INITIAL COMPARISON

Aerospike vs. ScyllaDB—An Initial Comparison

As a leading NoSQL database, Aerospike delivers predictable performance at scale, superior uptime, and high availability at the lowest total cost of ownership (TCO) compared to first-generation NoSQL and relational databases. To maintain our competitive advantages, we routinely perform benchmarks against other DBMSs. Our most recent Cassandra benchmark found that Aerospike performs 14x better than Cassandra at 1/5th of the cost. When ScyllaDB recently released a new version with throughput and latency comparisons to Cassandra, we decided to take a quick look at ScyllaDB and gauge its performance in the same test scenario.

In general, we found that ScyllaDB's performance claims are based on data that has very high cache hit rates—that is, a vast majority of the data can be stored in DRAM. This test load represents a user profile or predictive analytics use case in which the dataset is too large to efficiently fit in DRAM. When used with Flash storage, ScyllaDB's performance is closer to Cassandra's.

Table 1 summarizes the high-level results, and the raw numbers are remarkable. Aerospike demonstrates extraordinary speed at scale, performing 9x better than ScyllaDB.

Summary of Results

	Read Throughput (Transactions per Second)	**Update Throughput** (Transactions per Second)	**95th Percentile Read Latency** (Milliseconds)	**95th Percentile Update Latency** (Milliseconds)
Aerospike	**125,000**	**125,000**	**2.3**	**4.0**
ScyllaDB	13,200	13,200	31	8.0
Ratio	**9x better**	**9x better**	**13.5x better**	**2x better**

Table 1

The ratio row of Table 1 shows the relative performance of Aerospike against ScyllaDB, underscoring Aerospike's clear performance advantage. Beyond a 9x better throughput, Aerospike demonstrated 95th percentile read latencies that were 13x lower than ScyllaDB's and update latencies that were 2x lower. The performance of ScyllaDB with storage is not markedly better than Cassandra's performance in the original benchmark.

Using the benchmark results, we also calculated the TCO of running both Aerospike and ScyllaDB on bare metal servers, achieving 1M operations per second. Table 2 shows that Aerospike is 9x cheaper than ScyllaDB with a significantly lower average latency.

To determine TCO, we priced the equipment used, the cost of utilities, and the labor costs. The bare metal server for this example is a customized Dell PowerEdge R530, and we assumed an engineer to server ratio of 1:75, with an average annual DevOps engineer salary of $130,000.

Total Cost of Ownership

	Operations per Second	Average Latency (Milliseconds)	# of Servers Required	Annual TCO
Aerospike	**1,000,000**	**3.15**	**12**	**$108,448**
ScyllaDB	1,000,000	19.5	108	$976,032

Table 2

Running the Comparison

For our initial Cassandra benchmark, we used YCSB to generate a 50% read, 50% write workload with 400M 1K objects against both Aerospike and Apache Cassandra. The tests ran for 12 hours on a three node cluster of data center class servers. This comparison presents the results of running the same benchmark against ScyllaDB. You can find more details of the benchmark environment in the blog.

ScyllaDB represents itself as a self-tuning, zero configuration database, so the setup process for ScyllaDB is different than the setup process for Cassandra. The tuning process is automated by the database and setup scripts, which take care of most of the ScyllaDB recommended configuration. The instructions for setup and script execution were followed as outlined in the administration documentation on the ScyllaDB website.

To run our ScyllaDB comparison, we had to make a few changes to our published Cassandra benchmark configuration. The original benchmark was run with Centos 6.7, but ScyllaDB is only compatible with Centos 7.2. Thus, we upgraded Linux to Centos 7.2. ScyllaDB also replaces the TCP stack, which affects how packets are routed to CPUs. This appears to prevent efficient use of ScyllaDB with only one client. For this comparison, we ran 12 client instances across two separate client servers for ScyllaDB, whereas Aerospike, using the standard Linux TCP stack, required only one YCSB instance on a single client server.

After installing ScyllaDB and running the setup scripts, we followed the test methodology detailed in our earlier blog to execute the benchmark, making sure that the compactions from the load process were complete and the cache was cleared before executing the workload.

In the process of running the multi-client tests, we experienced a few problems with ScyllaDB. On several test runs, the servers dropped connections, causing the clients to time out. Only by reducing the number of client instances and threads did the timeouts stop. In one test run, the compactions were not being completed in a timely manner, causing the data drive to fill up and the servers to stop, resulting in data corruption.

The Results

Load Time (Insert Rate)

During our testing, we experienced significantly faster load times for Aerospike than for ScyllaDB. To load the data to Aerospike, we used a single instance of YCSB with 200 threads, whereas we used 12 instances of 20 threads each and two separate client servers to load the data to ScyllaDB. Our findings are summarized in Table 3.

Insert Rate

	Records Inserted	Load Time (in min)	Compaction Time (in min)	Total Prep Time (in min)	Insert Rate (ktps)
Aerospike	**400 M**	**28**	**0**	**28**	**240**
ScyllaDB	400 M	101	35	136	66
Ratio		**3.6x**	**35x**	**4.86x**	**3.64x**

Table 3

Aerospike's insert rate was measured at 3.6x better than ScyllaDB's. We took this measurement during YCSB's load phase, which is not included as part of the workload phase and simulates a 100% write workload. When you take compaction into account, the performance difference between the two databases increases to 4.86x.

Read Results (Throughput and Latency)

While most benchmarks are designed to produce the highest possible numbers, ours is designed to stress the database with heavy workloads. During the workload phase of the benchmark, YCSB was used to apply a mixed 50% read and 50% write load against the databases.

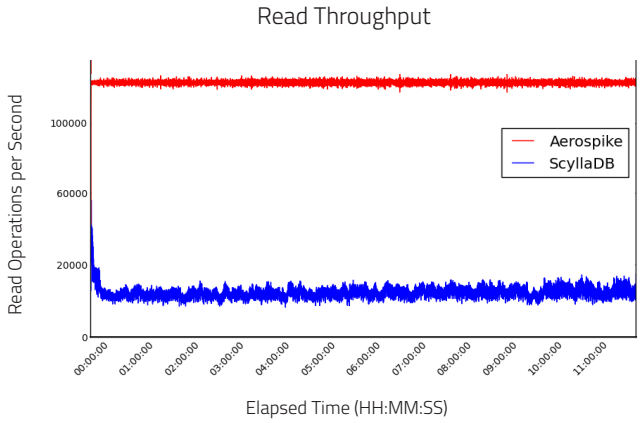

Figure 1

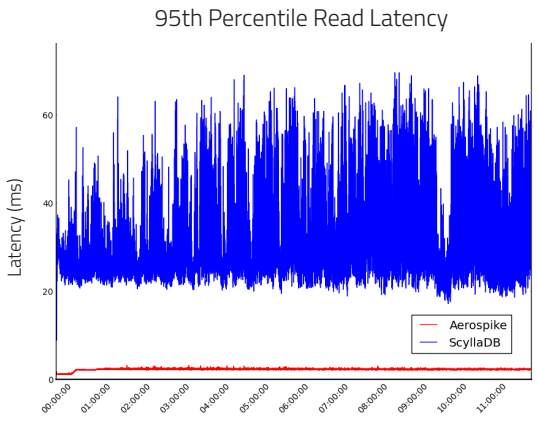

Figure 2

Aerospike demonstrated 125K TPS versus ScyllaDB's 13.25 TPS for the read portion of the workload. Not only is the performance 9x better, but Aerospike demonstrates less variance in throughput than ScyllaDB.

In most operational use cases, predictable read latency is critical. Aerospike has 13.6x lower latency with very predictable, less variant response times, as can be seen in Figure 2. Contrastingly, ScyllaDB produced more variant response times, and these larger variances create greater uncertainty in the read latency. Aerospike generates a narrow range of variance that will produce a tighter SLA for your application.

Update Results: Throughput and Latency

During the update part of the workload, Aerospike demonstrated 125K TPS versus ScyllaDB's 13.2 TPS. Aerospike produced an average 4ms 95th percentile latency compared to ScyllaDB's average 95th percentile latency of 8ms, making Aerospike more predictable than ScyllaDB.

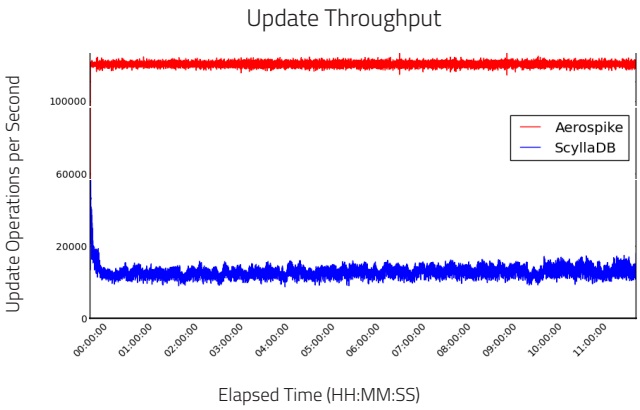

Figure 3

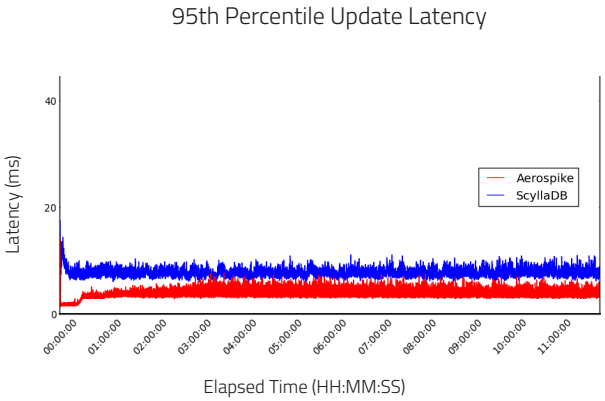

Figure 4

What Does This Mean for You?

Our follow up benchmark comparison on ScyllaDB shows Aerospike's clear performance advantages with 9x greater throughput, 13.5x lower read latency, 2x lower update latency, and 3.6x greater insert throughput. But what does this mean for your business? Aerospike is predictably performant and more easily scalable than ScyllaDB at a 9x lower TCO. Aerospike requires fewer clusters, and these clusters can process more transactions at lower latencies than competitor clusters, reducing the complexity and costs of operationalizing your applications.

AEROSPIKE

2525 E. Charleston Road
Suite 201

Tel: 408. 462. AERO (2376)
www.aerospike.com

Aerospike is the high-performance NoSQL database that delivers speed at scale. Aerospike is purpose-built for the real-time transactional workloads that support mission-critical applications. These workloads have the mandate to deliver informed and immediate decisions for verticals like Financial Services, AdTech, and eCommerce. The unique combination of speed, scale, and reliability can deliver at least 5x performance at 1/5th the cost when compared to most other databases.