GreenCHT: A Power-Proportional Replication Scheme for Consistent Hashing based Key Value Storage Systems

Authors: Nannan Zhao, Jiguang Wan, Jun Wang and Changsheng Xie

Presented by: Nannan Zhao
Email: nnzhaocs@hotmail.com

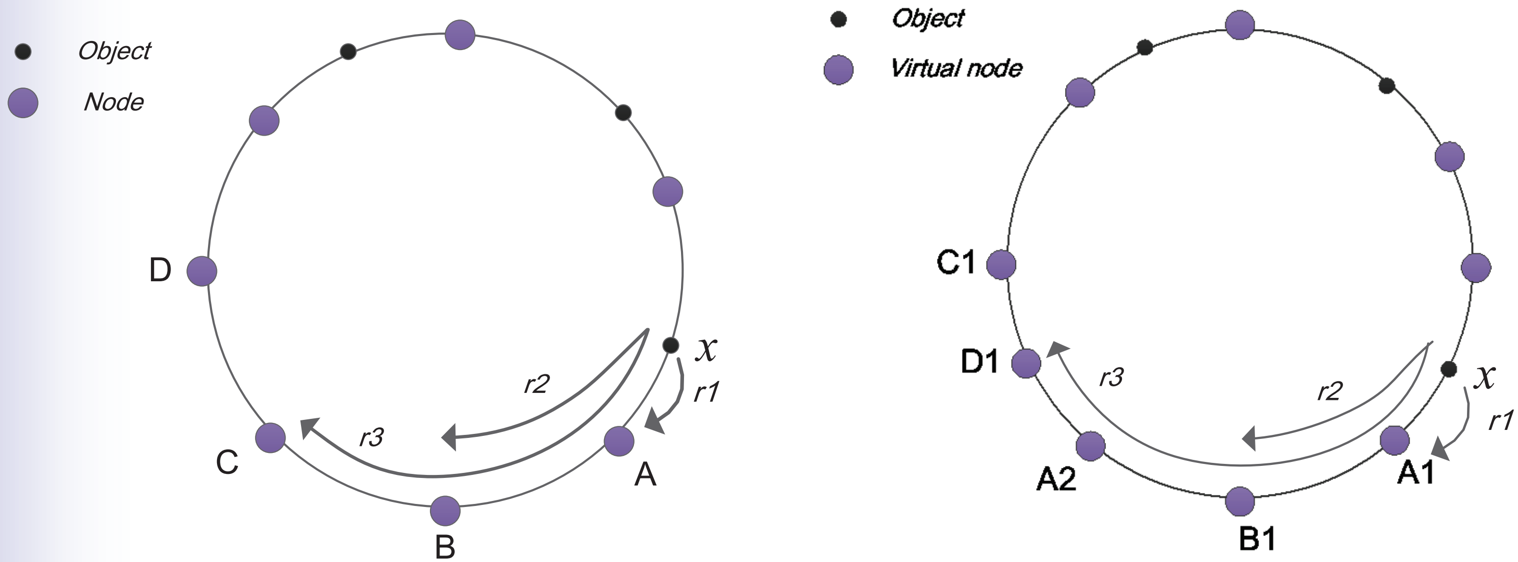
Outline

1. Current energy issues with key value storage systems and server energy consumption
2. Traditional replication scheme for consistent hashing
3. GreenCHT design and implementation
4. Conclusions

Current energy issues with key value storage systems

- **Key value storage systems**
  - Dynamo at Amazon, Cassandra at Facebook, and Voldemort at LinkedIn
    … …
  - Consistent Hash Table (CHT)
    - High Scalability
    - Load balance
    - Simplify the lookup operations
- **The server energy conservation has become a priority**
  - With the increase in the sheer volume of the digital data, storage and server demands are on a rapid increase.
  - Server energy cost constitutes a significant part of a data center ’power bill

Traditional replication under consistent hashing



(a) Without virtual nodes (b) With virtual nodes

Figure 1 Traditional replication under consistent hashing

- The traditional replication strategy prevents subsets of nodes from powering down without violating data availability[1].

[1] D. Harnik, D. Naor, and I. Segal, "Low Power Mode in Cloud Storage Systems,"

GreenCHT design

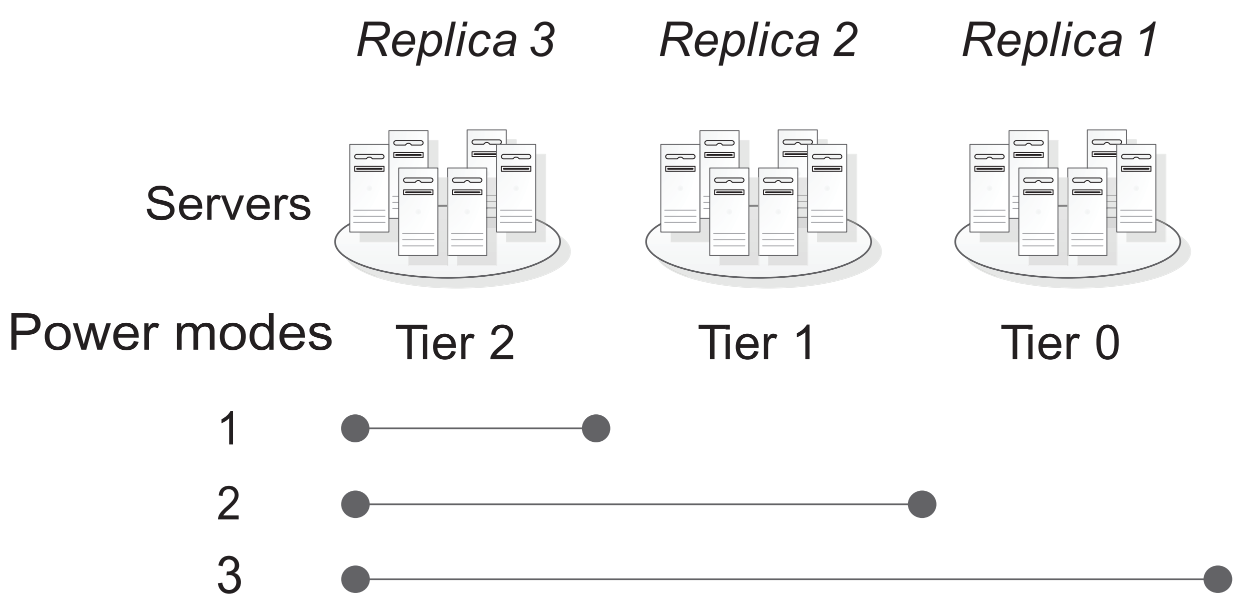


Figure 2 Tiering and power modes with a replication factor of 3

- **Availability**
  - allows $\frac{(R-t)N}{R}$ of the nodes to be powered down
- **Different power modes**
  - sustain different workload levels

Multi-tier replication

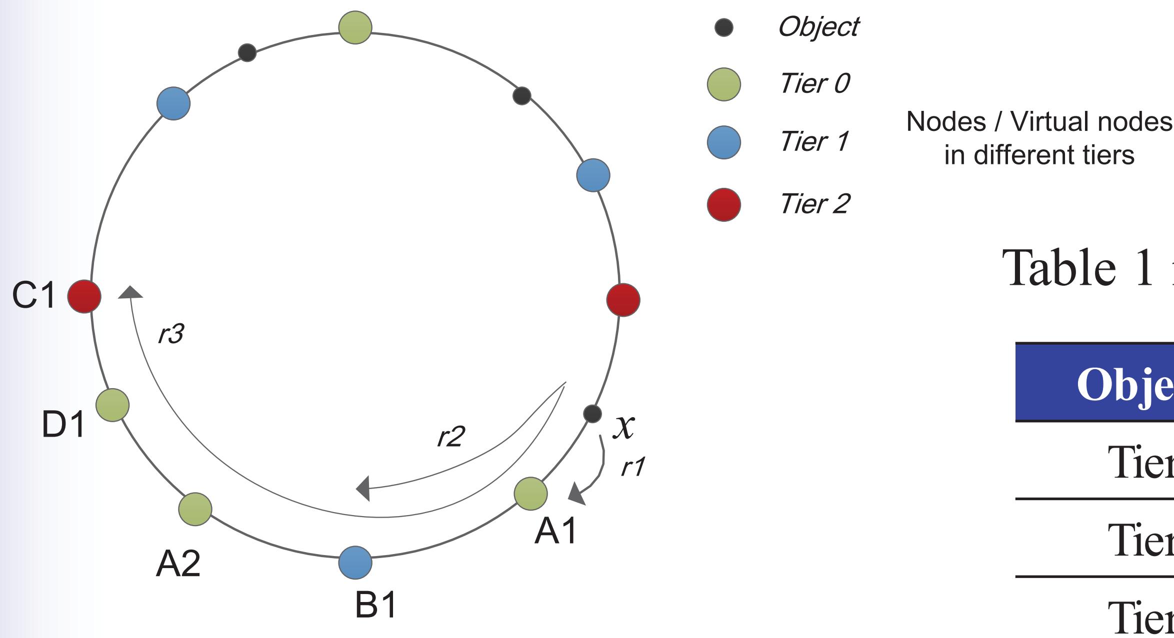


Figure 3

Table 1 multi-tier replication

Object *x*	Successor_1
Tier 0	r1
Tier 1	r2
Tier 2	r3

- **Scalability**
  - when a server *n* joins or leaves the system, certain objects will be migrated between server n and its successor in the same tier.

Log-store

Table 2 log-replicas allocation

Object *x*	Successor_1	Successor_2	Successor_3	… …	Successor_R
Tier 0	r1	-	-	-	-
Tier 1	r2	Log-r1	-	-	-
Tier 2	r3	Log-r1	Log-r2	-	-
… …	… …	… …	… …	… …	… …
Tier (R-1)	rR	Log-r1	Log-r2	… …	Log-r(R-1)

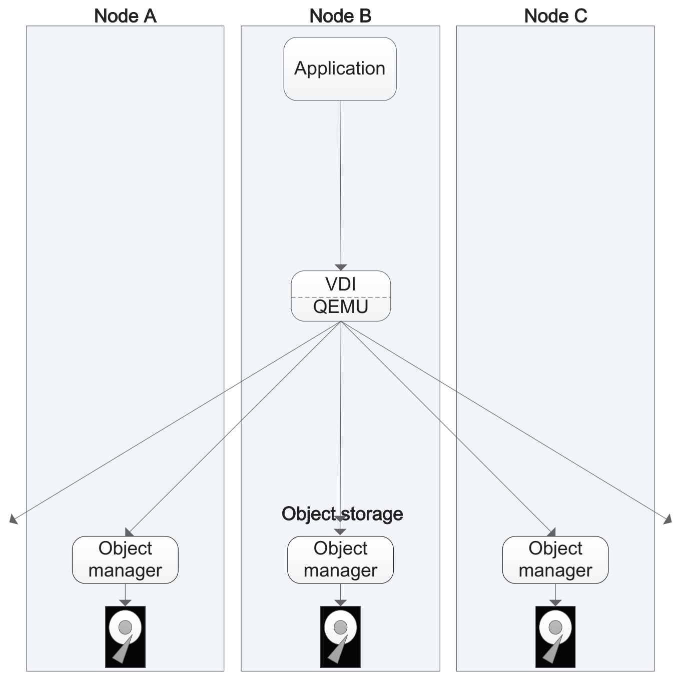
- **Availability and Reliability**
  - All the writes to standby replicas are offloaded to log-store, which exists in active nodes in higher tiers.
- **Parallelism of writes**
  - Replicas and log-replicas are stored in different nodes.
- **Scalability**
  - When a node enters or leaves, certain objects will be migrated between the node and its successor in the same tier. It won't influence other nodes.

Implementation

Power mode scheduler

- Track the load
  - Hour
- Predict the load
  - ARMAX model
- Choose the power mode
  - $P = \left\lceil \frac{L_{predict}}{L_{tier}} \right\rceil$

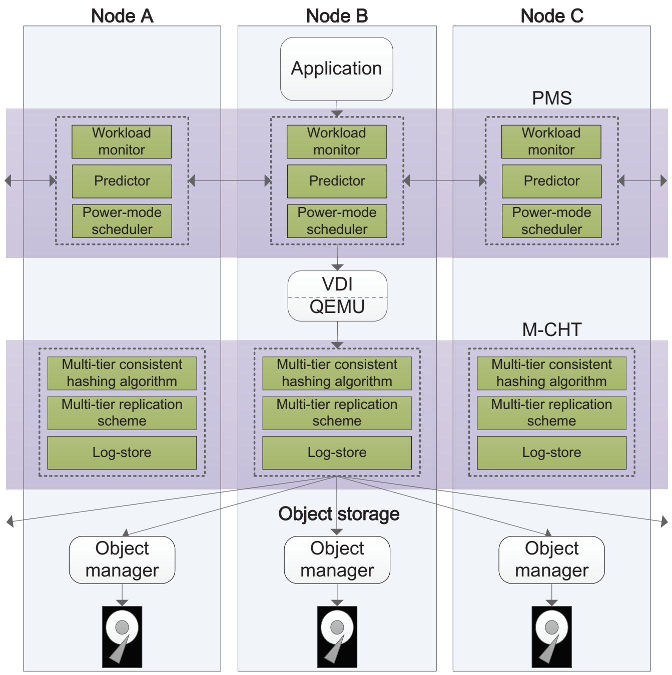
GreenCHT Prototype



GreenCHT was prototyped on Sheepdog, which was chosen for its open source code and its consistent hashing based data distribution and replication mechanism.

Figure 4

GreenCHT Prototype



- We implemented our multi-tier replication scheme on Sheepdog
  - 1. modify its original data distribution and replication algorithm.
  - 2. the power mode scheduler runs in the user space to schedule nodes to be powered-down and powered-up

Figure 5

Evaluation

Power savings

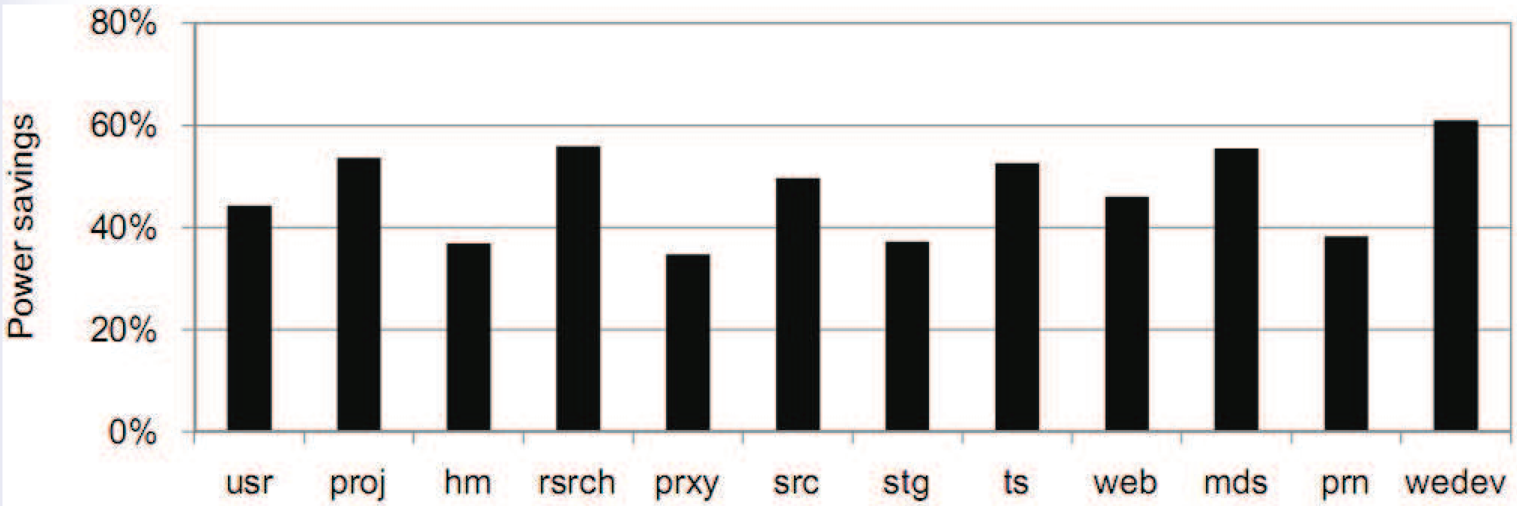


Figure 6

- 35%-61%

Latency

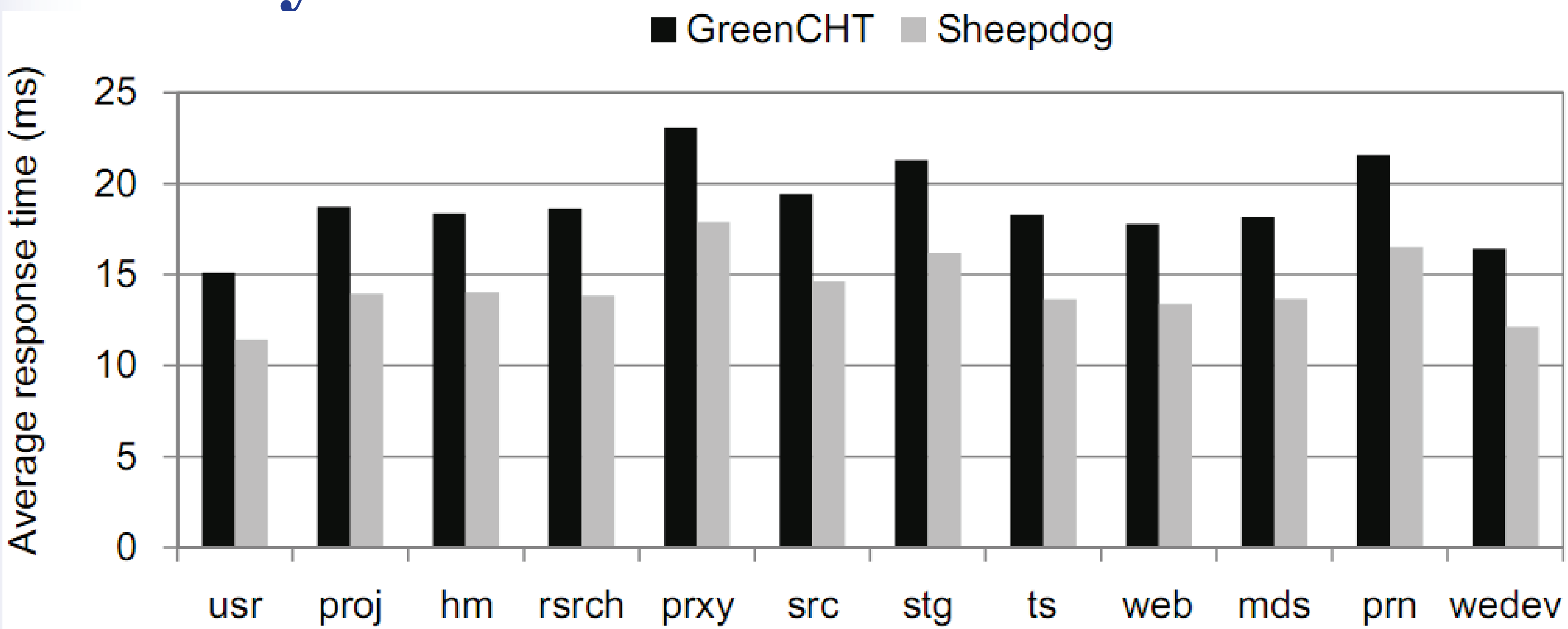


Figure 7

Conclusion

- **Compared with CHT:**
  - GreenCHT saves significant energy
- **Meanwhile, GreenCHT ensures various properties of CHT:**
  - Data availability
  - Scalability
  - Reliability
  - Load balance
  - Maintains a good performance

Thanks

Q&A