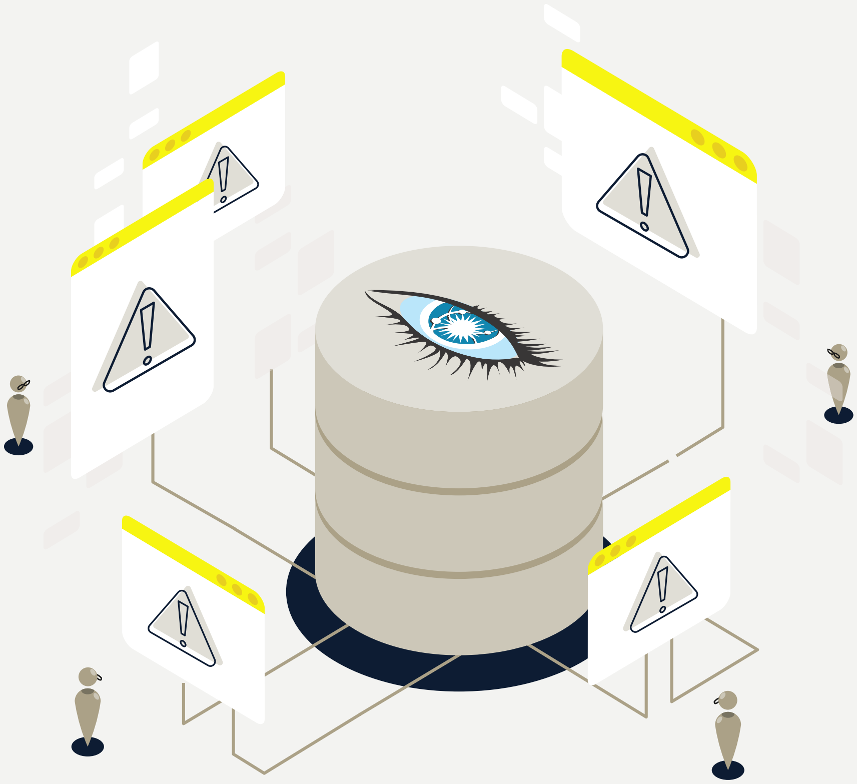
Aerospike



WHITE PAPER

Five signs you've outgrown Cassandra

Introduction

Many organizations choose Cassandra for its promise of scalability and distributed architecture. However, as workloads and data volumes grow, they experience very large amounts of hardware usage and commensurate staffing but inconsistent database performance and lots of maintenance. This can lead to service-level agreement (SLA) violations, high ownership costs, operational instability, and, ultimately, customer dissatisfaction.

That's when Aerospike can help. Aerospike's speed, high availability, operational efficiency, strong data consistency, and self-management features have prompted many Cassandra users to switch.

What are five signs you may have outgrown Cassandra?

1

You're worried about server sprawl and high TCO

- Are growing business demands driving ever-larger clusters?
- Can your budget accomodate the resulting cost increases?

2

Peak loads are disrupting your SLAs

- Are you missing application SLAs? Is this impacting your bottom line?
- Are you provisioning more hardware or caches to meet your latency and throughput requirements?

3

You need fast, predictable performance with high availability

- Are high tail latencies troubling you?
- Has your staff struggled to keep Cassandra running smoothly and your data highly available?

4

Your operations team keeps growing (and so do your costs)

- Are you fighting garbage collection, tombstones, and Java Virtual Machine (JVM) tuning?
- Do you have to re-tune for each new workload, major software release, or hardware refresh?
- Is configuring and tuning for nine consistency levels causing issues?

5

You need more flexible data modeling and deployment options

- Are you wanting to deliver key-value, document, graph, and vector search workloads on a single platform?
- Would you like a single platform for caching as well as persisting data to streamline your infrastructure to lower costs?

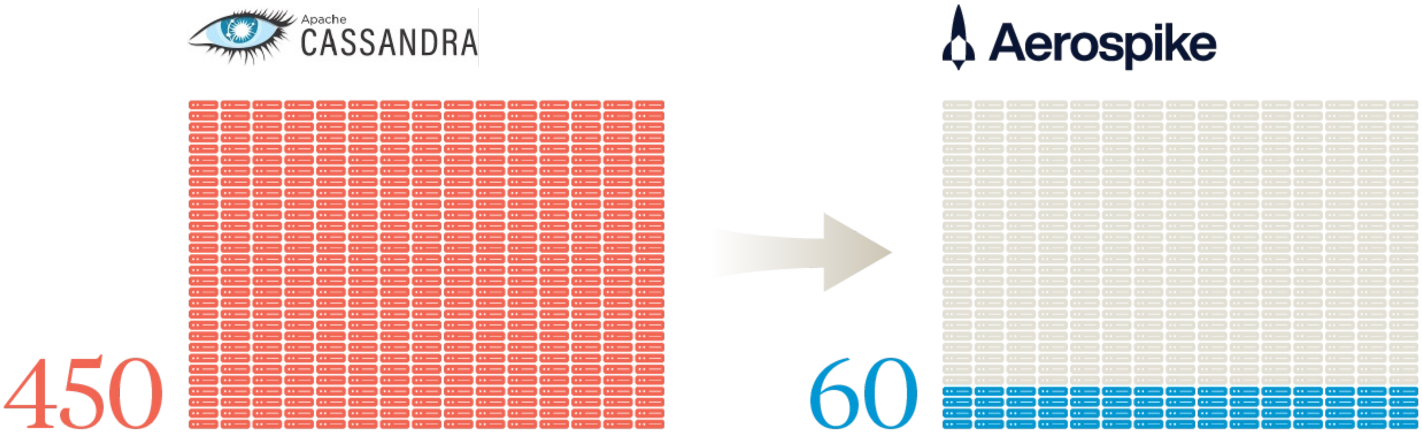
1. You're worried about server sprawl and high TCO

Cassandra is known for its ability to scale horizontally, though it may not utilize resources as efficiently. That's why many production users need large clusters. Vendors that price by the node love that; they simply encourage you to add more nodes whenever you hit a resource limit — CPU, IOPs, RAM for the JVM heap, etc. However, the larger the cluster, the more components you'll have with more hardware failures, not to mention more labor to monitor, tune, and manage the environment.

Aerospike's efficient use of computing resources enables it to reduce TCO to a fraction of Cassandra's. Dynamic Cluster Management, a Smart Client layer, massive parallelization to fully exploit multi-core CPUs, and a patented Hybrid Memory Architecture™ contribute to Aerospike's exceptional scalability and cost efficiency.

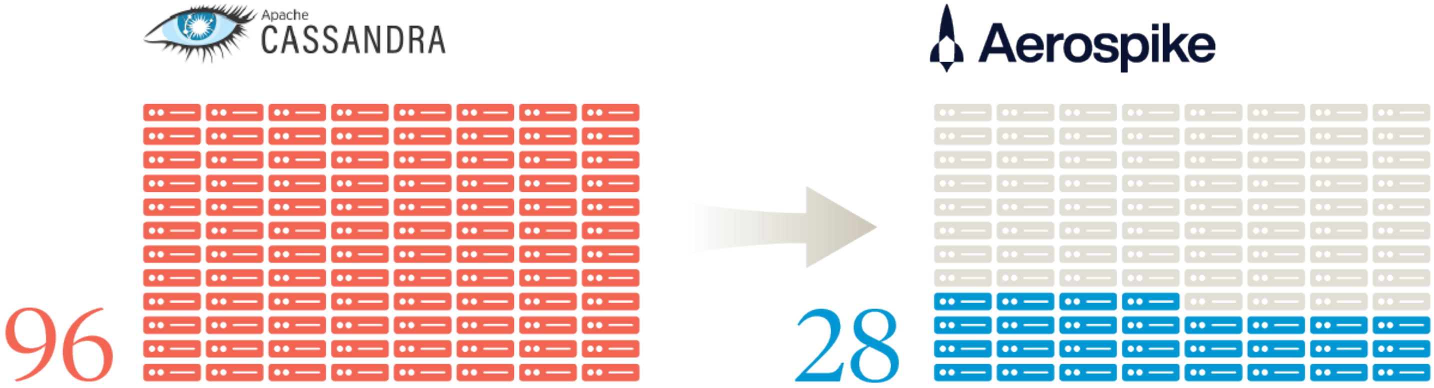
Unlike Cassandra, which was originally designed for commodity servers, Aerospike exploits modern hardware. For example, Aerospike data structures are partitioned with fine-grained locks to avoid memory contention for more efficient use of multi-core CPUs. In addition, Aerospike treats NVMe, Flash, and SSDs as raw block devices to reduce I/O, avoiding overhead from standard storage drivers and file systems that drive up data access latencies. Such features enable Aerospike clusters to manage more aggressive workloads and higher data volumes with fewer nodes than the equivalent Cassandra cluster, reducing operational complexity and TCO.

Customer experiences demonstrate the savings Aerospike can offer. TransUnion migrated from Cassandra to Aerospike and reduced TCO by 68% over three years. It cut the number of deployed servers from 450 to 60, improved performance 100x at the 99th percentile, and executed business processes in 1/10th of the time.



TransUnion used 87% fewer nodes when switching to Aerospike.

Similarly, LexisNexis Risk Solutions (ThreatMetrix) moved from 96 Cassandra nodes to 28 Aerospike nodes, reduced latency from 120 to 30 milliseconds while expecting to save $3 million over three years, and improved core performance of a critical business application. Those are just two examples.



LexisNexis used 67% fewer nodes when switching to Aerospike.

2. Peak loads are disrupting your SLAs

Achieving strong, consistent performance can be elusive when running mixed workloads on Cassandra. Cassandra, written in Java with a log-structured merge (LSM) tree storage design, requires users to tune JVMs, manage garbage collection, and optimize compactions. Mistakes can cause system instability and unpredictable latencies. Furthermore, any Cassandra node can accept any read or write request even if it doesn't manage the data involved, so extra network traffic is unavoidable.

By contrast, Aerospike delivers predictable, ultra-fast data access speeds for mixed read/write workloads at scale. Written in C by developers with deep expertise in database, storage, and networking technologies, Aerospike minimizes data access latencies without requiring extensive operator effort. Aerospike users don't have to contend with Java heap tuning, multiple garbage collection strategies, and other complex performance optimizations required by Cassandra. In addition, Aerospike's Smart Client™ layer maintains a dynamic partition map that identifies the primary node for each partition. This enables the client layer to route the read or write request directly to the correct node without additional network hops for all data sizes, minimizing latencies and providing consistently fast access to user data.

Scalability and elasticity are essential features for any operational data platform to handle peak workloads and seasonal variations. In many respects, Cassandra falls short. Cassandra enables users to scale horizontally and takes pride in referencing users with tens of thousands of nodes. Aerospike fully supports both horizontal and vertical scaling (i.e., scaling out and scaling up), so you're not confined to one approach. It also takes pride in reducing Cassandra's infrastructure footprint by 80%.

Cassandra uses consistent hashing to distribute data across nodes. Still, workload imbalances can occur from partition hotspots, poor token distribution, and other factors. For example, tables with low partition cardinality may be unevenly distributed. Aerospike automatically shards data into 4,096 logical partitions evenly distributed across cluster nodes. When cluster nodes are added, partitions from other cluster nodes are automatically migrated to the new node, resulting in very little data movement. The Aerospike data rebalancing mechanism distributes query volume evenly across all cluster nodes without operator involvement.

3. You need fast, predictable performance with high availability

Achieving fast, predictable performance at scale can be challenging with Cassandra, particularly in the 99th percentile. Cassandra's performance challenges, such as JVM garbage collection, compaction processes, inefficient data distribution, and network overhead, often result in large latency spikes in the 99th percentile and beyond. But it's not just Cassandra's implementation language (Java) that makes it difficult to achieve ultra-fast, predictable performance. Cassandra's use of LSM trees for storage helps it efficiently manage writes but tends to slow reads. This forces operators to explore various techniques to mitigate potential read performance challenges.

In contrast with Cassandra, Aerospike was designed to efficiently process reads and writes in mixed workload scenarios. Its performance is fast and predictable.

Providing predictably fast data access involves maintaining consistent response times and ensuring that operations are completed smoothly and without significant latency spikes. Aerospike is designed to efficiently process reads and writes in mixed workload scenarios, delivering both fast and predictable performance.

On the other hand, high availability is achieved by maintaining copies (replicas) of user data. Cassandra and Aerospike provide high availability through replication but differ in their strategies. Cassandra typically uses a replication factor of 3, while Aerospike recommends that users run with only a factor of 2 (typical in production environments). Aerospike's ability to deliver 24x7 data availability with fewer replicas helps speed operations and reduce costs.

4. Your operations team keeps growing (as do your costs)

Controlling operational costs as workloads and data volumes increase is essential, and using a data platform that is easy to manage and self-healing can give firms a critical edge. Yet many Cassandra installations require a large operations team with rare, specialized skillsets to continually monitor clusters for changes in application patterns and re-tune frequently. Failure to do so leads not only to poor performance but also (eventually) to CPU pressure, which limits garbage collection capabilities; this causes memory pressure, and in turn, outages. To increase utilization, clusters are forced to scale out — an approach that can solve one problem but introduce others. Cassandra suggests running a full repair after adding a node to the cluster or increasing the keyspace's replication factor to prevent data inconsistencies. However, full repairs can require considerable disk and network I/O, slowing performance and leading to unpredictable data access speeds until the job is completed. Managing very large clusters increases operational complexity, requiring more time from your operations staff to plan and deploy as well as more time to diagnose and fix increasingly complex and compounding failures.

By contrast, Aerospike's greater efficiency with computing resources enables firms to do more with fewer nodes. It automatically detects and responds to many network and node failures, preventing data loss or performance degradations without requiring operator intervention. Data distribution, and redistribution when cluster status changes, are automatic and designed to maintain optimal balance. Simply put, there's less need to monitor, tune, and test with a production Aerospike cluster, which allows your staff to focus more on business issues and less on infrastructure issues.

Finally, Cassandra's tunable consistency requires programmers to understand nine consistency-level settings and select what is appropriate for their operational needs, taking availability, latency, throughput, and data correctness into account. Note that

Cassandra has not passed the Jepsen test for strong consistency under partition tolerance (CP), whereas Aerospike has, and published benchmark data demonstrating a negligible performance difference between AP and CP mode in Aerospike.

In short, it's not just the number of nodes that challenge Cassandra operators; it's the complexity of tuning, configuration, and data modeling to avoid performance, availability, and scalability pitfalls associated with wide partitions, tombstones, data skew, and more.

5. You need more flexible data modeling and deployment options

As customer demands and competitive landscapes change, flexibility and agility are increasingly critical. Aerospike supports a wider range of data modeling options and deployment patterns than Cassandra, making it better suited to serving a broader set of applications. Having a single platform that can efficiently handle a greater number of business needs simplifies IT infrastructures and helps cut costs.

Cassandra's JSON support is limited: although Cassandra has a JSON data type, document-based operations are not supported, requiring operations on JSON elements to be applied on the client side. This adds extra latency from clients who are fetching records over the network. Graph use cases would require the addition of, e.g., JanusGraph, a separate solution that can use Cassandra as its storage layer.

By contrast, Aerospike is a multi-model database storing sets of records in namespaces ("databases"). Each record has a key and named fields ("bins"), and each bin can contain different types of data, from simple (e.g., integer, string) to complex (e.g., nested data in a list of maps of sets). Aerospike enables users to model, store, process, and manage key-value data, JSON documents, and graph data with high performance at scale. Knowledge and support for these data models are built into server-side software. Aerospike's schema flexibility enables firms to deploy the platform for various applications beyond the typical NoSQL applications.

Many organizations have diverse data needs, requiring both high-volume, storage-heavy use cases and others that should run in memory. This sometimes necessitates a complex setup combining a cache and a persistent database. Cassandra is limited as a persistent data store because, although it caches frequently accessed data internally and temporarily stores writes in memory before flushing to disk, it cannot serve as a front-end cache.

Aerospike, on the other hand, can serve as both a persistent data store and an in-memory-only caching platform, allowing firms to configure it to store all data and indexes in DRAM, all data and indexes on SSDs (Flash), or a combination of the two (data on SSDs and indexes in DRAM). These flexible deployment options enable firms to standardize on Aerospike for many business needs, reducing the overall complexity of their data management infrastructures and avoiding the need to train staff on multiple technologies. Many firms initially deploy Aerospike as a cache to promote real-time access to other systems of record or systems of engagement and later leverage Aerospike's built-in persistence features to support additional applications. Aerospike supports on-premises, cloud, and hybrid deployments.

Summary

Many firms once turned to Cassandra as a reliable workhorse for their NoSQL workloads for its scalability and flexibility despite the substantial effort required for tuning and maintenance. However, as data volumes and workloads grow, these firms face unpredictable performance, sprawling server footprints, and ballooning infrastructure and personnel costs.

Aerospike offers a compelling alternative. As many former Cassandra users have already discovered, Aerospike provides ultra-fast, predictable performance for mixed operational workloads at scale. And it does so at a fraction of the operational cost of Cassandra and other alternatives. If you're struggling to achieve what you want with Cassandra or have experienced any of the five signs just discussed, why not explore what Aerospike can do for you? Contact Aerospike to estimate TCO savings for your workload, or try Aerospike to discover the benefits for yourself.

About Aerospike

Aerospike is the real-time database built for infinite scale, speed, and savings. Our customers are ready for what's next with the lowest latency and the highest throughput data platform. Cloud and AI-forward, we empower leading organizations like Adobe, Airtel, Criteo, DBS Bank, Experian, PayPal, Snap, and Sony Interactive Entertainment. Headquartered in Mountain View, California, our offices include London, Bangalore, and Tel Aviv.

For more information, please visit https://www.aerospike.com.