94

An Empirical Evaluation of Distributed Key/Value Storage Systems

Avinash Singh (A2035276)

CS-550 AOS, Department of Computer Science, Illinois Institute of Technology

Abstract

Distributed hash table is class for decentralized system that provides lookup services with key-value pairs. This distributed hash table stores key-value pair and any peer node can efficiently retrieve the value depending on the given key. Distributed hash table is responsible for maintaining the mapping from keys to values is distributed among the nodes in such a way that the particular set of key-value pair will always route a particular node only. Such a implementation allows distributed hash table to scale to large numbers of nodes & to handle addition, removals. Such system forms an infrastructure that can be used to build complex services such as peer-to-peer file sharing , cooperative web caching, content distribution systems.

Motivation and Objectives

To Evaluate throughput and latency of distributed key/ Value Systems like MongoDb, Redis, Couch & PA2-DHT on Amazon EC2 and to compare all.

System under test

1. MongoDB
2. Redis
3. Couch
4. PA2-DHT

Experiment

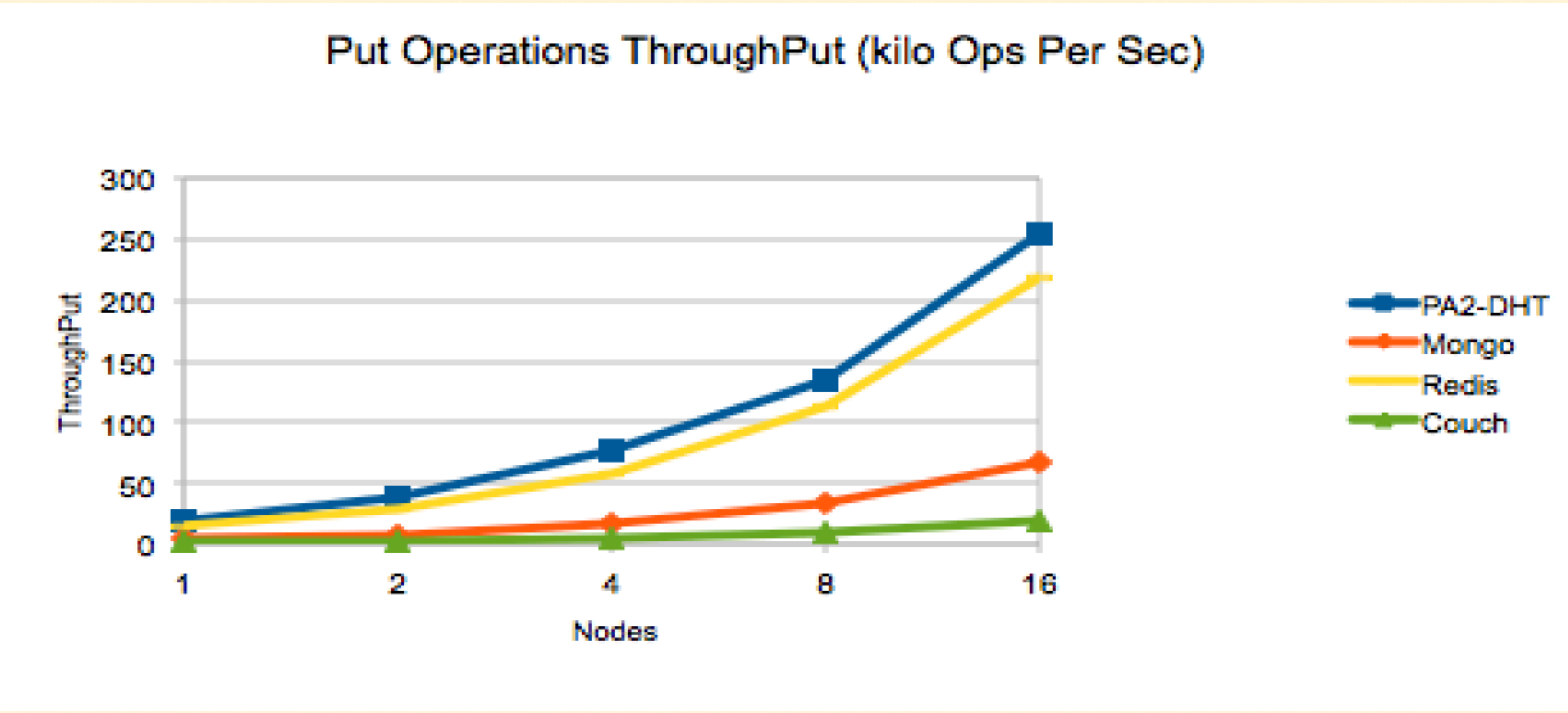
1. Experiment is done on Amazon EC2.
2. M3.medium instance is used.
3. Totall 16 instance for one system is tested with 100k request per instance.

Steps for experiment

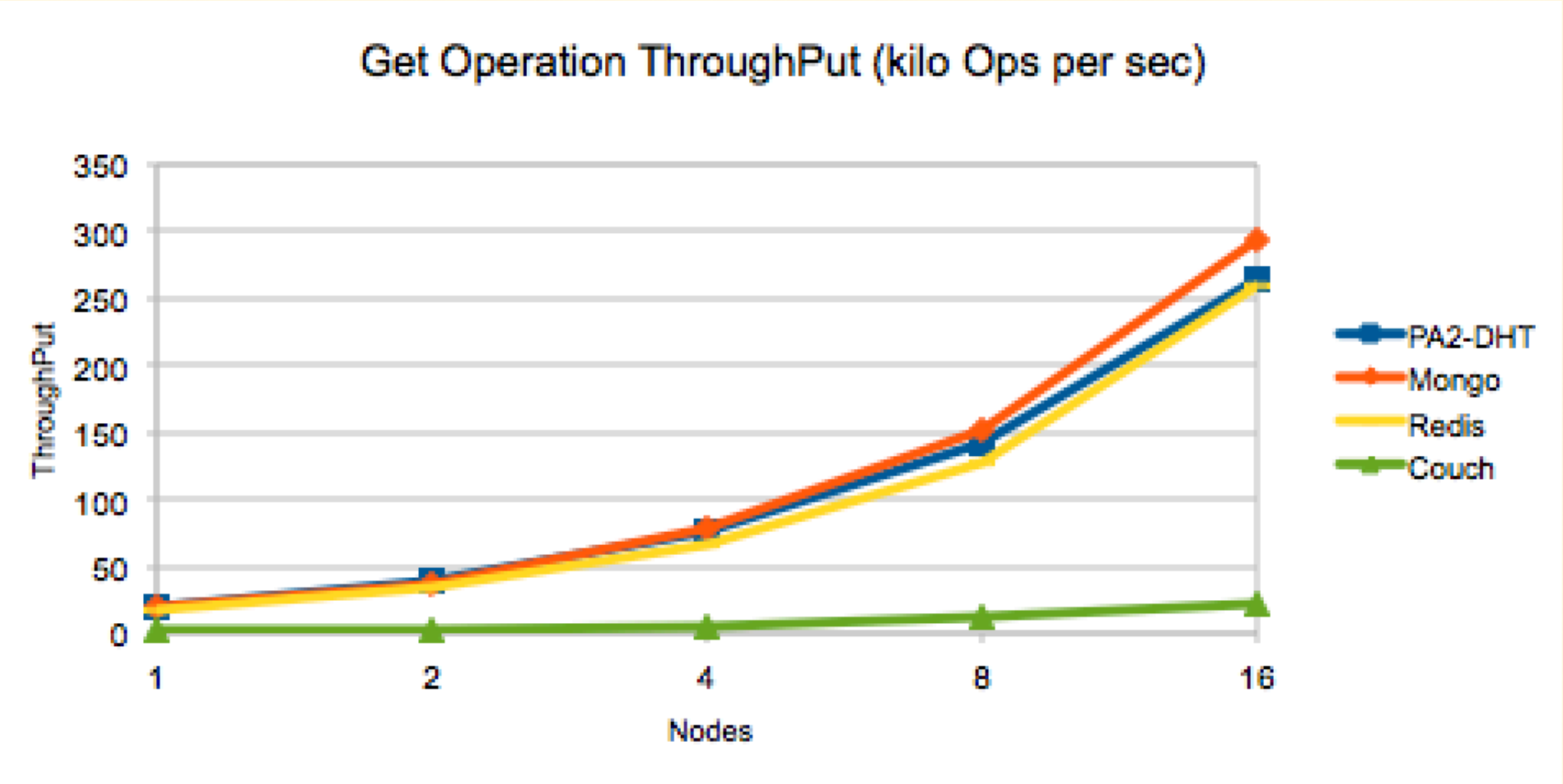
1. Installation shell scripts and Evaluation class for each system is copied to the instances.
2. Shell scripts are executed to install java, ant and databases and made database Up.
3. Then 1,2,4.8.16 instances are executed to capture throughput & latency.

Evaluation - ThroughPut

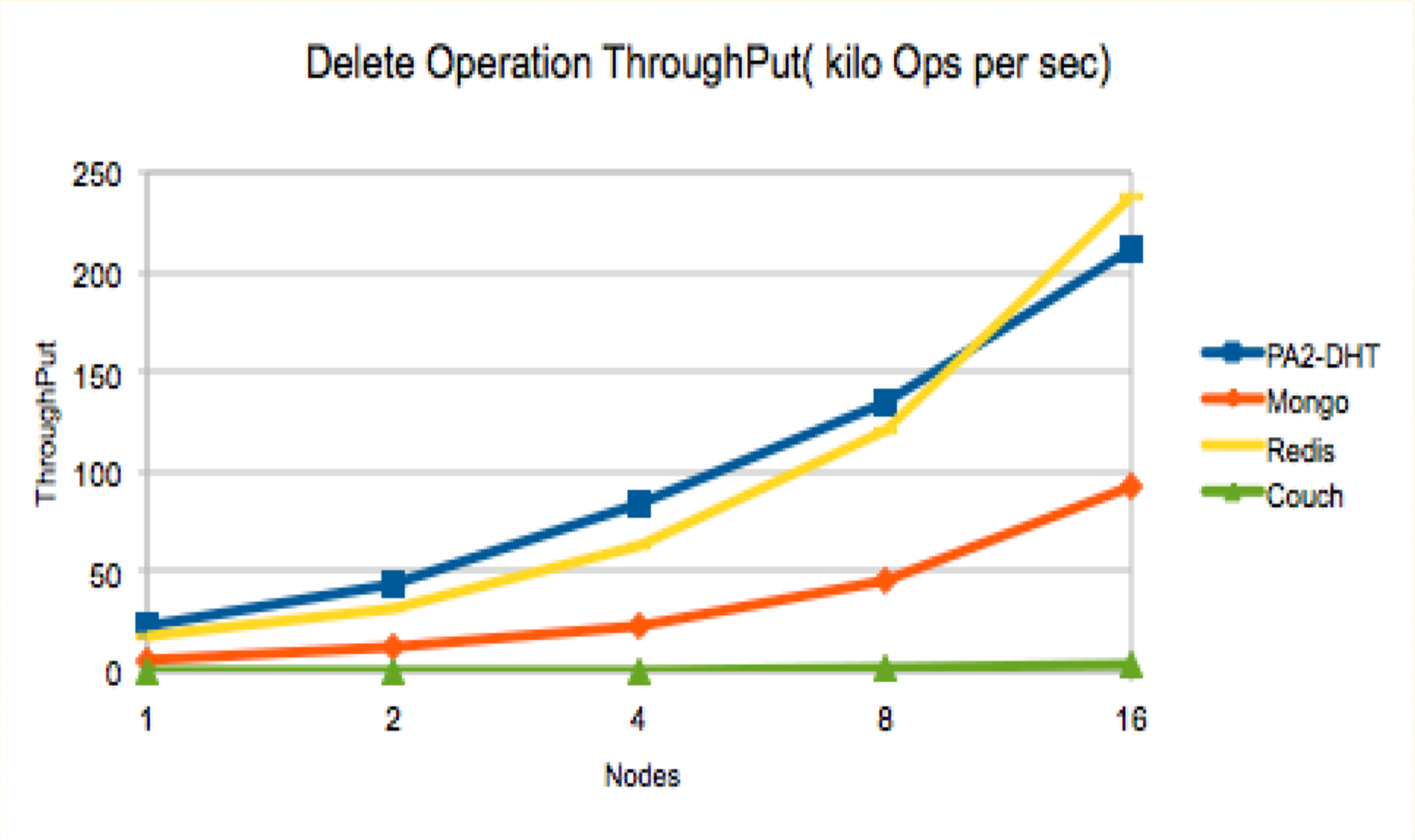
Put Operation ThroughPut



Get Operation ThroughPut



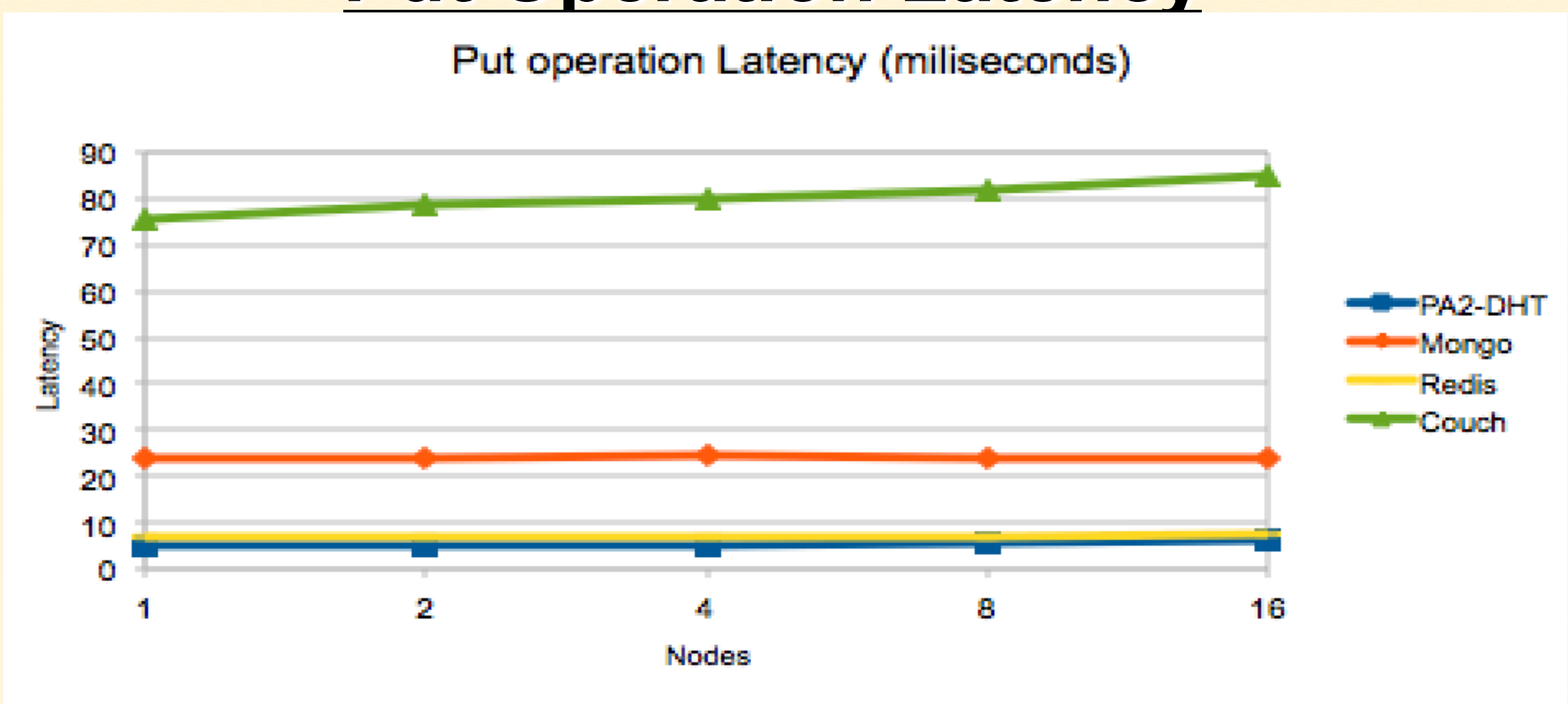
Delete Operation ThroughPut



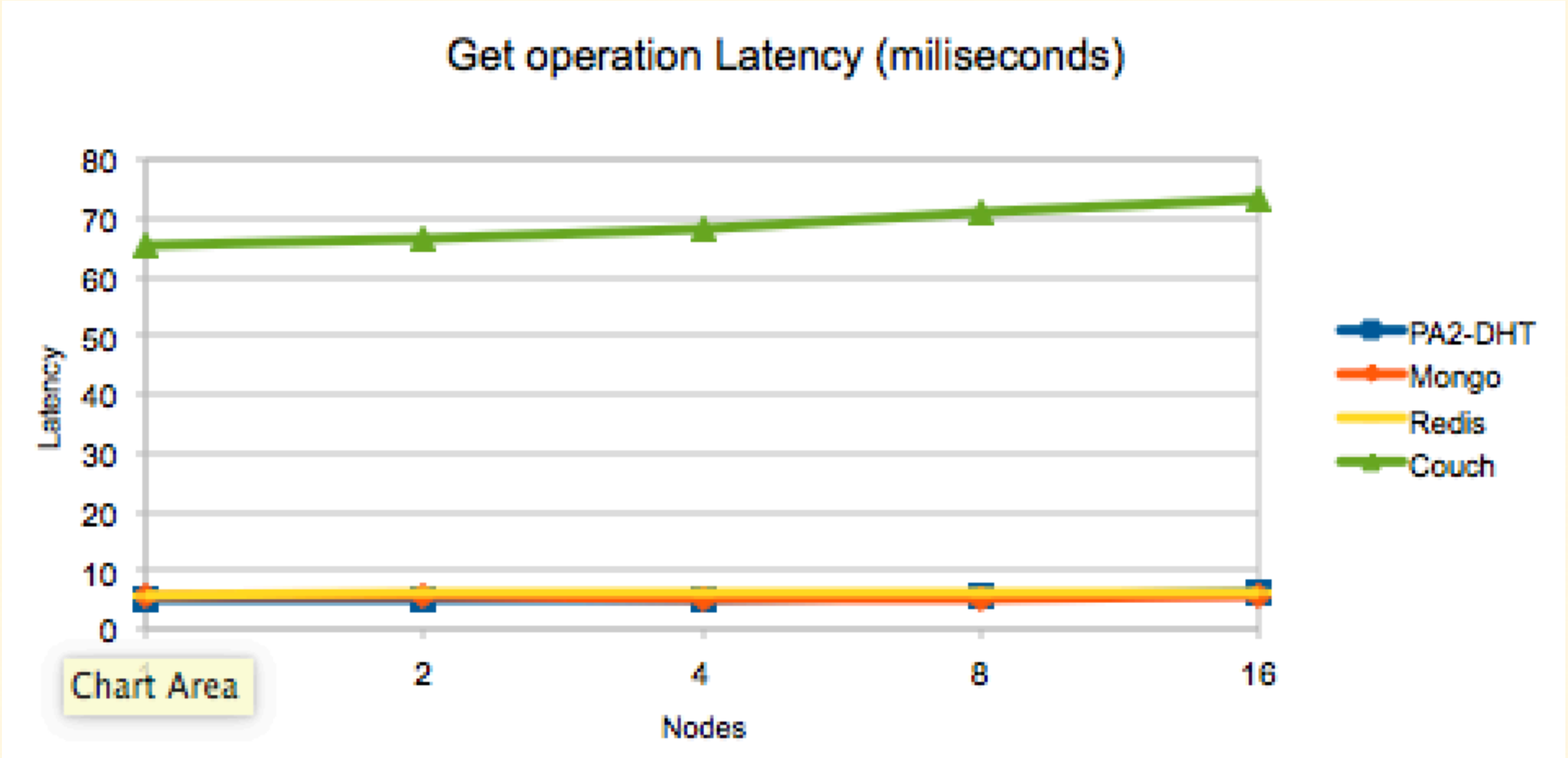
Throughput : The number of operations the System can handle over some period of time, measured in kilo operations per second.

Evaluation - Latency

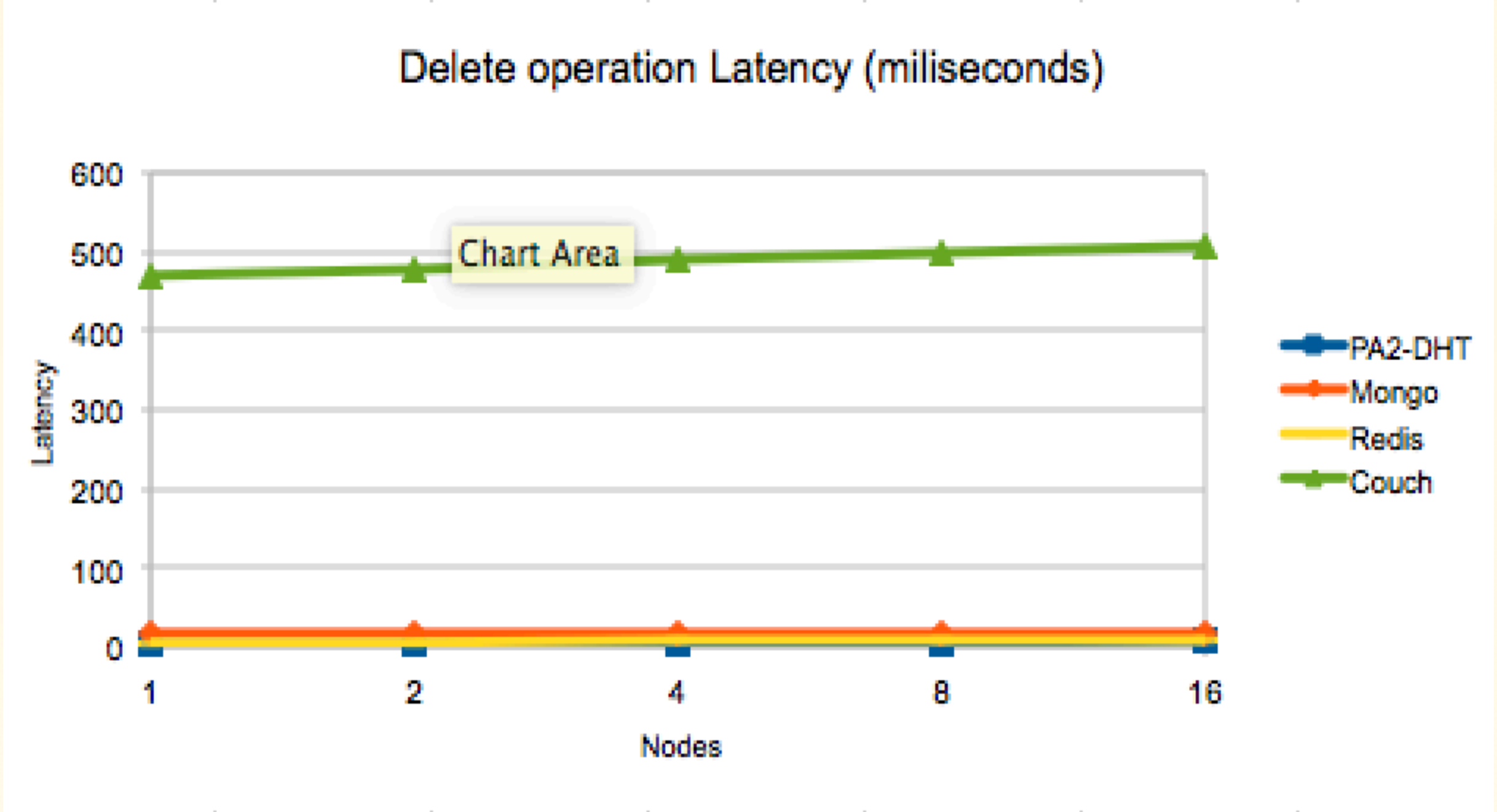
Put Operation Latency



Get Operation Latency



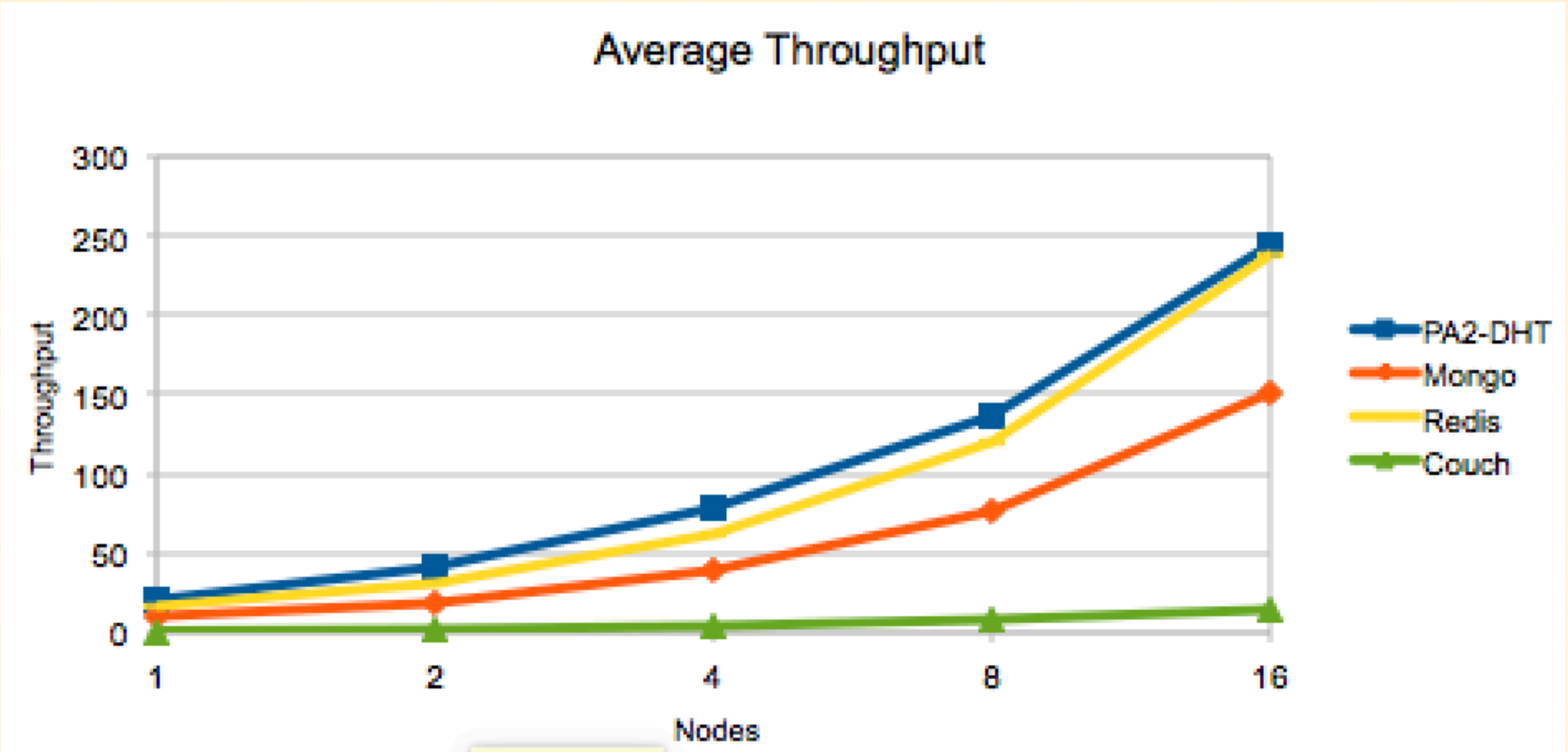
Delete Operation Latency



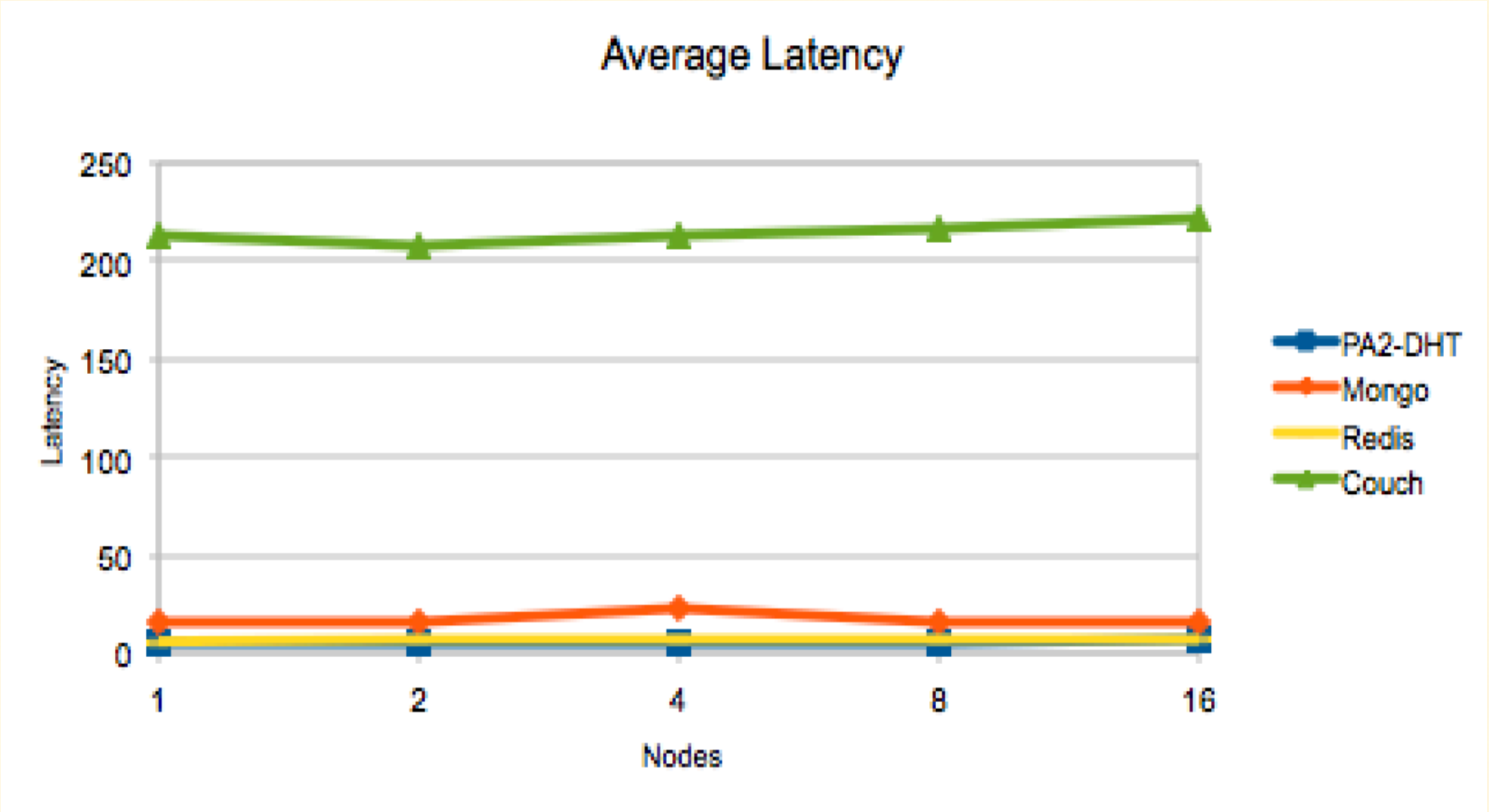
Latency presents the time per operation (insert/lookup/ remove) taken from a request to be submitted from a client to a response to be received by the client, measured in milliseconds (ms).

Average Throughput & Latency

Average Throughput



Average Latency



Conclusion

- Throughput of PA2-DHT system is Highest out of four system for 100k request.
- Throughput of PA2-DHT is high because it is light weight but its performance will degrade for large number of bulky request.
- Latency of PA2-DHT is low as compared to other system.
- For Couch Db throughput is low but latency is high, couch db in case of heavy load will not let the system fail.

References

1. http://datasys.cs.iit.edu/reports/index.html.
2. Stackoverflow
3. Youtube vedios