MicroFuge: A Middleware Approach to Providing Performance Isolation in Cloud Storage Systems

Akshay K. Singh, Xu Cui, Benjamin Cassell, Bernard Wong and Khuzaima Daudjee
{ak5singh, xcui, becassel, bernard, kdaudjee}@uwaterloo.ca

Abstract

Most cloud providers improve resource utilization by having multiple tenants share the same resources. However, this comes at the cost of reduced isolation between tenants, which can lead to inconsistent and unpredictable performance. This performance variability is a significant impediment for tenants running services with strict latency deadlines. Providing predictable performance is particularly important for cloud storage systems. The storage system is the performance bottleneck for many cloud-based services and therefore often determines their overall performance characteristics.

In this paper, we introduce MicroFuge, a new distributed caching and scheduling middleware that provides performance isolation for cloud storage systems. MicroFuge addresses the performance isolation problem by building an empirically-driven performance model of the underlying storage system based on measured data. Using this model, MicroFuge provides adaptive deadline-aware cache eviction, scheduling and load-balancing policies, minimizing deadline misses. MicroFuge can also perform early rejection of requests that are unlikely to make their deadlines. Using workloads from the YCSB benchmark on an EC2 deployment, we show that adding MicroFuge to the storage stack substantially reduces the deadline miss rate of a distributed storage system. Additionally, MicroFuge is significantly more effective at reducing deadline misses than Memcached, a popular distributed caching middleware.