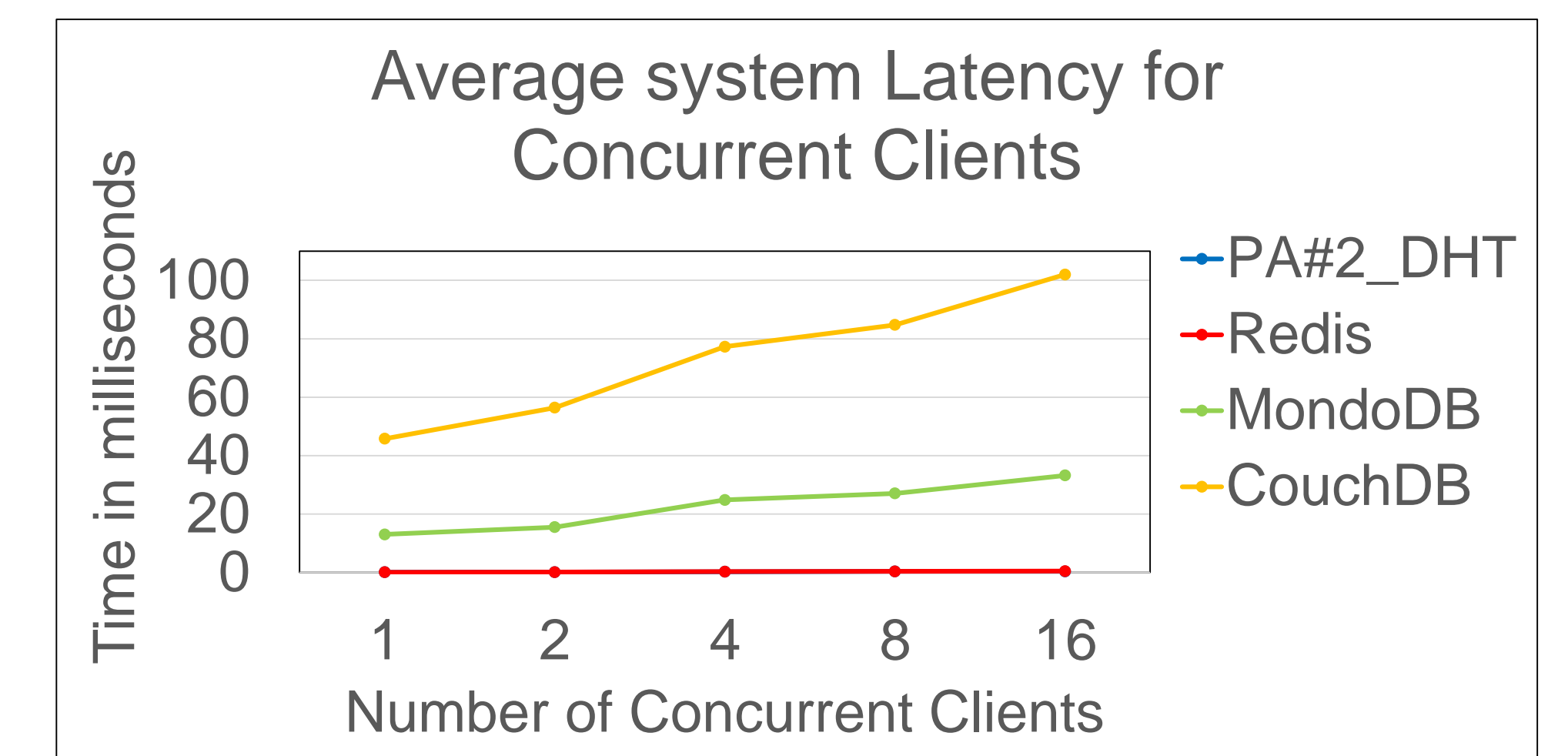
Throughput for Get Operations



Throughput for Delete Operations



Average Latency and Throughput for different storage systems.



Conclusions

Different Key/value Storage Systems				
Category	PA#2_DHT	Redis	MondoDB	CouchDB
Latency	Low	Low	High	High
Throughput	High	High	Low	Low
Consistency	-	Delayed	Quick	Eventual
Storage Type	Hash Table (In memory)	Hash Table (In memory)	Documents	Documents
For Large Amount of Data	Bad	Bad	Good	Good

- Comparing all the system we can conclude that for small amount of data we can use Redis which has delayed persistence and thus perform fast, however for large amount of data and good persistence MongoDB and CouchDB are better choice.

References

- <https://docs.mongodb.org/manual/tutorial/install-mongodb-on-ubuntu/>
- <http://redis.io/topics/quickstart>