Abstract

- Many large-scale data store uses the consistent hashing algorithm or its variants for better scalability and manageability, e.g. Dynamo, Cassandra, Ceph, Sheepdog.
- Lacking support for heterogeneous storage devices and elastic storage.
- Propose of a two-mode consistent hashing algorithm that better support heterogeneous storage devices to offer both performance improvement and balanced capacity utilization.
- Propose of an elastic consistent hashing algorithm to offer agile cluster resizing and selective data re-integration.

Consistent Hashing

- Initially used for load balance in web caching
- Each key (data) generates a node on the ring as well and matches to the next server node in the clockwise direction
- Node for server 1
- Node for server 2
- Node for server 3

Research Problem and Existing Solutions

- Support heterogeneous storage
  - Flash-based SSD and HDD co-exist in many large-scale storage systems
  - SSDs offer better performance but have small capacity
  - HDDs have much more abundant capacity in most large-scale systems
  - Consistent hashing only puts weights on storage servers according to their capacity, which could underutilize the SSDs' performance
  - Existing heterogeneous storage systems are managed via a caching layer or tiered storage solution, which requires an extra layer to manage heterogeneous devices
- Support elastic storage
  - Many large-scale storage systems resize cluster according to workload demand to save power consumption
  - Need an elastic data layout that a full data copy stored on a small set of servers
  - Resizing may incur excessive data migration that degrades performance
- Existing study like SpringFS only works on HDFS-like distributed file systems

Summary

- Consistent hashing algorithm is a promising solution for large-scale data stores
- We propose two variants of consistent hashing to achieve a high performance and power-efficient distributed data store

Acknowledgements

The research studies are supported in part by the National Science Foundation under grant CNS-1162488, CNS-1338078, and CCF-1409946.