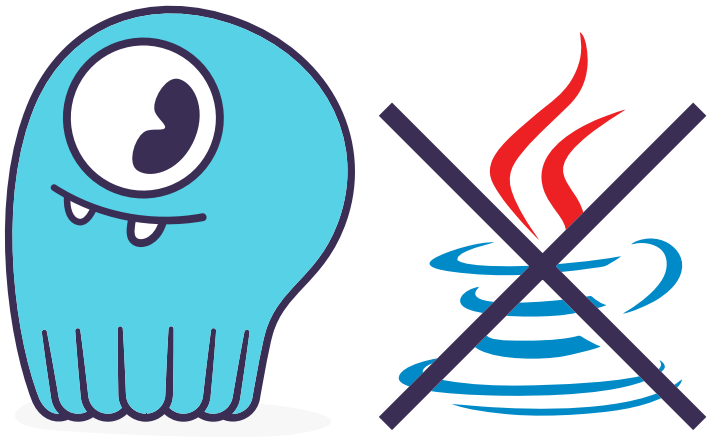
SCYLLA

ScyllaDB: The Best of Cassandra, without the Java Overhead

Greater Performance.
Simpler Administration.
Lower Cost.



WHY CASSANDRA CAN BECOME A LOVE-HATE RELATIONSHIP

Your application's data and scalability needs warrant a distributed NoSQL database. A wide-column NoSQL database will deliver the fastest querying speed, so you look to Cassandra. It's utilized by thousands of companies including Netflix and Apple, is consistently ranked high in DB-Engines popularity, and is free with a robust open source community. So why do many Cassandra users later find themselves seeking alternatives?

It's not for lack of support. There are several companies with Cassandra expertise that provide 24x7 enterprise-class support if needed. And if you wish to be free of cluster infrastructure and administration, you can utilize a fully managed Cassandra cloud service such as DataStax Astra and Amazon Managed Apache Cassandra Service. But these won't solve slow release cycles and a lack of critical features. And they won't address the two biggest challenges with Cassandra — performance at scale and the associated costs.

Cassandra runs in a Java Virtual Machine (JVM) and as a result, must contend with Java overhead and the inherent impact on throughput and latency. For small clusters, it's often tolerable. But as environments grow, performance challenges mount. Java consumes a lot of memory and managing memory is an art form. Garbage collection (GC) pauses — when all application traffic stops — can run hundreds of milliseconds to seconds and become a frequent headache. Determining optimum heap size and adjusting JVM parameters to mitigate GC issues and improve performance requires expertise, varies by workload, and consumes staff resources.

Adding more nodes is a common remedy for GC challenges and boosting performance, but with often disappointing results. Because Java's write once, run anywhere architecture hides low level system attributes, Cassandra is hindered from making efficient use of system hardware. It's why clusters with fewer, larger nodes fail to outperform clusters with more, smaller nodes. Why keeping nodes under 1TB is generally recommended. And why servers with newer, faster processors, faster memory, and faster I/O often fall short of expectations. All of which leads to node sprawl, rising costs, and never-ending performance tuning.

WHY IS SCYLLADB SUCH A POPULAR CASSANDRA ALTERNATIVE?

There is admittedly much to love about Cassandra's approach. If one could model a database after Cassandra but without the Java overhead, you would have an ideal distributed NoSQL solution for a myriad of next gen applications. So that's what we did.

ScyllaDB uses the same highly available ring architecture as Cassandra. The same SSTable data format. The same Apache Cassandra Query Language (CQL) as the primary language for commands and database queries. It even supports the same drivers, making migration to ScyllaDB seamless. But ScyllaDB is built from the ground up in C++. No Java overhead. No garbage collection. And performance tuning? It's automated.

Unlike Java, ScyllaDB's C++ code places the database "close to the metal" and enables it to take full advantage of low-level Linux primitives and harness the ever increasing computing power of modern infrastructures. (Our founders developed the KVM hypervisor in Linux, so we know a thing or two about virtualization, hardware interfaces, and Linux). Every ounce of CPU, memory, I/O and storage resources are utilized to their fullest. It all adds up to higher performance using dramatically fewer nodes, far less administration, and lower infrastructure costs. And if you're looking for a fully managed service, it just gets better with ScyllaDB Cloud.

400+ companies worldwide use ScyllaDB for data-intensive apps that require high throughput and low latency.

Discord Santander Expedia Fanatics Zillow

instacart Disney+ hotstar Tripadvisor IOTA OLA

crypto.com OpenSea tubi STRAVA KIWI.COM

ShareChat DREAM11 GE Healthcare DELHIVERY imvu natura bem estar bem

Rakuten Grab COMCAST paloalto NETWORKS

amdocs IBM sas SAMSUNG SDS

Sample results:

Fanatics went from **55 nodes to 6** and dramatically reduced their AWS EC2 bill moving to ScyllaDB.

Comcast reduced their infrastructure from **962 to 78 nodes** and improved P99 latency by more than 95% by moving to ScyllaDB.

Expedia **increased throughput 3x** and **lowered infrastructure cost by 30%** by moving to ScyllaDB.

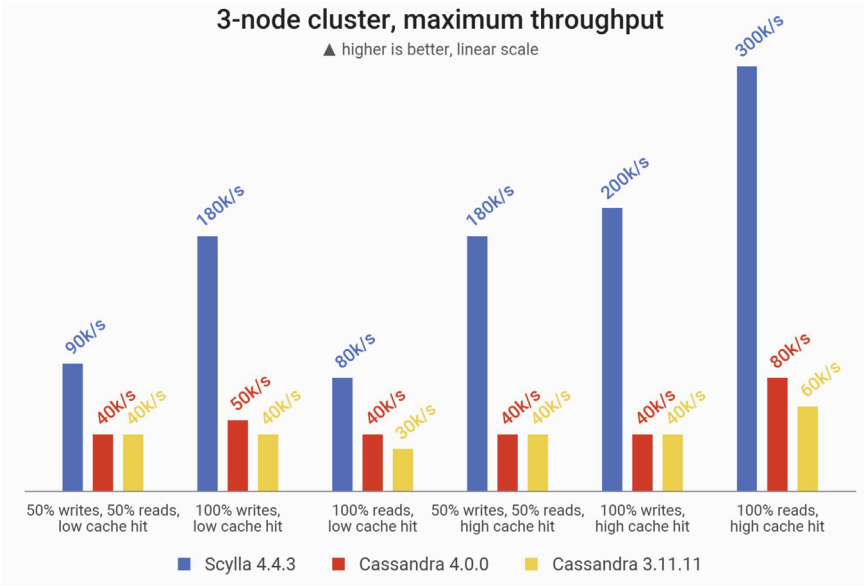


Figure 1: The maximum throughput (measured in operations per second) achieved on 3 x i3.4xlarge machines (48 vCPUs)

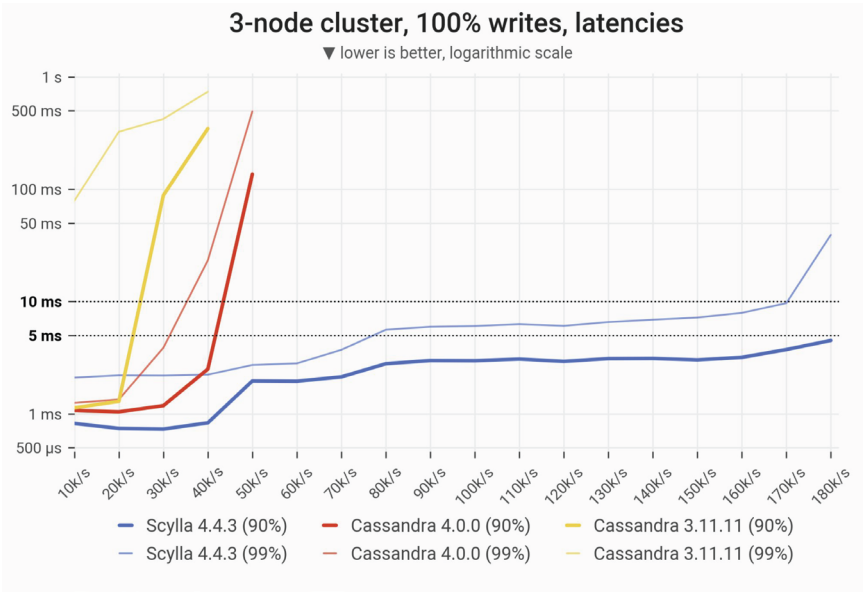


Figure 2: The 90- and 99-percentile latencies of UPDATE queries, as measured on three i3.4xlarge machines (48 vCPUs in total) in a range of load rates

HOW DOES SCYLLADB COMPARE TO CASSANDRA?

ScyllaDB:

- Achieves 2x-5x better throughput and much better latencies than Cassandra
- Adds and replaces nodes far faster than Cassandra
- Finishes compaction 32x faster than Cassandra
- Can be provisioned for 60% lower cost than Cassandra

WHY MIGRATE FROM CASSANDRA TO SCYLLADB

Optimal Price-Performance

The total cost of ownership (TCO) with ScyllaDB is much less than that of Cassandra. ScyllaDB was designed from the ground up to deliver the best possible price-performance. Its low-level design squeezes every cycle from your CPU. ScyllaDB is designed to run as close as possible to 100% CPU utilization, with every operation given a priority class. There is no need to wastefully over-provision.

Consistent Performance

ScyllaDB delivers consistent, reliable real-time performance. That's because its built-in schedulers prioritize reads and writes over maintenance tasks, such as repairs and compactions — eliminating latency spikes. With ScyllaDB, you never experience garbage collection stalls, which impact Cassandra performance. To avoid such interruptions, ScyllaDB adopts a comprehensive 'shared-nothing' design. No locks are taken, so latency is never affected.

Operational Simplicity

Speed doesn't have to come at the price of complexity. ScyllaDB simplifies everything for its users. It automatically configures the RAID device for you with the right striping and automatically assigns the NICs network queues to shards. ScyllaDB installs daemons in an isolated Linux control group to cap their memory/CPU usage. ScyllaDB's setup tool runs a disk benchmark to maximize throughput while keeping latency low. And with ScyllaDB, DevOps can leave tuning to the database itself.

Maintainability

ScyllaDB has a notable maintainability advantage as well. It scales up to any number of cores and can stream data to a 60TB meganode just as fast as it does to smaller nodes. These capabilities enable you to shrink your Cassandra cluster by 10x. Rolling restarts, for example, become 10x faster. ScyllaDB add-node and decommission operations are restartable. You can pause them and resume them from the previous point. Compaction, a maintenance headache with Cassandra, is a solved problem in ScyllaDB.

Better Functionality

You can simply do more with ScyllaDB than you can with Cassandra. ScyllaDB supports change-data-capture (CDC) as a CQL table, so you can easily track your database changes in a consistent way with the same query language you already know. On Cassandra, by comparison, CDC is a commitlog-like structure you have to write custom programs to interact with and requires deduplication. ScyllaDB supports global and local secondary indexes simultaneously (Cassandra supports only local secondary indexes). ScyllaDB's unique Workload Prioritization feature enables you to assign a relative priority to different user workloads in a simple role-based fashion. That way you can safely run transactional workloads alongside analytics workloads, all while consolidating your data infrastructure and making it easier to manage.

Easy Migration

Teams can easily migrate from Cassandra to ScyllaDB. ScyllaDB and Cassandra are identical where it counts: The CQL protocol and queries, nodetool, SSTables and compaction strategies — even JMX support. ScyllaDB supports many of the same open-source projects and integrations as Cassandra, including JanusGraph, Spark, Kafka (using our optimized ScyllaDB connector), Presto, KairosDB, Kong, and many others. ScyllaDB even provides a Spark-based ScyllaDB Migrator and Migration Guide to easily move your data from your existing Cassandra clusters into ScyllaDB. To enable even more NoSQL database consolidation, ScyllaDB also supports a DynamoDB-compatible API, so you can migrate even more use cases.

NEXT STEPS

Get started with ScyllaDB

Learn more at ScyllaDB University

Explore papers, videos, benchmarks & more

ABOUT SCYLLADB

ScyllaDB is the database for data-intensive apps that require high performance and low latency. It enables teams to harness the ever-increasing computing power of modern infrastructures – eliminating barriers to scale as data grows. Unlike any other database, ScyllaDB is built with deep architectural advancements that enable exceptional end-user experiences at radically lower costs. Over 400 game-changing companies like Disney+ Hotstar, Expedia, FireEye, Discord, Crypto.com, Zillow, Starbucks, Comcast, and Samsung use ScyllaDB for their toughest database challenges. ScyllaDB is available as free open source software, a fully-supported enterprise product, and a fully managed service on multiple cloud providers. For more information: ScyllaDB.com

SCYLLADB.COM

SCYLLA

United States Headquarters
2445 Faber Place, Suite 200
Palo Alto, CA 94303 U.S.A.
Email: info@scylladb.com

Israel Headquarters
11 Galgalei Haplada
Herzelia, Israel