CS443 Paper Review
Lei Yang
2005-5-04

****

Title

Venti: a new approach to archival storage

Author

Sean Quinlan and Sean Dorward

Summary

This paper presents Venti, a block-level network storage system intended for archival data, which provides a write-once archival repository that can be shared by multiple client machines and applications.

Most important ideas

The key idea behind Venti, is to identify data blocks by a hash of their contents, also called fingerprint in this paper. Fingerprint is the source for all the obvious benefits of Venti: (1) As blocks are addressed by the fingerprint of their contents, a block cannot be modified without changing its address (write-once behavior); (2) Writes are idempotent, since multiple writes of the same data can be coalesced and do not require additional storage; (3) Without cooperating or coordinating, multiple clients can share the data blocks with Venti server; (3) Inherent integrity checking of data is ensured, since both the client and the server can compute the fingerprint of the data and compare it to the requested fingerprint, when a block is retrieved; and (4) Features like replication, caching, and load balancing are facilitated; because the contents of a particular block are immutable, the problem of data coherency is greatly reduced.

The main challenge of the work, on the other hand, is also brought about by hashing. The design of Venti requires a hash function that could generate a unique fingerprint for every data block that a client may want to store. Venti employs a cryptographic hash function, Sha1, for which it is computationally infeasible to find two distinct inputs that hash to the same value. (To date, there are no known collisions with Sha1.) As to the choice of storage technology, the authors make a good enough argument to use magnetic disks, by comparing the prices and performance of disks and optical storage systems.

Three applications, Vac, physical backup, and usage with Plan 9 file system, are demonstrated to show the effectiveness of Venti. In addition to the development of the Venti prototype, a collection of tools for integrity checking and error recovery were built. The authors also gave some preliminary performance results for read and write operations with the Venti prototype. By using disks, they've shown an access time for archival data that is comparable to non-archival data. However, they also indicated the main problem: the uncached sequential read performance is particularly bad, due to the requirement of random read of the index of the sequential reads. They've pointed it out one possible solution: read-ahead.

Flaws/Questions

Overall, I like the idea in this paper. I think it is quite neat. Although at first, I was a bit concerned about a basic assumption of Venti. That is, the growth in capacity of disks combined with the removal of duplicate blocks and compression of their contents enables a model in which it is not necessary to reclaim space by deleting archival data. But the authors have carefully given statistics in the results part to convince me of the feasibility of the write-once model for archival storage. I also doubted about the collision problem with the hashing function, but they also made a strong argument and convinced me.

Relevance/Potential Future Research

It's worth solve the problem of poor performance in sequential read by implementing the read-ahead technique mentioned in the paper. It would also be interesting to see Venti in use in other systems (file systems), too.