RAMCloud: Scalable storage system in memory

Nanda Kumar Jayakumar, Diego Ongaro, Stephen Rumble, Ryan Stutsman, John Ousterhout, and Mendel Rosenblum

Stanford University

What if applications in datacenters today had access to a storage system that had all the properties we have come to expect of a cloud-based system yet were able to access this storage system with the sort of latency and performance associated with using main-memory? This storage system would be built on top of a cluster of commodity servers in the datacenter and would grow or shrink to accommodate the need of applications elastically. The end-to-end time for an application in the datacenter to access a small storage object, also in the same datacenter, would be no higher than 10μs.

The RAMCloud project at Stanford University is attempting to build this storage system. All data will be stored in the main memory of servers in the RAMCloud cluster. The data will be persisted and made reliable using backing disks on multiple servers in the cluster; servers other than the server with the data in memory. The failure of servers will be handled by the RAMCloud cluster within the recovery-time objective of 1-2 seconds.

Beyond the extremely low RPC latency desired, challenges in building RAMCloud include the distribution and scaling of data in the cluster and the graceful handling of server failures. Providing strong durability guarantees and the data model exposed by RAMCloud are other important considerations.

Uneven evolution of storage capacity versus access times on disks means that disks are set to go the way of tapes and become archival technologies as far as large-scale online storage is concerned.

Large-scale applications limited by these disk-based technologies resort to complicated storage platforms based on layers of caching or accept compromises to their functionality. Memory-based storage systems such as RAMCloud will be required to meet the challenges posed by these large-scale applications in the datacenter.

Results achieved in the project so far include an implementation of the log-structured memory approach to storing data in the server. These logs are then organized into segments and backed by a set of disks on other nodes in the cluster. The plan is to handle failures using very fast recovery instead of storing redundant copies of the data in RAM. This recovery is in the process of being tested and attempts to meet a goal end-to-end recovery time of 1-2 seconds.

Related work includes Google's BigTable and other similar "NoSQL" data stores which have a similar data model but do not have latency targets as low as RAMCloud. Main-memory databases have exploited RAM for several years but do not scale to clusters. Memcached does not provide persistence and, unlike RAMCloud, can only be used as a cache backing other data stores.

More information about the project can be found at http://fiz.stanford.edu:8081/display/ramcloud/Home