Database Executive Summary Benchmark of Aerospike vs. Couchbase:

Hybrid Memory vs. a Memory-First Architecture

AEROSPIKE

Executive Summary

In this paper, we present the results of our comparative 12-hour benchmark of Aerospike Server Enterprise Edition 3.12.1 vs. Couchbase Server Enterprise Edition 4.5.1 using a 50/50 read/write workload typical of modern Internet applications.[1]

The benchmark uses the Yahoo! Cloud Serving Benchmark (YCSB) and contains detailed guidelines to reproduce the results using modern bare-metal hardware. For detailed test results you can download the full Database Benchmark of Aerospike vs. Couchbase.

For our most comparable test - with Couchbase having only 250GB of data in a 1 TB benchmark - showed Aerospike serving 3.6 times more transactions per second than Couchbase (410,000 transactions per second vs. 113,000), at an update latency that was 2.9 times lower (0.7 ms vs 2.0 ms). Both databases yielded excellent sub-millisecond read latencies (95th percentile well below 0.5 ms in both cases). Couchbase's requirement for very high cache hit rates required a substantial (4x) data size reduction to meet their operational requirements, which would in turn require far higher hardware costs for a similar size project. Based on these results, we conclude that in this standard YCSB test Aerospike has both a 4x benefit in storage cost, and a simultaneous 3.6x increase in transactions per second, resulting in at best a cumulative 14.4x reduction in hardware cost, or at least 3.6x, depending on use case.

The test results are summarized in the table below:

	Read Throughput	Update Throughput	95th Percentile Read Latency	95th Percentile Update Latency
Aerospike	410,000	410,000	0.3	0.69
Couchbase	113,000	113,000	0.202	1.98
Ratio	3.6 x	3.6 x	0.67 x	2.9 x

Table 1. Summary of Results for the Modified Test Using 99% a Cache Hit Rate Test

Net-net, for the same workload, Aerospike requires fewer servers compared to Couchbase, and hence a much lower total cost of ownership (TCO). Phrased another way, an Aerospike cluster can process 3.6 to 8.0 times more requests than a similarly sized Couchbase cluster.

Introduction

At Aerospike, we've earned a reputation for holding database benchmarks to a high standard. We strongly feel that the only benchmarks worth running and publishing are those that emulate real-world scenarios and provide useful guidance to technical teams as they go through the database selection process; doing any less cheats these teams and results in poor technology choices. Accordingly, in 2015, we wrote a benchmarking manifesto in which we defined guidelines for conducting a benchmark in a manner that is fair and transparent, and yields reproducible results (e.g., our benchmark of Datastax's Cassandra vs. Aerospike).

In this benchmark, we compare Aerospike, a database with a hybrid memory architecture, and Couchbase, a memory-first database. The benchmark is run with high-performance enterprise Flash and allows multiple terabytes per server, which is uneconomical with pure DRAM.

Couchbase implements what they call a "memory-first architecture". All data passes through a caching layer, so Couchbase's performance is very sensitive to cache hit ratios. In his 2014 Couchbase Connect talk titled "Managing a Healthy Couchbase Server Deployment", Justin Michaels, a Couchbase solutions engineer, clearly states that the cache miss ratio for Couchbase needs to be less than 1%; in other words, the database cache hit ratio—a more familiar term—needs to be near 100%. Couchbase's documentation also suggests that the cache miss ratio is a key monitoring variable. It states that "it should ideally be as low as possible; most deployments are under 1% but some accept upwards of 10%. SSD's versus spinning disks have a big effect on what is a reasonable value". Said otherwise, it recommends sizing a cluster for a cache hit rate between 90% and 100%. The documentation further recommends running Couchbase benchmarks using 50% DRAM as well as a Zipfian distribution, which favors hot keys.

Aerospike, in contrast, implements a hybrid memory architecture wherein the index is purely in-memory (not persisted), and data is stored on persistent storage (SSD) and often read directly from disk. This approach works very well for random distributions, or when the distribution is unknown. Flash storage is capable of tens of thousands of reads per second, as long as those reads are in parallel—which the Aerospike hybrid memory index enables. Aerospike's performant use of Flash is also enabled through direct device access, which bypasses the operating system's internal cache and file system layers. Couchbase requires a file system and uses the operating system's cache; both of which decreases performance.

In this benchmark, we exercise both databases with a 50% read/50% write workload similar to transactional workloads seen in fraud detection, real-time bidding, and adtech use cases. We test both a normal and a Zipfian distribution in order to show results with and without high cache hit rates. To implement the workload, we selected the Yahoo! Cloud Serving Benchmark (YCSB), a respected benchmark tool used to compare the relative performance of NoSQL databases. In this paper, we describe the process of the benchmark and its results.

Benchmark Configuration and Process

Our benchmarking process begins by defining the test workload and data. Based on the workload definition, we then select the hardware and define the configuration settings for the databases and the YCSB client. The YCSB benchmark focuses on CRUD operations (Create, Read, Update, and Delete) using primary key access. This pattern is used for user profile storage in adtech, trade status in financial services, session management, fraud analytics, and a wide variety of other use cases. While Aerospike and Couchbase both provide a variety of other features regarding indexing and searching, it is the core CRUD operations that form the bulk of Internet-scale workloads.

Leveraging the scale of Flash is critical to a database's ability to process next-generation operational workloads. While Aerospike's performance rises with a DRAM configuration, the operational capabilities of today's Flash economically enables very large (above 100 TB) "in-memory" datasets, while providing data persistence and very high performance. Aerospike's internal cache, which focuses on recently written records, would not be effective for this form of highly random workload.

In this test, we used Samsung's PM1725 NVMe drives. Designed by Samsung for enterprise workloads and environments, each drive supports 1.6 TB of storage. Our testing shows that these high-endurance drives requires no extra overprovisioning to achieve substantial performance. They have performance characteristics similar to other vendors' high-endurance, enterprise-class offerings that were available in 2016. Storage performance can be further improved by adding more devices per chassis.

Workload

This benchmark is designed to emulate a transactional workload, characterized by data entry (writes/updates) and retrieval (reads); we therefore selected a 50% read / 50% write benchmark workload. Based on Couchbase's definition of their database as a key-value store, for which a 50/50 read/write workload is common, and their use of this same workload in their own benchmarks, we felt that using this workload for Couchbase was reasonable. This 50/50 read/write workload is particularly useful due to its ability to show inefficiencies in locking, concurrency, and parallelism, as both reads and writes are executed to the same data store.

Duration

While there exists many benchmarks for operational and NoSQL databases, the vast majority are conducted for short test periods—normally less than an hour. In truth, Couchbase's Avalon benchmark only ran for 32 minutes. In our Aerospike Database Manifesto, published in 2016, we pointed out the problems of such small, short-term tests, and proposed instead that benchmarks be performed over an extended period of time, as this provides a more realistic characterization of database performance over time. In this benchmark, we compared the Aerospike Server Enterprise Edition 3.12.1 to the Couchbase Server Enterprise Edition 4.5.1 over a 12-hour period, during which we ran our tests continuously. Our goal was not just to understand initial peak performance, but also to comprehend the performance of the system over time as various periodic management processes ran, such as disk defragmentation and compaction. These periodic processes can and do affect the overall throughput and latency of operations.

Number of Objects

Factors such as quantity, object size, or data distribution can hide or expose database behaviors in benchmark results—for instance, small data sets can hide flaws. In our test, since effective use of storage is a key part of an efficient system, we used an amount of keys sufficiently large (1 billion) to ensure that there was more data than would fit purely in DRAM. This made sure that the benchmark exhibited the efficiency of the database persistence. In order to comprehend the extent of Couchbase's dependence on high cache hit ratios for optimal performance, we also conducted a modified test with 215M objects—less than a quarter of the object size of the initial test. Reducing the number of objects for Couchbase helped define a working key set that resided in memory to create a high cache hit ratio.

Data Distribution Type

As the YCSB executes a workload, the keys used in an operation are selected based on a data distribution type. We find that Internet workloads have some cache-friendly characteristics, and thus support in-database caching through Aerospike's recently-written cache. However, the Zipfian distribution as typically configured in YCSB results in the hottest objects being very, very hot - far hotter than any we have seen with Aerospike deployments. While use of objects is not uniform either, the uniform distribution better models Internet transactional workloads. We adopted the uniform distribution for the primary test with Aerospike and Couchbase. This distribution type causes a much higher percentage of operations to access storage; any inefficiencies in the code stack or layers of an architecture are more clearly exposed. In the modified test for Couchbase, however, in order to provide a cache hit ratio of 99%, we used the Zipfian distribution to create a working key set that resided completely in memory.

AEROSPIKE

Consistency

Replication provides for durability and availability of internet-facing applications that our customers demand. This benchmark uses a replication factor of two to create the durability seen in the typical use cases. Adding replication to the cluster can create potential consistency problems. By configuring the write policy to respond only after all replicas have been updated, the databases will insure that the data is consistent between masters and replicas. Although this consistency level deviates from the default eventual consistency in Couchbase, it is a key requirement and the typical configuration of Aerospike users. Therefore, testing both databases under these conditions creates the most realistic comparison.

Tests Conducted

In our primary test, we compared Aerospike and Couchbase. Based on our results, we then decided to run a second test (the "modified test") on Couchbase only. Our test parameters are shown below.

Workload definition for the primary test (Aerospike and Couchbase):

- 1B unique records
- Object size: 10 fields of 100 bytes each (total of 1,000 bytes) per record
- Distribution: Uniform
- Replication factor of 2
- Strong consistency
- Writes: Configured for replace

Workload definition for the modified test (Couchbase only):

- 215M unique records (less than a quarter of the data size in the primary test)
- Object size: 10 fields of 100 bytes each (total of 1,000 bytes) per record
- Distribution: Zipfian
- Replication factor of 2
- Strong consistency

- Writes: Configured for replace

These test parameters define the basis for selecting hardware and determining a testing process for the benchmark. The following table summarizes the hardware used for both the client and servers:

	Database Servers	Client Servers
Server Model	PowerEdge R730xd	PowerEdge R730xd
CPU(s)	56	56
Memory	256G	256G
Network	Intel X710 for 10GbE	Intel X710 for 10GbE
Storage	Samsung PM1725 NVMe 1.6 TB SSD	N/A
OS	CentOS Linux release 7.2.1511	CentOS Linux release 7.2.1511
Count	3	4

Table 2. Summary of Hardware Used in the Benchmark

Methodology

The rest of the benchmark definition consists of the testing process. Below are the steps that we followed during the testing process:

1. Load supporting monitoring tools and applications (e.g. NTP, iostat, dstat, htop).
2. Install and configure a three-node cluster with recommended settings. Verify the cluster is operating normally.
3. Configure each database to get the best performance with the given hardware. Run short test runs (10 minutes) to validate changes in configuration. See the references below for the recommended settings:

   Aerospike

   Configuring Aerospike

   Couchbase

   Couchbase Documentation

   Couchbase Tuning Presentation

AEROSPIKE

4. Load data set of 1B records (215M records for the modified test) into the test cluster's storage system of two locally attached Samsung PM1725 NMVe drives per server.
5. Clear the OS filesystem cache (this was only done for Couchbase, as Aerospike does not use a filesystem).
6. Run tests for 12 hours.
7. Collect data, generate graphs, and analyze results.
8. For the modified test, repeat steps 4-7 for Couchbase only.

Note: We allowed both databases to warm up for two hours in order to reach a steady state. So, while we conducted the tests for 12 hours, the graphs only display test results from the second hour to the twelfth hour.

For detailed test results you can download the full Database Benchmark of Aerospike vs. Couchbase.

About Aerospike

Aerospike enterprises overcome seemingly impossible data bottlenecks to compete and win with a fraction of the infrastructure complexity and cost of legacy NoSQL databases. Aerospike's patented Hybrid Memory Architecture™ delivers an unbreakable competitive advantage by unlocking the full potential of modern hardware, delivering previously unimaginable value from vast amounts of data at the edge, to the core and in the cloud. Aerospike empowers customers to instantly fight fraud; dramatically increase shopping cart size; deploy global digital payment networks; and deliver instant, one-to-one personalization for millions of customers. Aerospike customers include Airtel, Banca d'Italia, Nielsen, PayPal, Snap, Verizon Media and Wayfair. The company is headquartered in Mountain View, Calif., with additional locations in London; Bengaluru, India; and Tel Aviv, Israel.

AEROSPIKE

2525 E. Charleston Road, Suite 201
Mountain View, CA 94043
Tel: 408 462 AERO (2376)
www.aerospike.com