MCKNIGHT
CONSULTING GROUP

Database performance comparison: Aerospike, DynamoDB, and DynamoDB with DAX

Aerospike
DynamoDB
DynamoDB with DAX

William McKnight

Jake Dolezal

AUGUST 2025

Table of contents

Executive summary

This benchmark report compares the performance and cost-effectiveness of Aerospike, DynamoDB, and DynamoDB with DAX across various data sizes (1 TB, 5 TB, and 10 TB) for both 70/30 read/write and 100% write workloads. Aerospike consistently outperformed both DynamoDB and DynamoDB with DAX in terms of latency, throughput, and cost efficiency.

The findings of this benchmark provide valuable insights for organizations seeking to optimize their performance and costs for large-scale data storage and management while ensuring uptime.

Key findings:

- Aerospike delivered p99 latencies ranging from 77% to 93% lower than DynamoDB across workloads and data sizes, and 77% to 91% vs. DynamoDB with DAX.
- Across different workloads and data sizes, Aerospike achieved throughput that was 27 to 112 times higher than DynamoDB, and 25 to 118 times higher than DynamoDB with DAX.
- Aerospike's cost per transaction is at least 50 times lower than that of DynamoDB or DynamoDB with DAX.[1]

These results suggest that Aerospike is a strong choice for applications requiring high performance and cost efficiency, particularly for large-scale data sets.

[1] Aerospike software licensing costs are not published and thus are estimated per Pricing section, along with DBA costs. Also, Aerospike was tested in a single Availability Zone (AZ), and thus did not incur data transfer costs.

Benchmark methods and platforms

To evaluate and compare the performance characteristics, we designed a benchmarking framework focused on the top three critical dimensions for their workloads: latency, throughput, and cost-efficiency. These metrics reflect common priorities in real-time and high-throughput application scenarios, where data systems must deliver consistent performance at scale while maintaining operational affordability.

Test design

Our methodology is grounded in realistic usage patterns, simulating read-heavy, write-heavy, and mixed workloads across varying data volumes. Each database was deployed and tested in a controlled environment that ensures consistency in hardware specifications, network conditions, and data models. To reflect the differences between in-memory and disk-backed use cases, we used out-of-the-box configurations to align with each system's deployment model and best practices recommended by vendors.

YCSB

The Yahoo! Cloud Serving Benchmark (YCSB) is a widely adopted open-source framework designed to facilitate the evaluation of the performance characteristics of modern cloud-based and NoSQL data stores. Originally developed by Yahoo! Research, YCSB provides a standardized methodology to measure key performance metrics, primarily latency and throughput, under various workloads that simulate real-world usage patterns. We cloned the YCSB GitHub repository.

To run the 5 TB and 10 TB tests, you need to slightly alter the core/src/main/java/site/ycsb/workloads/CoreWorkload.java file:

Line 367, change this:

```
 protected int recordcount
```

To this:

```
 protected long recordcount
```

Lines 434:435, change this:

```
  recordcount =
    Int.parseInt(…
```

To this:

```
  recordcount =
    Long.parseLong(…
```

Lines 448:451, change this:

```
  int insertstart =
    Int.parseInt(...
  int insertcount=
    Int.parseInt(...
```

To this:

```
  long insertstart =
    Long.parseLong(...
  long insertcount=
    Long.parseLong(...
```

And then recompile with Maven.

Because the `recordcount` property in those workloads is 3,333,333,333 and 6,666,666,667, respectively, and that is greater than the max value for a `Java Int` (2,147,483,647). Although we did not use them, the `insertstart` and `insertcount` parameters also depend on `recordcount` as well, and Java will complain if they are not `Long,` too.

Scale factor

We conducted our tests using three scale factors of pre-generated YCSB data: 1 TB, 5 TB, and 10 TB. The schema is a single table (`usertable`) with the following YCSB configuration:

YCSB parameter	Value
`fieldcount`	10
`fieldlength`	150
`fieldlengthdistribution`	`constant`

Thus, each record is 1.5KB. To create the three data sizes, we used the following YCSB parameters:

YCSB parameter	Scale	Value
`recordcount`	1 TB	666,666,667
	5 TB	3,333,333,333
	10 TB	6,666,666,667

Workloads

We tested two different workloads with different read-to-write ratios:

- **70/30** – 70% reads/30% writes (updates only, no inserts)
- **100/0** – 100% reads only

In YCSB, you can set the read/write ratio to simulate different workload scenarios. We chose the 70/30 read-heavy and 100/0 read-only ratios to benchmark as they represent different, yet typical workload scenarios: typical transactional mixed workload activity and read-intensive cases. These provide a comprehensive evaluation of database performance, scalability, and efficiency.

To run these workloads, we used the following YCSB settings:

YCSB parameter	Scale	Value
`readproportion`	0.7	1.0
`updateproportion`	0.3	0
`scanproportion`	0	0
`insertproportion`	0	0
`requestdistribution`	`uniform`	`uniform`

We also used a high number of operations to ensure the workload ran long enough to assess the stability of the workload on each platform:

YCSB parameter	Scale	Value
`operationcount`	1 TB	100,000,000
	5 TB	500,000,000
	10 TB	1,000,000,000
`hdrhistogram.percentiles`	-	95, 99, 99.9

We ran these workloads and captured latencies (average, 95th, 99th, and 99.9th percentiles) and throughput (operations per second), as shown in the next section.

Systems under test

Our test involved the following systems:

Aerospike

Enterprise Edition
v.8.0.0.5

Amazon DynamoDB with DynamoDB Accelerator (DAX)

We added DAX to DynamoDB to leverage its in-memory caching capabilities, which can significantly improve performance by reducing latency and increasing throughput. By integrating DAX, we aimed to assess whether DynamoDB's performance would become more competitive with Aerospike in terms of latency and throughput, while also considering the impact on cost-efficiency.

Amazon Web Services (AWS) recommends using Amazon DynamoDB Accelerator (DAX) for applications that require microsecond latency, high-throughput performance, and consistent response times[2]. By using DAX, developers can optimize their DynamoDB performance and improve application responsiveness.

[2] https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/dax-prescriptive-guidance.html

Additional configurations

While we installed and tested these platforms with an “out-of-the-box” configuration, we made a few minor configuration changes based on documented best practices for this particular use case:

Aerospike

- Used a replication factor of 2 (RF2)
- Enabled read-page-cache[3] to cache reads the first time a record is accessed and improve read latency on subsequent reads of the same record
- Partitioned each NVMe SSD into four equally sized partitions to improve I/O throughput

DynamoDB + DAX

- Used DynamoDB Standard table class for frequently accessed data, with throughput (reads and writes) as the dominant table cost
- Used Provisioned capacity mode

Infrastructure

We also took into account the sizing implications of scaling each system under these workload demands. This includes infrastructure for a self-managed Aerospike deployment and DynamoDB’s managed cloud service. The goal is to provide a comprehensive view of the trade-offs between raw performance and cloud infrastructure requirements, enabling practitioners to make informed architectural decisions.

Aerospike

Since Aerospike recommends using fast NVMe solid-state drives (SSDs), we chose EC2 instances that had an adequate amount of disk storage for the primary index using Aerospike’s documented guidance[4].

Data size	**1 TB**	**5 TB**	**10 TB**
Instance type	c7gd.4xlarge	i8g.4xlarge	i8g.4xlarge
vCPU per node	16	16	16
Memory per node	32 GB	128 GB	128 GB
SSD disk space per node	1 x 950 GB (NVMe)	1 x 3750 GB (AWS Nitro)	1 x 3750 GB (AWS Nitro)
Root disk size per node	gp2 20GB 100iops	gp2 20GB 100iops	gp2 20GB 100iops
Node count	5	7	14

Table 1: Cluster configurations

[3] https://support.aerospike.com/s/article/How-and-when-to-set-read-page-cache-to-true

[4] https://aerospike.com/docs/database/manage/planning/capacity

DynamoDB + DAX

As previously mentioned, we used provisioned capacity for DynamoDB rather than auto-scale, so that we would not have an extended warm-up period waiting for DynamoDB to catch up to our high-volume workloads.

Data size	**All**
Provisioned reads	70,000
Provisioned writes	30,000

Table 2: DynamoDB, DynamoDB provisioned reads and writes

For DAX, we provisioned an even number of compute nodes that could cache nearly 50% of the data volume.

Data size	**1 TB**	**5 TB**	**10 TB**
Instance type	dax.r5.8xlarge	dax.r5.8xlarge	dax.r5.16xlarge
Node count	2	10	10
Memory per node	244 GB	244 GB	488 GB

Table 3: DynamoDB Accelerator (DAX) instances

Benchmark results

Performance latency

In YCSB tests, latency is a crucial metric because it directly impacts the user experience and system performance. Lower latency indicates faster response times, which is essential for:

- **Real-time applications**: Latency affects how quickly data is retrieved or updated, impacting applications that require immediate responses.
- **User experience**: Higher latency can lead to slower page loads, frustrated users, and decreased engagement.
- **System scalability**: As latency increases, systems may become bottlenecked, limiting their ability to handle increased traffic or workload.

DynamoDB Accelerator (DAX) is used for caching on top of DynamoDB to improve latency. The cache hit ratio targeted was 50%, and results were within 5% of this value across workloads and data sizes.

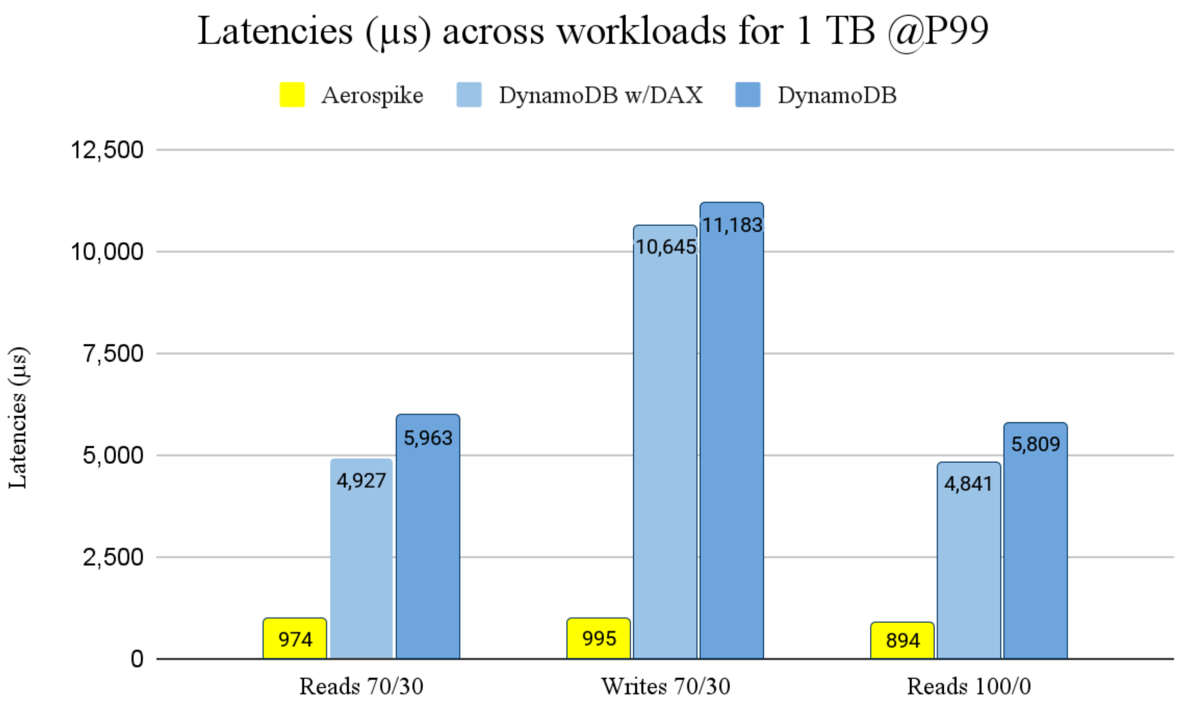


Figure 1: Latencies across workloads for 1 TB @P99

The performance comparison of Aerospike, DynamoDB with DAX, and DynamoDB under different workloads (70/30 reads/writes and 100/0 reads) with 1 TB data size and replication reveals that Aerospike performs the best, with the lowest latency across all workloads (974, 995, and 894 microseconds). DynamoDB has the highest latencies for each workload, as expected. The results show that DAX improves DynamoDB latencies, but does not match Aerospike. Aerospike is optimized for efficient operations in these tests, handling reads and writes effectively.

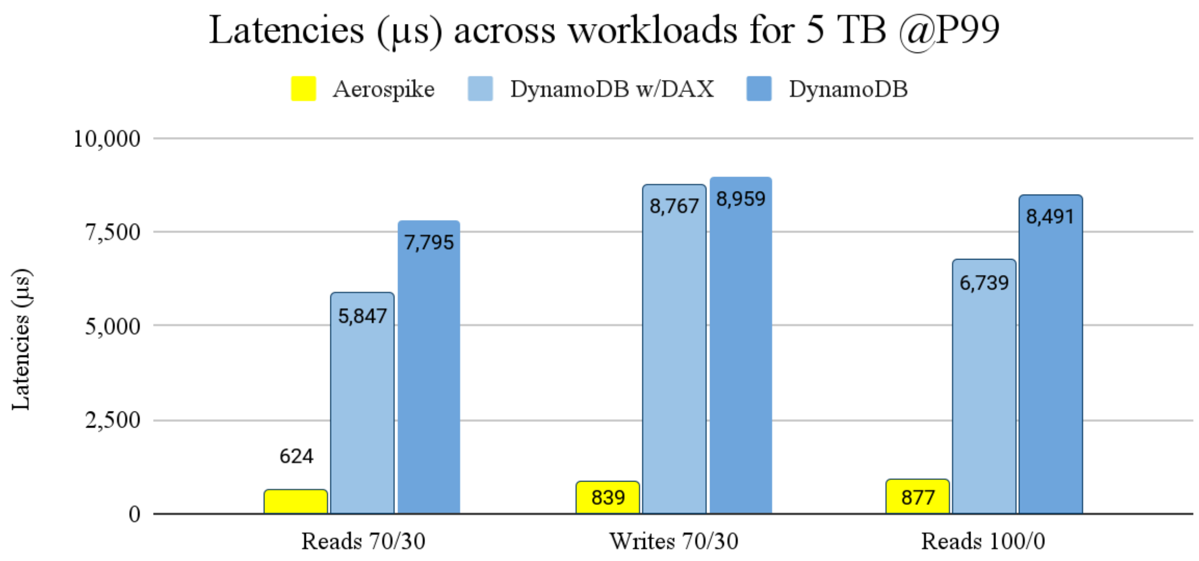


Figure 2: Latencies across workloads for 5 TB @P99

The performance comparison of Aerospike, DynamoDB with DAX, and DynamoDB under different workloads (70/30 reads/writes and 100/0 reads) with 5 TB data size shows Aerospike performs the best, with the lowest latency across all workloads (624, 839, and 877 microseconds). DynamoDB has the highest latencies for all workloads at this scale. The results suggest that Aerospike's efficiency and scalability hold up well with the increased 5 TB data size.

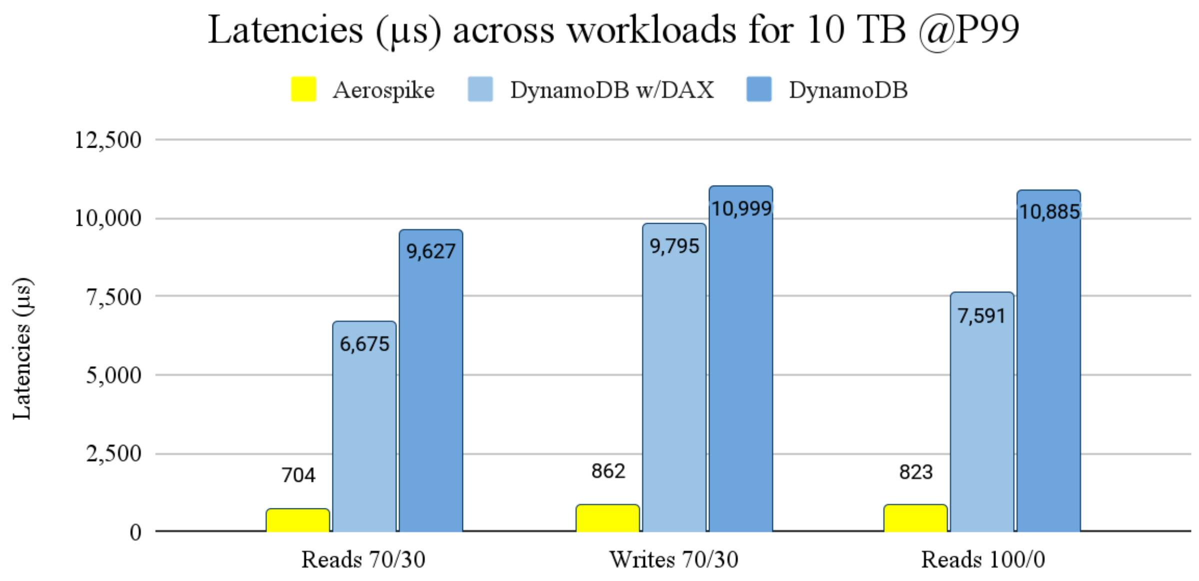


Figure 3: Latencies across workloads for 10 TB @P99

The established pattern continued at 10 TB, with Aerospike once again showing the lowest latencies across all workloads. It is reasonable to conclude that this pattern would continue into higher volume.

Performance latency by workload and SLA

As performance by service level agreement (SLA) is often a deciding input into system design, these three charts exhibit each system's relative latency by SLA.

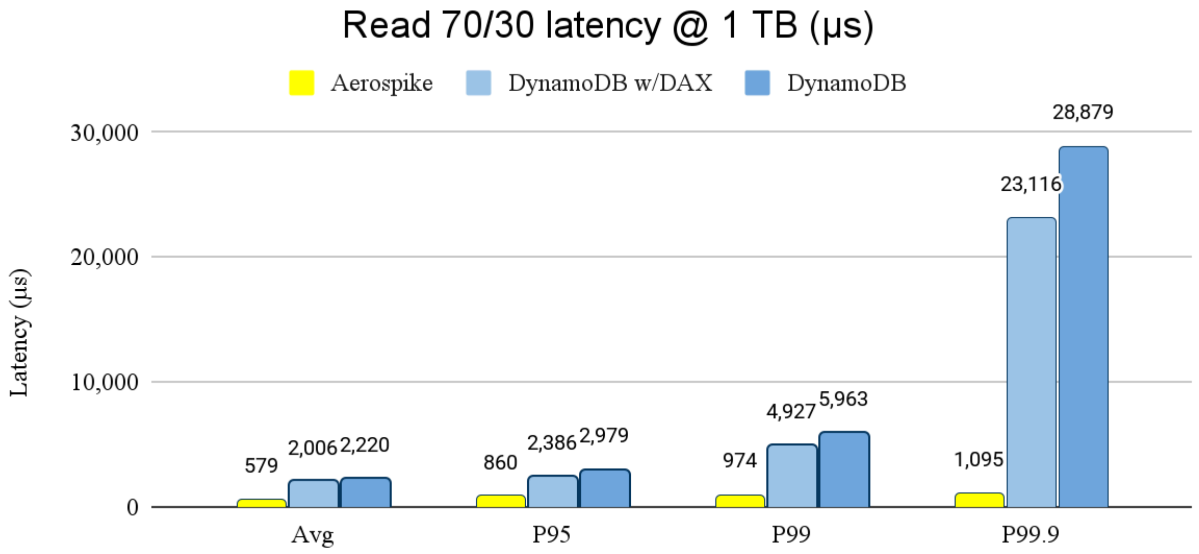


Figure 4: Read 70/30 latency @ 1 TB

Aerospike's relatively low latency across SLAs is less than both DynamoDB with DAX and DynamoDB's for 1 TB for the 70% reads from the 70/30 workload.

We see relatively similar behavior even with 30% writes of the 70/30 workload at 5 TB:

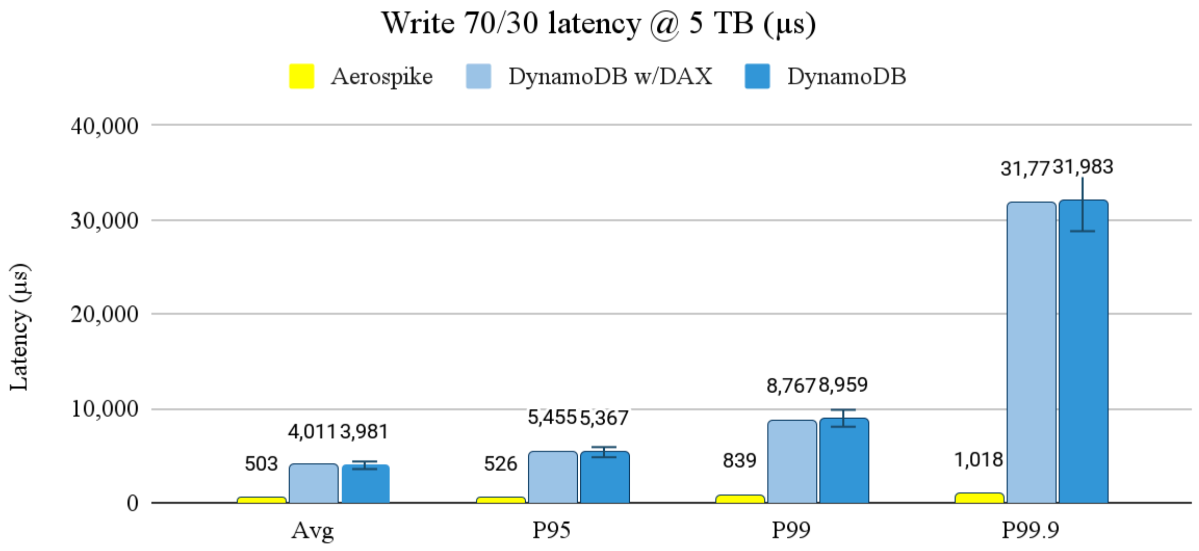


Figure 5: Write 70/30 latency @ 5 TB

For 100% reads from the 100/0 read-only workload latencies across SLA for 10 TB we found:

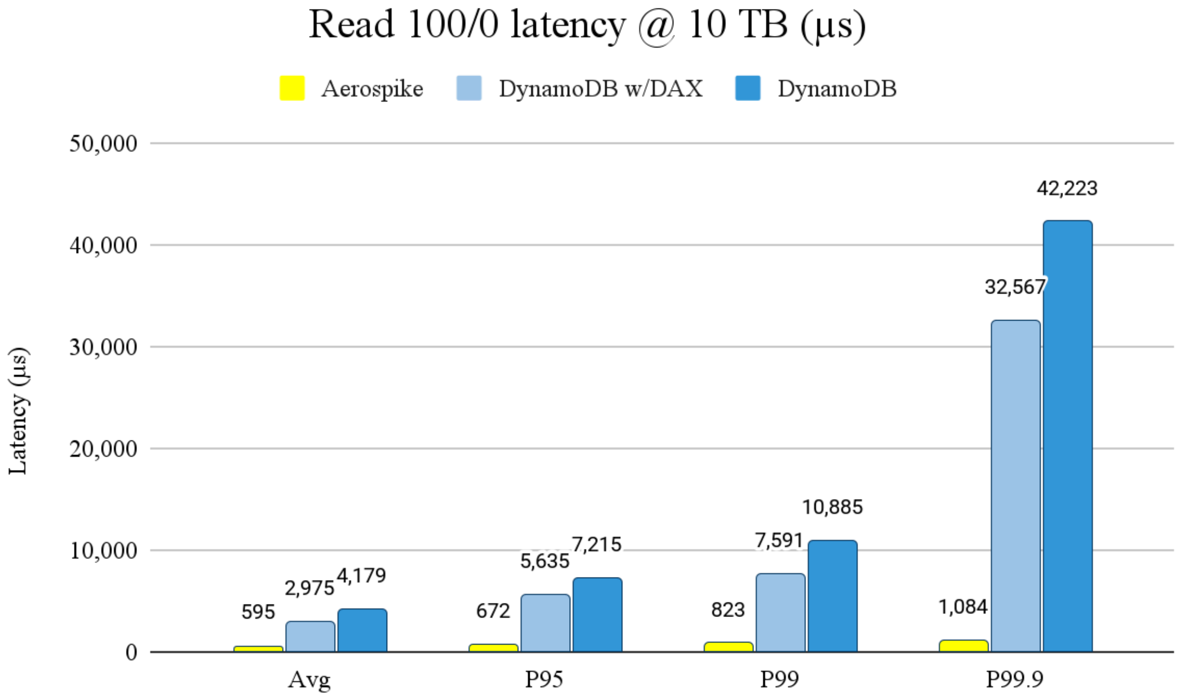


Figure 6: Read 100/0 latency @ 10 TB

Performance throughput

In YCSB tests, throughput is a critical metric that measures the number of operations (e.g., reads, writes, updates) a database or system can handle per unit of time. In our case, it is operations per second. **Higher throughput indicates better performance and scalability.**

Throughput is essential because it directly impacts:

- **System scalability**: Higher throughput enables systems to handle increased traffic, workload, or user demand.
- **Performance under load**: Throughput testing reveals how well a system performs under stress, helping identify potential bottlenecks.
- **Capacity planning**: Throughput metrics inform capacity planning, ensuring systems can handle expected workloads.

We measured throughput for the 70/30 reads/writes test and the read-only test because these workloads are more representative of typical use cases where read operations dominate yet with a modicum of writes. This allowed us to evaluate the system's performance under realistic conditions and identify potential bottlenecks in both read-heavy and mixed workload scenarios.

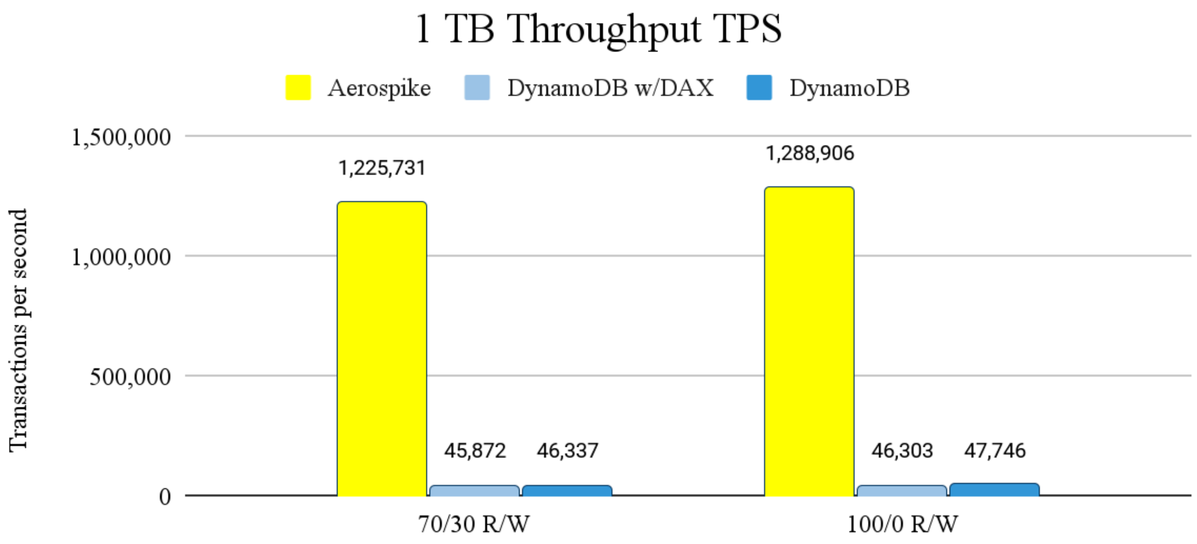


Figure 7: 1 TB throughput TPS

The throughput comparison reveals that Aerospike significantly outperforms other platforms at 1 TB, with notably higher throughput in both 70/30 reads/writes (1,225,731 TPS) and 100/0 reads (1,288,906 TPS) workloads. DynamoDB with DAX has much lower throughput (45,872 and 46,303), with similar results for DynamoDB. Compared to DynamoDB with DAX, Aerospike has greater than 25x the throughput for both workloads. Compared to DynamoDB, Aerospike also has greater than 25x the throughput for both workloads. Aerospike's high throughput suggests it is optimized for efficient operations in both read-heavy and mixed workload scenarios.

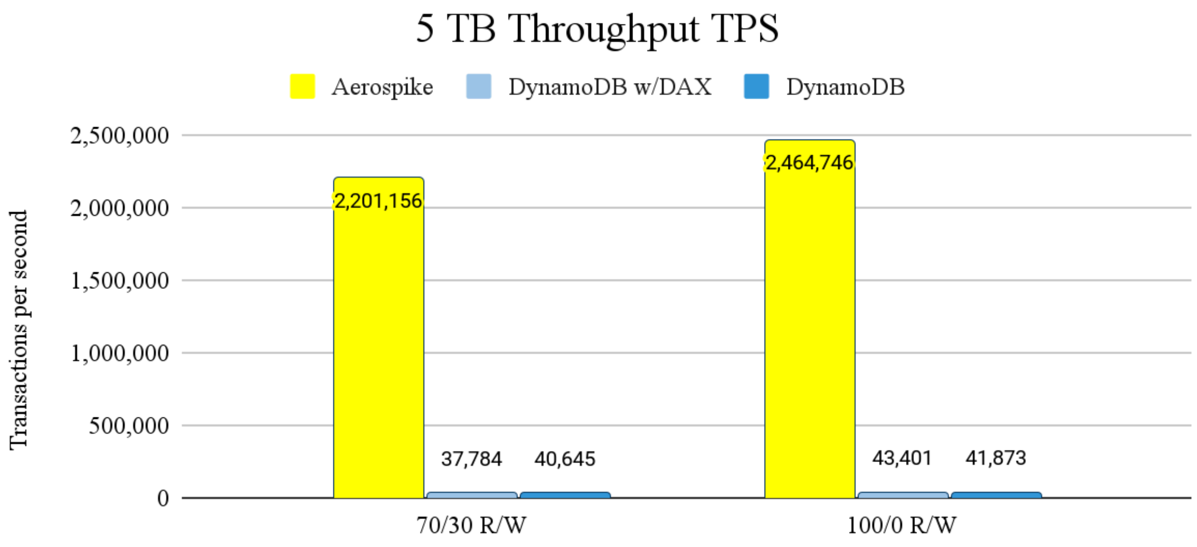


Figure 8: 5 TB throughput TPS

The throughput comparison shows Aerospike outperforming DynamoDB with DAX and DynamoDB in both 70/30 reads/writes and 100/0 reads workloads with 5 TB data size. Aerospike's throughput increases from the 1 TB cluster to 2,201,156 and 2,464,746 TPS, while both DynamoDB with DAX and DynamoDB slightly decrease from their 1 TB configurations to 37,784 and 43,401 TPS and 40,645 and 41,873 TPS, respectively. These results indicate Aerospike's strong performance and scalability in handling datasets of 5 TB.

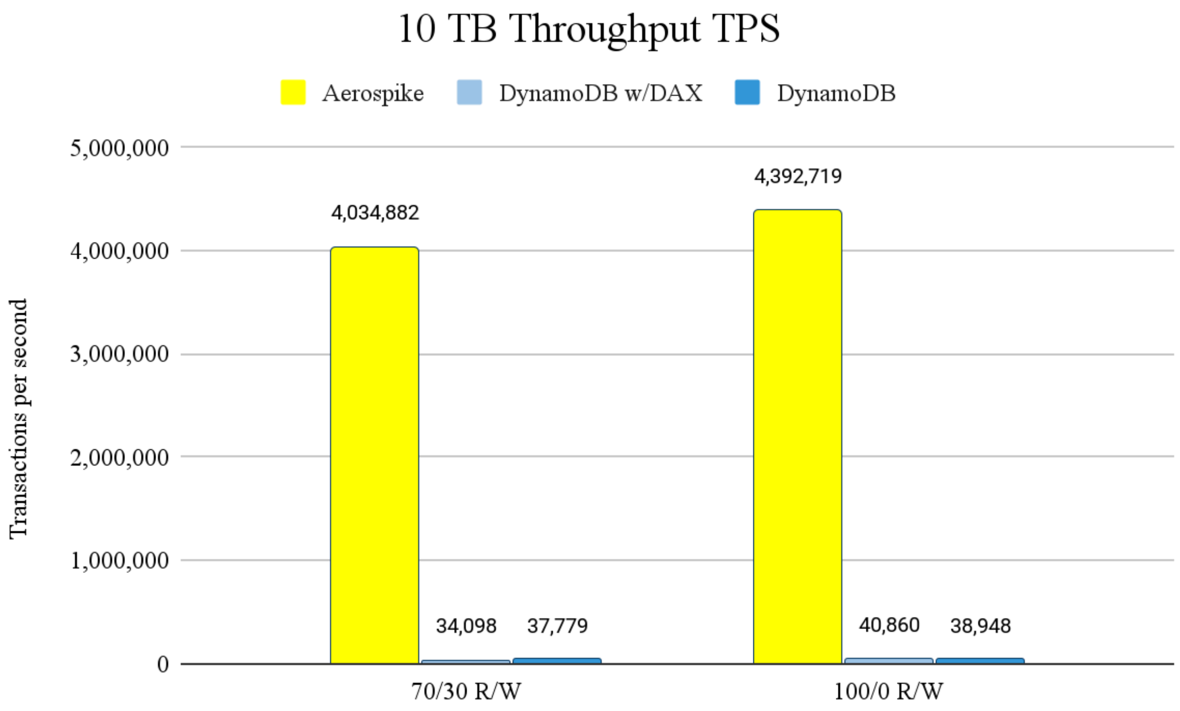


Figure 9: 10 TB throughput TPS

The established pattern continued at 10 TB, with Aerospike once again showing the most throughput by far across all workloads. Aerospike outperforms DynamoDB with DAX and DynamoDB in the 10 TB test by greater than a factor of 100 times the throughput in both the 70/30 read/write and 100/0 read/write workloads. It is reasonable to conclude that this pattern would continue into higher volumes.

Pricing

With these measured differences in latency and throughput, it might be expected that Aerospike would be the highest cost so we projected costs to an annual level to find out.

We note specifically that the pricing used is with upfront payment, which means that the costs are calculated based on the sum of on-demand (i.e., pay-as-you-go pricing) in addition to the upfront costs to reserve 70,000 reads and 30,000 writes per second for the DynamoDB and DynamoDB with DAX models. These upfront costs keep the on-demand cost rates down, and since we know the test will push the limits, it makes sense to price in this manner with these reserved amounts. For DynamoDB and DynamoDB, these throughput levels were selected in part for practicality as higher levels would have made benchmark testing prohibitively expensive.

Note, however, that Aerospike software licence costs, which are not publicly available, are estimated, along with the annual cost for a DBA to manage. Also, Aerospike was tested in a single Availability Zone (AZ), and thus did not incur data transfer costs. We encourage those interested to reach out to sales@aerospike.com. We also note that Aerospike prices are per unreplicated volume of data and not per server. See aerospike.com/products/features-and-editions/ to learn more.

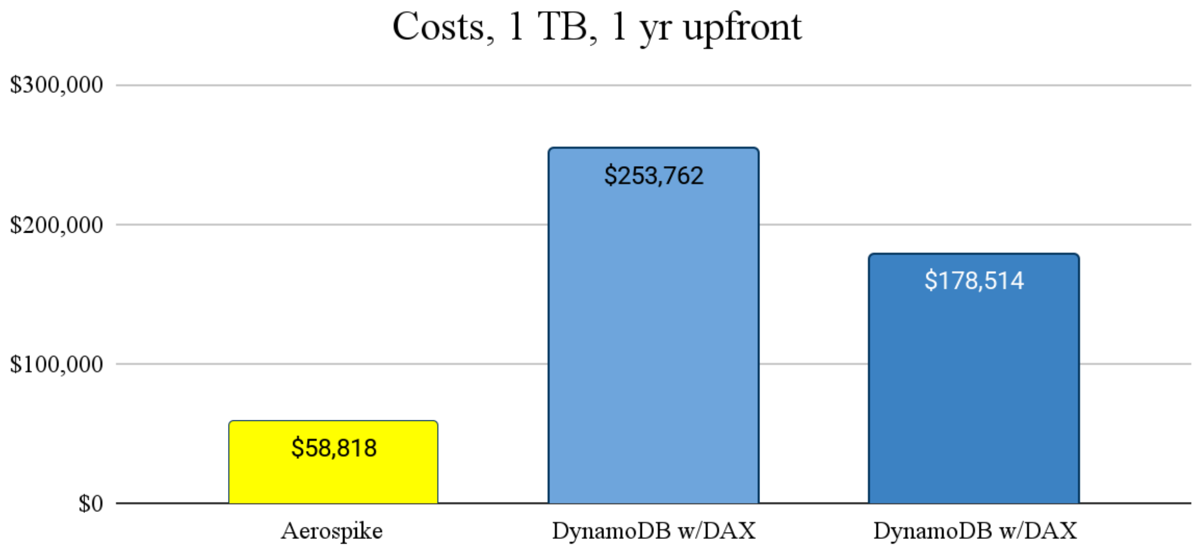


Figure 10: Costs, 1 TB, 1 year upfront

Aerospike is the most cost-effective option for a 1 TB data size, with an annual infrastructure cost of $18,818[5] plus an estimated software with DBA support costs added totaling $58,818. Compared to DynamoDB with DAX's $253,762 and DynamoDB's $178,514, Aerospike offers significant savings of 77% and 66%, respectively, making it the most economical choice.

[5] See Appendix: Pricing and infrastructure for instance costs.

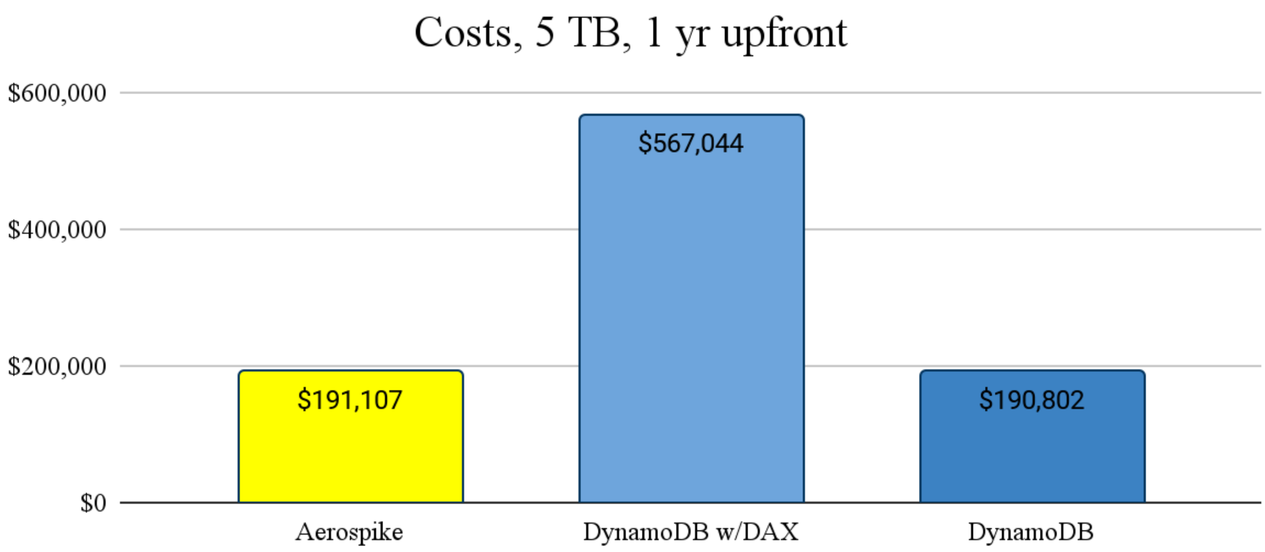
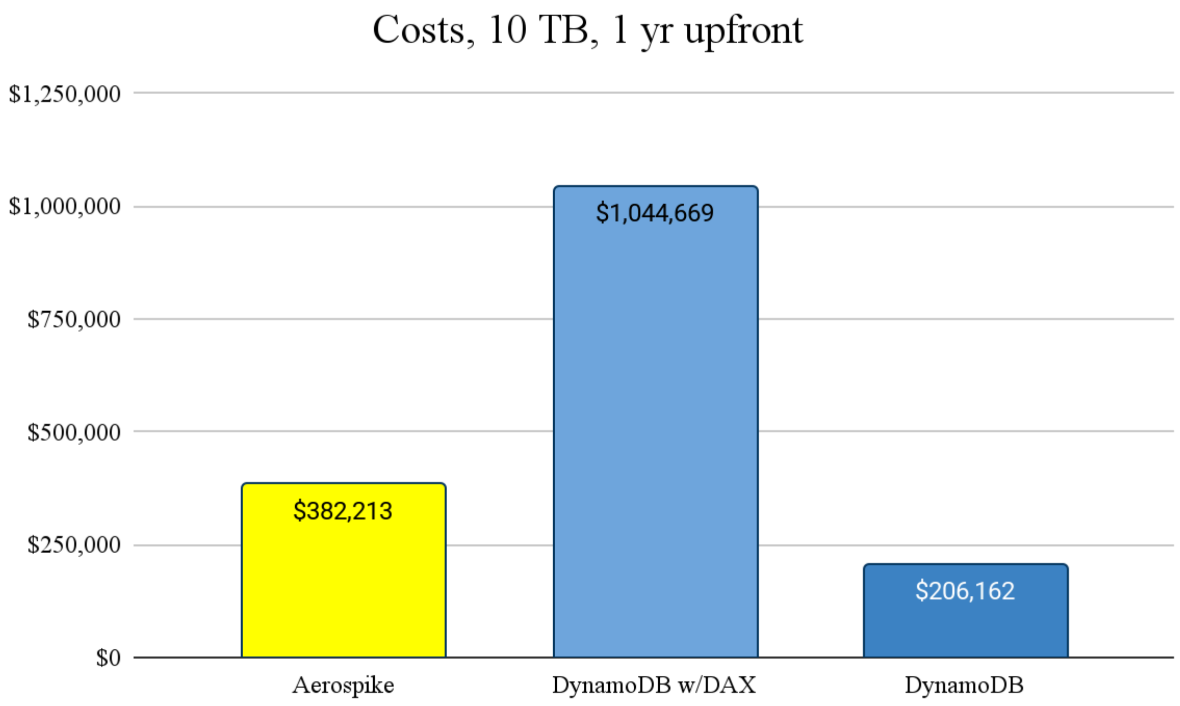


Figure 11: Costs, 5 TB, 1 year upfront

For the larger 5 TB data set, it is estimated that Aerospike would cost $191,107 with $51,107[6] in infrastructure, and the balance in estimated software licensing and DBA support (with some efficiencies of scale for licensing plus DBA support as the number of servers is only increased from 5 to 7 for 1 TB to 5 TB). Thus, Aerospike offers 66% cost savings to DynamoDB with DAX but no cost savings relative to DynamoDB.



[6] See Appendix: Pricing and infrastructure for instance costs.

Figure 12: Costs, 10 TB, 1 year upfront

Aerospike is 63% more cost-effective versus DynamoDB with DAX, but at an 85% cost disadvantage relative to DynamoDB. However, Aerospike was able to process far more transactions per second as per Figures 7, 8, and 9. Thus, we recommend looking at the next section, Price/Performance for further consideration.

Price/Performance

Given that one can scale infrastructure for more performance, this analysis employs the infrastructure used, the resulting costs, and the resulting throughputs.

Price/Performance, 1 TB					
	Aerospike	**DynamoDB with DAX**		**DynamoDB**	
70/30 ops/sec*	1,225,731	45,872		46,337	
100/0 ops/sec*	1,288,906	46,303		47,746	
Infrastructure Cost	$18,818	$253,762		$178,514	
Software + DBA	$40,000	$0		$0	
Total Cost	**$58,818**	**$253,762**	**Aerospike Advantage**	**$178,514**	**Aerospike Advantage**
Price/Performance 70/30 R/W ($/operation/sec)**	0.048	5.53	115x	3.85	80x
Price/Performance 100/0 R/W ($/operation/sec)**	0.046	5.48	120x	3.74	82x

Table 4: Price/Performance @ 1 TB

Aerospike has a price/performance advantage of more than 80x over DynamoDB and DynamoDB with DAX for both 70/30 R/W and 100/0 R/W tests at 1 TB.

Price/Performance, 5 TB					
	Aerospike	**DynamoDB with DAX**		**DynamoDB**	
70/30 ops/sec*	2,201,156	37,784		40,645	
100/0 ops/sec*	2,464,746	43,401		41,873	
Infrastructure Cost	$51,107	$567,044		$190,802	
Software + DBA	$140,000	$0		$0	
Total Cost	**$191,107**	**$567,044**	**Aerospike Advantage**	**$190,802**	**Aerospike Advantage**
Price/Performance 70/30 R/W ($/operation/sec)**	0.087	15.01	173x	4.69	54x
Price/Performance 100/0 R/W ($/operation/sec)**	0.078	13.07	169x	4.56	59x

Table 5: Price/Performance @ 5 TB

Aerospike has a price/performance advantage of more than 50x over DynamoDB and DynamoDB with DAX for both 70/30 R/W and 100/0 R/W tests at 5 TB.

Price/Performance, 10 TB					
	Aerospike	DynamoDB with DAX		DynamoDB	
70/30 ops/sec*	4,034,882	34,098		37,779	
100/0 ops/sec*	4,392,719	40,860		38,948	
Infrastructure Cost	$102,146	$1,044,669		$206,162	
Software + DBA	$280,000	$0		$0	
Total Cost	**$382,146**	**$1,044,669**	**Aerospike Advantage**	**$206,162**	**Aerospike Advantage**
Price/Performance 70/30 R/W ($/operation/sec)**	0.095	30.64	323x	5.46	58x
Price/Performance 100/0 R/W ($/operation/sec)**	0.087	25.57	294x	5.29	61x

Table 6: Price/Performance @ 10 TB

Aerospike has a price/performance advantage of more than 50x over DynamoDB and DynamoDB with DAX for both 70/30 R/W and 100/0 R/W tests at 10 TB.

Conclusion

Aerospike's superior performance and cost efficiency, as demonstrated in this report, have significant implications for organizations handling large-scale data. With its low latency and high throughput, Aerospike enables faster data processing than alternatives, allowing businesses to gain insights and make decisions more quickly. This can be particularly beneficial for applications that require real-time data processing and analytics.

Aerospike's cost-effectiveness can also lead to substantial savings on infrastructure costs, freeing up resources for innovation and growth. Aerospike’s resilience is notable for a self-managed solution. For organizations with rapidly growing data sets, Aerospike's performance and cost efficiency make it an attractive option. It is a gem for those familiar with its capabilities.

Appendix

Pricing and infrastructure details

Data Size: 1 TB			
Platform	**Aerospike**	**DynamoDB w/DAX**	**DynamoDB**
Instance type	c7gd.4xlarge	dax.r5.8xlarge	
vCPU per node	16		
Memory per node (GB)	32	256	
SSDs per node (GB)	1x950	-	
Node count	5	2	1
Instance cost per hour, on demand	$0.7260	$4.295	
Total per hour, all nodes	$3.630	$6,526.70	
EBS disks/ DynamoDB storage	gp2 20GB 100iops	1024 GB	1024 GB
Disk per month	$2.00	$6,245.15	$6,501.15
Total disks/mo	$10.00	$6,245.15	$6,501.15
Per month, on demand	$2,659.17	$12,771.85	$6,501.15
Provisioned reads/sec	n/a	70,000	70,000
Provisioned writes/sec	n/a	30,000	30,000
Total upfront cost (1 yr reserved capacity)	n/a	$100,500.00	$100,500.00
Per year, on demand	$31,910	$153,262	$78,014
1-Year, all upfront	$18,818	n/a	n/a
Total cost (lower option)	**$18,818**	**$253,762**	**$178,514**
Relative cost to Aerospike	-	*13.48x*	*9.49x*

Data Size: 5 TB			
Platform	**Aerospike**	**DynamoDB w/DAX**	**DynamoDB**
Instance type	i8g.4xlarge	dax.r5.8xlarge	
vCPU per node	16		
Memory per node (GB)	128	256	
SSDs per node (GB)	1x3750	-	
Node count	7	10	1
Instance cost per hour, on demand	$1.373	$4.295	
Total per hour	$9.61	$32,633.50	
EBS disks/ DynamoDB storage	gp2 20GB 100iops	5120 GB	5120 GB
Disk per month	$2.000	$6,245.15	$7,525.15
Total disks/mo	$14.00	$6,245.15	$7,525.15
Per month, on demand	$7,026.21	$38,878.65	$7,525.15
Provisioned reads/sec	n/a	70,000	70,000
Provisioned writes/sec	n/a	30,000	30,000
Total upfront cost (1 yr reserved capacity)	n/a	$100,500.00	$100,500.00
Per year, on demand	$84,314.52	$466,543.80	$90,301.80
1-Year all upfront	$51,106.52	n/a	n/a
Total cost (lower option)	**$51,107**	**$567,044**	**$190,802**
Relative cost to Aerospike	-	*11.1x*	*3.73x*

Data Size: 10 TB			
Platform	**Aerospike**	**DynamoDB w/DAX**	**DynamoDB**
Instance type	i8g.4xlarge	dax.r5.16xlarge	
vCPU per node	16		
Memory per node (GB)	128	512	
SSDs per node (GB)	1x3750	-	
Node count	14	10	1
Instance cost per hour, on demand	$1.373	$8.137	
Total Per Month	$19.22	$72,435.60	
EBS disks/ DynamoDB storage	gp2 20GB 100iops	10240 GB	10240 GB
Disk per month	$1.60	$6,245.15	$8,805.15
Total disks/mo	$22.40	$6,245.15	$8,805.15
Per month, on demand	$14,052.42	$78,680.75	$8,805.15
Provisioned reads/sec	n/a	70,000	70,000
Provisioned writes/sec	n/a	30,000	30,000
Total upfront cost (1 yr reserved capacity)	n/a	$100,500.00	$100,500.00
Per Year, on demand	$168,629.04	$944,169.00	$105,661.80
1-Year all upfront	$102,145.85	n/a	n/a
Total cost (lower option)	**$102,146**	**$1,044,669**	**$206,162**
Relative cost to Aerospike	-	*10.23x*	*2.02x*

About Aerospike

Aerospike is the real-time database for mission-critical use cases and workloads, including machine learning, generative, and agentic AI. Aerospike powers millions of transactions per second with millisecond latency, at a fraction of the cost of other databases. Global leaders, including Adobe, Airtel, Barclays, Criteo, DBS Bank, Experian, HDFC Bank, PayPal, Sony Interactive Entertainment, and Wayfair rely on Aerospike for customer 360, fraud detection, real-time bidding, and other use cases. Headquartered in Mountain View, California, our offices are also located in London, Bangalore, and Tel Aviv.

About McKnight Consulting Group

Information Management is all about enabling an organization to have data in the best place to succeed to meet company goals. Mature data practices can integrate an entire organization across all core functions. Proper integration of that data facilitates the flow of information throughout the organization which allows for better decisions – made faster and with fewer errors. In short, well-done data can yield a better run company flush with real-time information... and with less costs.

However, before those benefits can be realized, a company must go through the business transformation of an implementation and systems integration. For many that have been involved in those types of projects in the past – data warehousing, master data, big data, analytics - the path toward a successful implementation and integration can seem never-ending at times and almost unachievable. Not so with McKnight Consulting Group (MCG) as your integration partner, because MCG has successfully implemented data solutions for our clients for over a decade. We understand the critical importance of setting clear, realistic expectations up front and ensuring that time-to-value is achieved quickly.

MCG has helped over 100 clients with analytics, big data, master data management and "all data" strategies and implementations across a variety of industries and worldwide locations. MCG offers flexible implementation methodologies that will fit the deployment model of your choice. The best methodologies, the best talent in the industry and a leadership team committed to client success makes MCG the right choice to help lead your project.

MCG, led by industry leader William McKnight, has deep data experience in a variety of industries that will enable your business to incorporate best practices while implementing leading technology. See www.mcknightcg.com.

Disclaimer

McKnight Consulting Group (MCG) runs all its tests to strict ethical standards. The results of the report are the objective and unbiased results of the application of queries to the simulations described in the report. The report clearly defines the selected criteria and process used to establish the field test. The report also clearly states the data set sizes, the platforms, the methods, etc. that were used. The reader is left to determine for themselves how to qualify the information for their individual needs. The report does not make any claims regarding third-party certification and presents the objective results received from the application of the process to the criteria as described in the report. The report strictly measures performance and cost and does not purport to evaluate other factors that potential customers may find relevant when making a purchase decision. This is a sponsored report. The client chose its configuration, while MCG chose the test, configured the database and testing application, and ran the tests. MCG also chose the most compatible configurations for the other tested platforms. Choosing compatible configurations is subject to judgment. The information necessary to replicate this test is included. Readers are encouraged to compile their own representative configuration and test it for themselves.