



WHITE PAPER

Comparing the cloud costs and performance of Redpanda vs. Apache Kafka®

Redpanda is at least 10x faster and saves up to 6x in total cloud costs



Redpanda: Better performance at (significantly) lower costs	1
Performance Benchmark: Redpanda is at least 10x faster than Apache Kafka	2
500 MB/sec results.....	3
1 GB/sec results	4
What Redpanda’s performance means for you	6
Cloud cost analysis: Redpanda is up to 6x more cost effective than Kafka	7
Cloud infrastructure costs	7
Cluster management costs.....	8
Total combined compute, storage, and management costs.....	10
Additional cost savings with Redpanda Enterprise	10
Conclusion	14

Redpanda: Better performance at (significantly) lower costs

The future of data is speed at scale. There's a growing need for efficiently moving gigabytes of real-time data for complex and instant transaction processing, AI/ML, IoT sensor monitoring, real-time API calls, event-driven apps, and so much more.

In the digital business economy, keeping latencies consistent and low at scale is critical. The latency and volume of data that can be processed impacts both the user experience, as well as the value and competitiveness of your business. In fact, Google found that increasing latency from 100 to just 400 milliseconds resulted in a traffic drop of 0.2% to 0.6%.¹

However, with great speed comes great cost—or at least it used to.

Redpanda is a Kafka-API compatible streaming data platform designed to maximize hardware usage and deliver the fastest performance possible. It's free from ZooKeeper® and JVMs, and builds-in everything you need to operate the platform in a single binary—like schema registry, HTTP proxy, etc. That makes it dependency free and ideal for local development/CI/CD, as well as simple to deploy and manage in production.

Written from scratch in C++ using a brand new architecture, Redpanda optimizes for low latency without data loss and significantly reduces your costs over legacy Kafka options.

To understand how Redpanda and Apache Kafka compare, we ran an independent performance benchmark. In our results, Redpanda processed data at least **10x faster**, using less infrastructure, as well as fewer deployment and administration requirements.

This means Redpanda is up to **6x more cost effective** than Kafka. For you, that means reduced cloud spend, significantly less ongoing maintenance and support, and the opportunity to offer a distinct business advantage.

In this paper, we explore the performance and cost savings of Redpanda over Kafka, so you can see for yourself what Redpanda's resource-efficiency could mean for your business.

Redpanda vs. Kafka – Key total cost of ownership stats

- Up to 10x faster tail latencies
- 6x more cost effective on large workloads
- \$552,298 in infra savings at 1 GB/sec

¹ Google, Speed Matters. 2009.

Performance Benchmark: Redpanda is at least 10x faster than Apache Kafka

In our benchmark, we looked at the end-to-end latency of the source-available Redpanda Community edition and Apache Kafka 3.2, under workloads of up to 1 GB/sec.² We compared the average latencies as well as the 99.99th percentiles (p99.99) to understand how both systems fare with identical hardware and configuration settings. We also ran additional tests where we tripled the number of Kafka nodes to see the impact of increasing its hardware capacity.

We used 500 MB/sec and 1 GB/sec as representative workloads. These values refer to the write throughput, and we assumed a 1:1 read to write ratio, so the total throughput of each workload can be effectively doubled. We chose these workloads simply because they're similar to many of our customers' environments. See Figure 1.

Size	Write Throughput	Total Throughput	Configuration	Instance Types
Medium	500 MB/sec (500,000 * 1KB messages per second)	1 GB/sec (1:1 read/write ratio)	1 topic, 144 partitions, 4 producers, 4 consumers	i3en.3xlarge 12vCPU, 96GiB RAM, 1 x 7.5TB NVMe, Up to 25 Gbps networking
Large	1 GB/sec (1,000,000 * 1KB messages per second)	2 GB/sec (1:1 read/write ratio)	1 topic, 288 partitions, 4 producers, 4 consumers	i3en.6xlarge 24vCPU, 192GiB RAM, 2 x 7.5TB NVMe, 25 Gbps networking

Figure 1 - Details of various workloads used in testing.

More details on our benchmarks

Unlike other published benchmarks, all of our tests ran with TLS and SASL-SCRAM enabled, as this is the default for most production workloads. Our goal was to highlight real-world usage, and in the era of GDPR, CCPA, PCI, and HIPAA, we find that most of our users run with security features enabled.

² We used The [Linux Foundation's OpenMessaging Benchmark](#), including a number of changes that were introduced by Confluent two years ago and more recent improvements such as avoiding the [Coordinated Omission problem](#) of incorrect timestamp accounting. For each test, we did three runs of each workload, each with a 30-minute warm-up. For the clients we ran on four [m5n.8xlarge](#) instances, which ensured guaranteed 25Gbps network bandwidth with 128GB of RAM and 32 vCPUs to ensure our clients were not the bottleneck. We used Kafka v3.2.0 and Redpanda v22.2.2 throughout. We devised three workloads based on increasing throughput and a partition count based on the number of CPUs in our target instance types. We used 50 MB/sec, 500 MB/sec, and 1 GB/sec as representative workloads. These values indicate the write throughput and we have assumed a 1:1 read to write ratio, so the total throughput of each workload can be effectively doubled. We regularly work with customers who have workloads within these ranges and often above as well. For these tests we measure end-to-end throughput and end-to-end latency, with 2 producers writing and 2 consumers reading from a single topic across a number of partitions. Consistency in end-to-end latency is important for ensuring that applications are able to meet SLAs at scale. All tests were conducted on AWS, with identical instance types for running both Kafka and Redpanda in each test.

500 MB/sec results

At 500 MB/sec, Redpanda Community was able to easily handle the workload with just three nodes. When we ran the same workload on Kafka, it couldn't sustain the publish rate with just three nodes. So, we re-ran the workload on Kafka with up to nine nodes to evaluate whether we could bring Kafka's throughput inline with Redpanda's, but we were unsuccessful.

In comparing tail latencies, shown in Figure 2 below, a 3-node Redpanda Community cluster was:

- 10x faster against a 4-node Kafka cluster
- 4x faster against a 9-node Kafka cluster

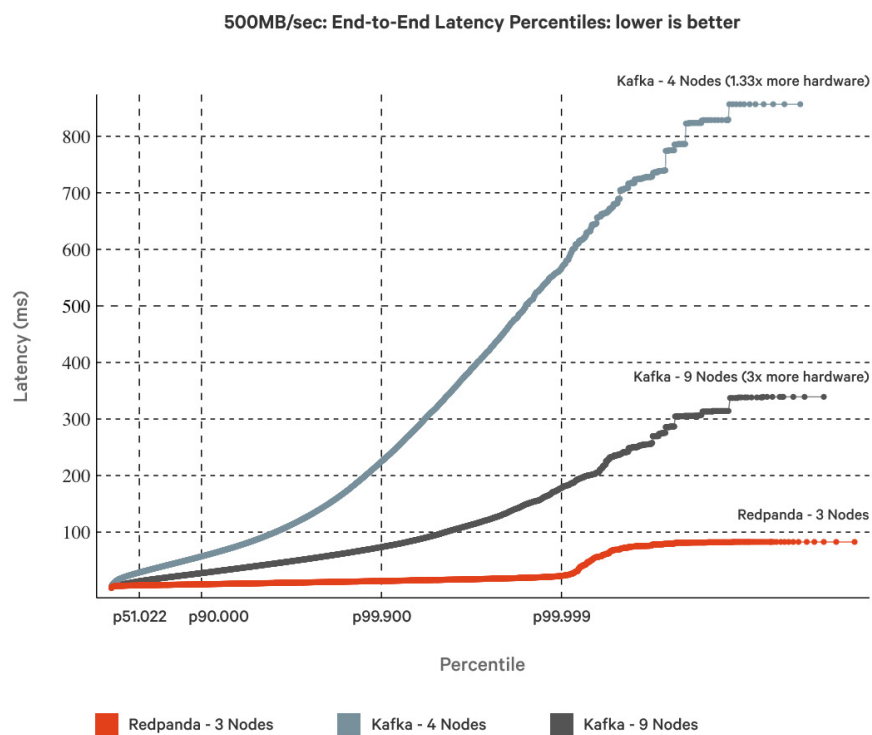


Figure 2 - End-to-end tail latencies of a 9-node and 4-node Kafka cluster vs. a 3-node Redpanda Community cluster, using 500 MB/sec workloads for all three.

In comparing average latencies, shown in Figure 3 below, a 3-node Redpanda Community cluster was:

- 5x faster when Kafka ran with four nodes
- 3x faster when Kafka ran with nine nodes

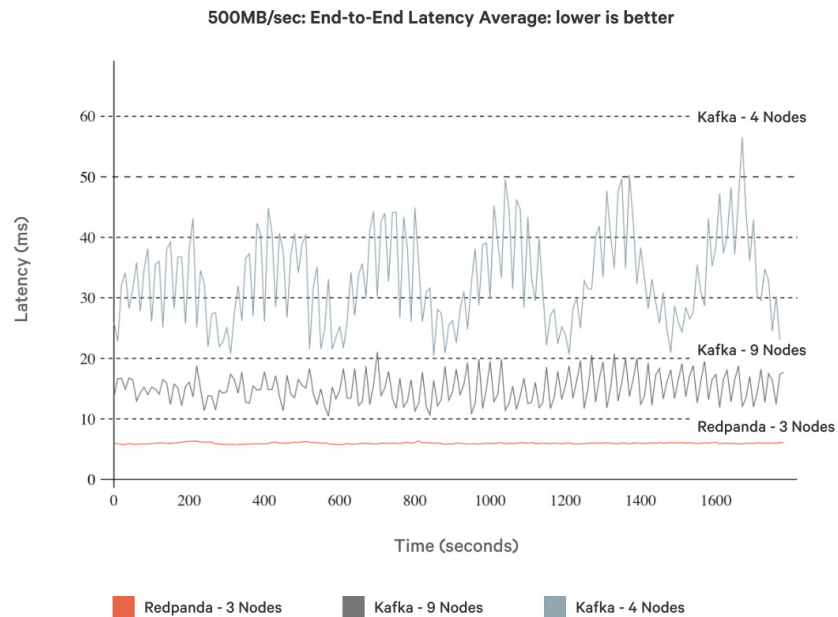


Figure 3 - Average end-to-end latency of a 9-node and 4-node Kafka cluster vs. a 3-node Redpanda Community cluster, using 500 MB/sec workloads for all three.

1 GB/sec results

Here we increased the size of the instance types for the 1 GB/sec workload. Once again, Redpanda Community was able to comfortably sustain this high throughput with only three nodes. On the other hand, Kafka failed to complete the test with three nodes. So we repeated the test multiple times, adding more nodes at every iteration, until Kafka inched closer to Redpanda's performance.

At 1 GB/sec throughput, as shown in Figure 4, the percentile graph for this workload shows that Redpanda Community performed a staggering **70x faster than Kafka** at the tail end with half the amount of hardware. Even with three additional nodes added to Kafka, Redpanda Community still remained 7x faster. In fact, with six additional nodes (total of nine), Kafka latencies were still higher than Redpanda, as shown in Figure 5.

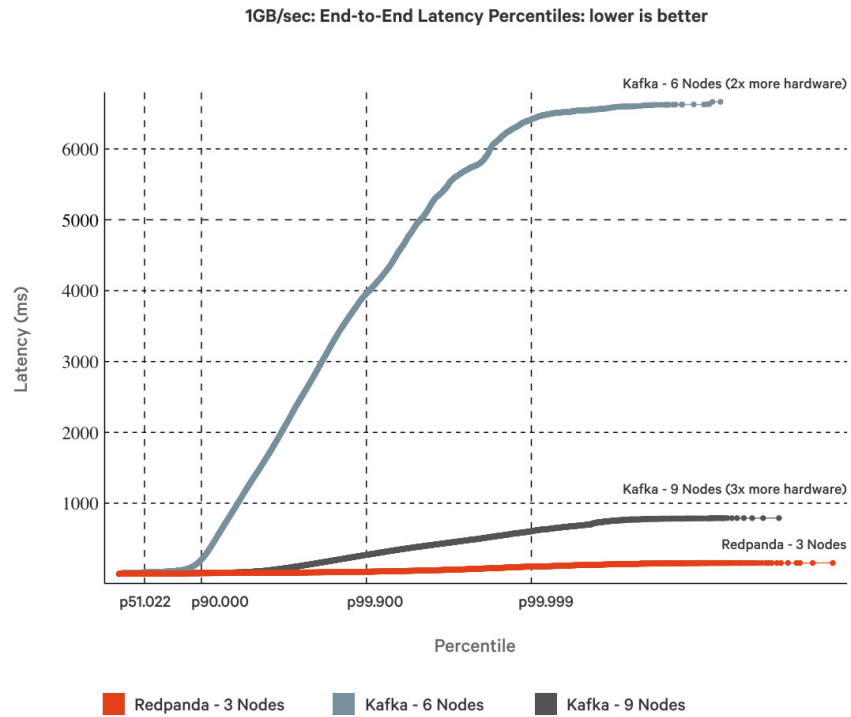


Figure 4 - End-to-end tail latencies of a 9-node and 6-node Kafka cluster vs. a 3-node Redpanda Community cluster, using 1 GB/sec workloads for all three.

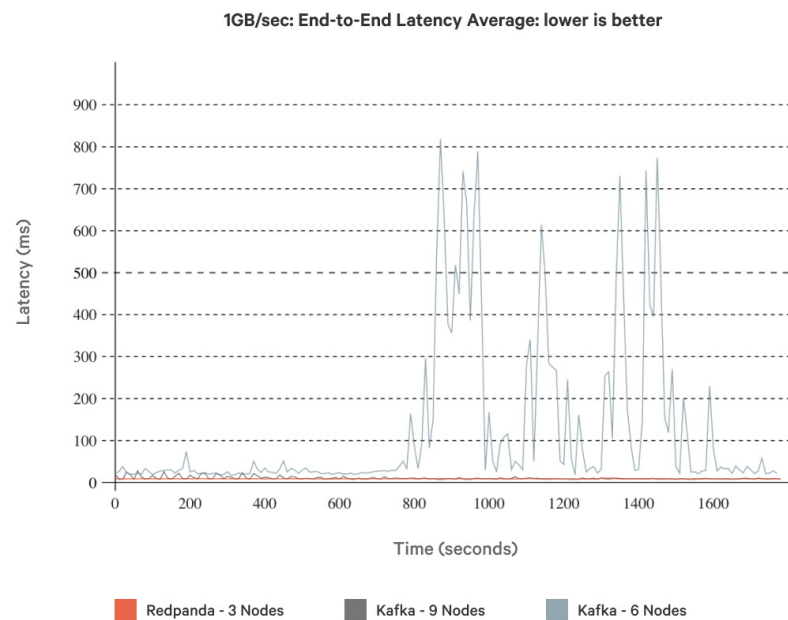


Figure 5 - Average end-to-end latency of a 9-node and 6-node Kafka cluster vs. a 3-node Redpanda Community cluster, using 1 GB/sec workloads for all three.

What Redpanda's performance means for you

The differences between Redpanda's and Kafka's performance are game-changing. With Redpanda, developers can achieve greater consistency and lower latency with far fewer resources. That translates into generous cost savings on things like cloud spend and cluster management, as well as allowing developers to build and innovate with confidence.

Redpanda, unlike Kafka, also has proven reliability semantics so it can be trusted to process your most critical business data. All data is committed to disk on write, meaning that there's no possibility of data loss due to transient failures, as validated by our independent Jepsen testing.

Knowing that your data streaming architecture can reliably meet your performance specifications unlocks use cases that weren't previously possible, and reduces latencies that might result in frustrated users or inefficient processes.

Cloud cost analysis: Redpanda is up to 6x more cost effective than Kafka

Redpanda Community maximizes hardware usage, reduces infrastructure needs, cuts cloud spend, and decreases deployment complexity for a total cost of ownership that’s significantly lower than Kafka. In fact, we found that Redpanda Community was 6x more cost effective than Kafka for workloads nearing 1 GB/s in throughput.

To thoroughly compare costs with Kafka, we’ll look at the following:

- 1. **Cloud infrastructure costs:** The cost of compute and storage, in this case AWS
- 2. **Cluster management costs:** The cost of deploying, installing and upkeep of clusters

Cloud infrastructure costs

Software should be able to make full use of the hardware it’s deployed on. Being able to keep cluster size down affects cloud spend and the resources used. With more companies looking to save money and reduce their carbon footprint, achieving greater hardware efficiency is critical.

When running our performance tests comparing Redpanda to Kafka, we found that Redpanda Community’s average and P99+ end-to-end latency profiles remain incredibly consistent even at high throughputs. In comparison, Kafka could not handle workloads at 500 MB/sec or above (1 GB/sec total throughput) with just three nodes.

We had to repeatedly create bigger Kafka clusters to keep latency profiles flat, as shown in Figure 6, but even with additional clusters, Kafka’s P99.9 latencies were above 200ms at 3x the cluster size of Redpanda. For smaller workloads, Redpanda Community was able to run slightly faster on the cheaper AWS Graviton (ARM) CPUs, whereas Kafka was unable to operate on these instance types at any level of performance.

Workload	Target P99.9 Latency	Kafka Infra Requirement		Redpanda Infra Requirement	
		Nodes	Latency	Nodes	Latency
500 MB/sec	< 20 ms	9 (i3en.3xlarge)	73.61ms	3 (i3en.3xlarge)	10.571ms
1 GB/sec	< 20 ms	9 (i3en.9xlarge)	271.47ms	3 (i3en.6xlarge)	16.216ms

Figure 6 - Comparing infrastructure requirements across medium and large workloads at a target latency profile.

One of the major benefits of running Redpanda Community is simplicity of deployment. Because Redpanda is deployed as a single binary with no external dependencies, **we don't need any infrastructure for ZooKeeper or for a Schema Registry**. Redpanda also includes automatic partition and leader balancing capabilities so there's no need to run Cruise Control. That greatly reduces the time spent managing the application, allowing teams to spend their time working on their product instead of their data streaming architecture.

Companies can expect to see cost savings of between \$81,026 and \$152,298 per use case depending on the size and scale of the workload, as shown in Figure 7. That represents up to a **3x cost saving** against Kafka, just on the infrastructure alone. For workloads larger than 1 GB/sec, the savings are likely to be closer to 6x or higher.

	500 MB/sec		1 GB/sec	
	Kafka	Redpanda	Kafka	Redpanda
Instance Size	i3en.3xlarge	i3en.3xlarge	i3en.6xlarge	i3en.6xlarge
Nodes required for throughput	9	3	9	3
EC2 Broker Cost / month	\$8,909	\$2,970	\$17,818	\$5,939
Auxiliary Instance Type	t2.xlarge		t2.xlarge	
ZK + SR + CC Cost / month (6 aux nodes)	\$813	\$0	\$813	\$0
Annual Costs	\$116,662	\$35,636	\$223,569	\$71,271
Times more expensive than Redpanda	3		3	
Difference	\$81,026		\$152,298	

Figure 7 - Infrastructure cost comparison for 500 MB/sec and 1 GB/sec workload between Kafka and Redpanda Community.

Cluster management costs

Redpanda is designed for usability and simplicity. Since Redpanda doesn't need a JVM or ZooKeeper, users can greatly reduce the amount of monitoring and tuning required for a Redpanda Community cluster, compared to an equivalent Kafka cluster. Redpanda is also designed with data safety in mind, as highlighted in [this report from Jepsen](#), cutting costs of the operations and management overhead of running a Redpanda cluster.

In building cost comparisons for Redpanda Community against Apache Kafka, we interviewed our customers to learn how they have simplified their operational demands since adopting Redpanda. Overall, they spend less time balancing partitions, tuning the JVM, ZooKeeper or the operating systems, and recovering from outages caused by ISR problems.

Based on this data, running a 3-node Redpanda Community cluster at medium and large instance sizes doesn't increase the operational complexity and can be done by an ops team that might be managing other platforms simultaneously. Meanwhile, running a 9-node Kafka cluster, plus three ZooKeeper nodes at high throughputs, is a significantly more complex undertaking, with frequent potential outages and maintenance, as shown in Figure 8. Kafka is also much more likely to require manual intervention on a regular basis.

	500 MB/sec		1 GB/sec	
	Kafka	Redpanda	Kafka	Redpanda
Node Count	9	3	9	3
FTEs	1.4	0.3	2.8	0.3
FTE Cost	\$160,000	\$160,000	\$160,000	\$160,000
Total Team Cost	\$224,000	\$48,000	\$448,000	\$48,000

Figure 8 - SRE team cost comparison for 500 MB/sec and 1 GB/sec workload between Kafka and Redpanda Community.

How Redpanda enables lower cluster management costs

1. **Autotuner** – Auto detects the optimum settings for your hardware and tunes itself.
2. **Leadership balancing** – Improves cluster performance by ensuring that leadership is spread amongst nodes.
3. **Continuous Data Balancing** – Automatically moves data from nodes that are running low on disk or on node failure, to ensure that performance is maintained throughout the cluster,
4. **Maintenance mode** – Allows graceful decommissioning of nodes by transferring leadership onto other nodes ahead of a shutdown.
5. **Rolling upgrades** – Upgrades the cluster without any interruption to consumers or producers.

Total combined compute, storage, and management costs

The differences between Redpanda's total cloud and management costs and Kafka's are stark, as shown in Figure 9 below. The cost of infrastructure alone for **running Kafka can be 3x more expensive** than running Redpanda Community. For larger and more complex workloads, that number can rise to 6x or even higher.

Depending on your use case, that represents money that could be better spent elsewhere, and time that could be used to focus on your core product rather than your infrastructure and maintenance.

		500 MB/sec	1 GB/sec
Redpanda	Prime Cluster Infrastructure	\$35,636	\$71,271
	Admin Costs (FTE @ 130K)	\$48,000	\$48,000
	Total	\$83,636	\$119,271
Kafka	Prime Cluster Infrastructure	\$116,662	\$223,569
	Admin Costs (FTE @ 130K)	\$224,000	\$448,000
	Total	\$340,662	\$671,569
Differential	Prime Infra	3x	3x
	TCO	4x	6x
	Cost Savings	\$257,026	\$552,298

Figure 9 - Consolidated cloud and maintenance cost comparison of Kafka and Redpanda Community edition at 500 MB/sec and 1 GB/sec across all workloads.

All of the prices above compare Kafka with Redpanda Community edition. **According to this model, savings in infrastructure and administrative costs can range from \$257,026 for a medium workload to \$552,298 for large workloads, a factor of 6x.**

Additional cost savings with Redpanda Enterprise

While we've primarily discussed our Redpanda Community offering, Redpanda Enterprise also brings a number of features designed to make operating clusters easier. With Redpanda Enterprise's tiered storage, we deliver infrastructure savings of between \$70,000 and \$1.2 million depending on the workload and size of the cluster, as shown in Figures 10 and 11. That's an **administration and infrastructure savings of 8x to 9x.**

500 MB/sec

	Apache Kafka	Redpanda Community	Commercial Kafka	Redpanda Enterprise
Node Count for throughput	9	3	9	3
Node Type	i3en.3xlarge	i3en.3xlarge	i3en.3xlarge	i3en.3xlarge
Tiered Storage Available	No	No	Yes	Yes
Additional Nodes per day of retention	17.28	17.28	N/A	N/A
Annual EC2 cost per additional node	\$11,878.56	\$11,878.56	\$11,878.56	\$11,878.56
Annual S3 Cost per day of retention	N/A	N/A	\$1,036.80	\$1,036.80
Additional Annual Cost per day of retention	\$2,463,138.20	\$2,463,138.20	\$124,416.00	\$124,416.00
Total Nodes for 3 days retention	52	52	9	3
Total Annual Cost with 3 day of retention	\$627,440.26	\$617,685.12	\$147,766.18	\$66,739.68
Total Nodes for 7 days retention	121	121	9	3
Total Annual Cost with 7 days of retention	\$1,447,060.90	\$1,437,305.76	\$189,238.18	\$108,211.68

Figure 10 - Annual infrastructure cost comparison for three-day retention for 500 MB/sec workload (comparing Kafka, Redpanda Community, Commercial Kafka, and Redpanda Enterprise).

1 GB/sec

	Apache Kafka	Redpanda Community	Commercial Kafka	Redpanda Enterprise
Node Count for throughput	9	3	9	9
Node Type	i3en.6xlarge	i3en.6xlarge	i3en.6xlarge	i3en.6xlarge
Tiered Storage Available	No	No	Yes	Yes
Additional Nodes per day of retention	17.28	17.28	N/A	N/A
Annual EC2 cost per additional node	\$23,757.12	\$23,757.12	\$23,757.12	\$23,757.12
Annual S3 Cost per day of retention	N/A	N/A	\$20,736.00	\$20,736.00
Additional Annual Cost per day of retention	\$4,926,276.40	\$4,926,276.40	\$248,832.00	\$248,832.00
Total Nodes for 3 days retention	52	52	9	3
Total Annual Cost with 3 day of retention	\$1,245,125.38	\$1,235,370.24	\$285,777.22	\$133,479.36
Total Nodes for 7 days retention	121	121	9	3
Total Annual Cost with 7 days of retention	\$2,884,366.66	\$2,874,611.52	\$368,721.22	\$216,423.36

Figure 11 - Annual infrastructure cost comparison for three-day retention for 1 GB/sec workload (comparing Kafka, Redpanda Community, Commercial Kafka, and Redpanda Enterprise).

In Figure 12, we show the incremental retention costs on clusters without tiered storage can be quite significant across both of the workloads. These numbers are not accounting for the indirect values of Redpanda Enterprise features such as Redpanda Console with SSO and RBAC, remote read replicas, continuous data balancing, and hot-patching.

Workload	Base Costs (excl. licensing) - 3 days retention, Redpanda Enterprise	Cost savings of Redpanda Enterprise over Commercial Kafka	Cost savings of Redpanda Enterprise over Redpanda Community	Cost savings of Redpanda Enterprise over Apache Kafka
500 MB/sec	\$66,739.68	\$81,026.50	\$550,945.44	\$560,700.58
1 GB/sec	\$133,479.36	\$152,297.86	\$1,101,890.88	\$1,111,646.02

Figure 12 - Summary incremental cost savings of Redpanda Enterprise over Kafka (infrastructure costs only).

Conclusion: Redpanda Community is at least 10x faster and reduces your cloud and maintenance costs by up to 6x

Redpanda Community doesn't just outperform Kafka at 10x the speed on tail latencies, it's also between 3x to 6x more cost effective than running the equivalent Apache Kafka infrastructure and team.

While we would've loved to compare Redpanda's and Kafka's performance on equal resources, Kafka couldn't deliver the medium and large workloads without additional hardware. On the same hardware, Kafka simply could not sustain the same throughput.

In short: Redpanda provides companies with vastly improved performance at significantly lower cloud and maintenance costs. With data systems increasing in complexity and expense, Redpanda is innovating to help you maximize hardware usage, reduce cloud spend, and reduce the complexity of deployment.

With Redpanda's speed and low cloud and management costs, you could open the door to new use cases that were previously thought impossible or too expensive. Take [Alpaca](#) for example, which boosted their performance by 100x by rearchitecting their order management platform around Redpanda.

Moreover, Redpanda enables companies to achieve this type of performance with the smallest hardware footprint possible in the market. One example is [LiveRamp](#), which can now process tens of terabytes of data per day in near real-time by migrating to Redpanda while significantly lowering their infrastructure cost and carbon footprint.

Visit us at redpanda.com to learn more.

What will **you** build with Redpanda?

Learn more

To find out more about Redpanda, please contact us or join our community. We're glad to meet with you to show the advantages Redpanda can bring to your company.

Website: redpanda.com

Documentation: docs.redpanda.com

Slack: <https://redpanda.com/slack>

Twitter: [@redpandadata](https://twitter.com/redpandadata)

Contact us: hi@redpanda.com

Github: github.com/redpanda-data/redpanda