Micron® 5100 ECO SSD: Better, Faster Analytics with Smaller Cassandra Clusters

5100 ECO[1] Cassandra Performance

YCSB Workloads vs. all-HDD cluster[2]

Workload A: 6.1x
Workload B: 6.0x
Workload C: 6.4x
Workload D: 9.1x
Workload F: 5.6x

More detailed iterative analytics, smaller clusters

1. 5100 ECO 2.5" 1.92TB tested. Other capacities and classes are available.

2. 5100 ECO advantage based on loading range. For each load, we divided 5100 ECO operation/second by the HDD value. The range shown is the average from 48 to 240 threads.

Overview

Apache Cassandra™ (an open-source NoSQL database) is widely adopted in big data analytics, automatic product recommendation systems, online catalog displays, messaging platforms, query analytics and a host of other real-time and near real-time applications.

Cassandra was designed from the ground-up to support a high-performance, node-distributed (single location) or geographically distributed (multiple locations) deployment model.

Cassandra's ability to support massive scale makes it powerful tool as data growth skyrockets and we increasingly depend on associated analytics. Our expectations have grown as well. We want more detailed analysis from the larger data sets – and we want them faster than ever. Legacy platforms built on hard disk drives (HDDs) can't keep up. Cassandra platforms built on our 5100 Enterprise SSDs can.

This technical brief highlights the performance advantages we measured when we compared two 4-node Cassandra clusters – one built with HDDs (legacy), the other build with our 5100 ECO Enterprise SSD**.** We also explore some implications of these results.

Due to the broad range of Cassandra deployments, we tested multiple workloads and multiple thread counts. You may find some results more relevant than others for your deployment.

Industry Standard Tests

Table 1 shows the configurations we compared, measuring Cassandra cluster database operations per second (OPS) using the Yahoo Cloud Serving Benchmark3 (YCSB). We measured several common Cassandra workloads (YCSB workloads A–D and F) across a wide range of common thread counts, reflecting common use and performance characterization.

Cassandra Cluster Node	Database Storage	System Boot	Nodes per Cluster
5100 ECO (SSD)	2x 1.92TB (mdadm RAID 0)	1x SSD (240GB)	4
10K RPM SAS (HDD)	2x 1.6TB (mdadm RAID 0)	1x SSD (240GB)	4

Table 1: Tested Storage Configurations

In the sections below, we organized performance results by workload (A–D and F), and within each workload section measured Cassandra performance in OPS for each configuration as well as the associated ratio (5100 ECO OPS) / (HDD OPS). In each section header the 5100 ECO OPS advantage is the *average across thread counts*.

Session Action/Recording: 6.1x Better

Workload A is an update-heavy workload, with 50% of the total I/Os writing data. At the application level, this workload is similar to recording recent session actions.

Figure 1 shows the cluster performance (in OPS) along the vertical axis. The HDD configuration data is in red and the 5100 ECO data is in green. In Figure 1, and all subsequent figures, taller is better as it shows superior performance.

The thread count ranges from 48 to 240 along the horizontal axis, the ratio of 5100 ECO / HDD OPS is shown for each thread count (circled in blue). Subsequent figures have a similar layout.

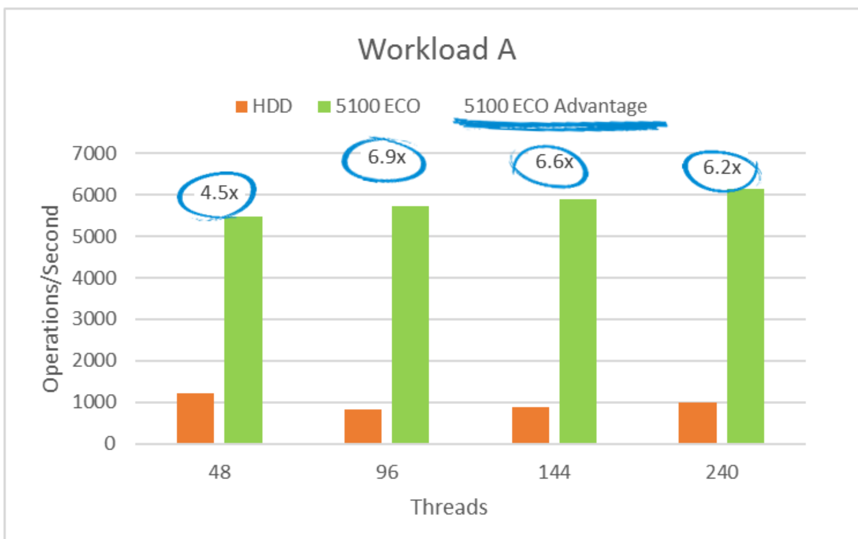

Figure 1: Workload A

The 5100 ECO configuration completed between 5470 and slightly more than 6100 database operations per second. The OPS ratio ranged between 4.5x and 6.9x, averaging 6.1x.

Adding Metadata/Tags: 6.0x Better

Workload B is an update-light, read-mostly workload with 5% of the total I/Os writing data. At the application level, this workload is similar to [tagging photographs and articles or adding information about videos and music.]

With this update-light workload, the 5100 ECO configuration completed between 4400 and slightly more than 4900 database operations per second. The OPS ratio ranged between 5.6x and 6.7x, averaging 6.0x.

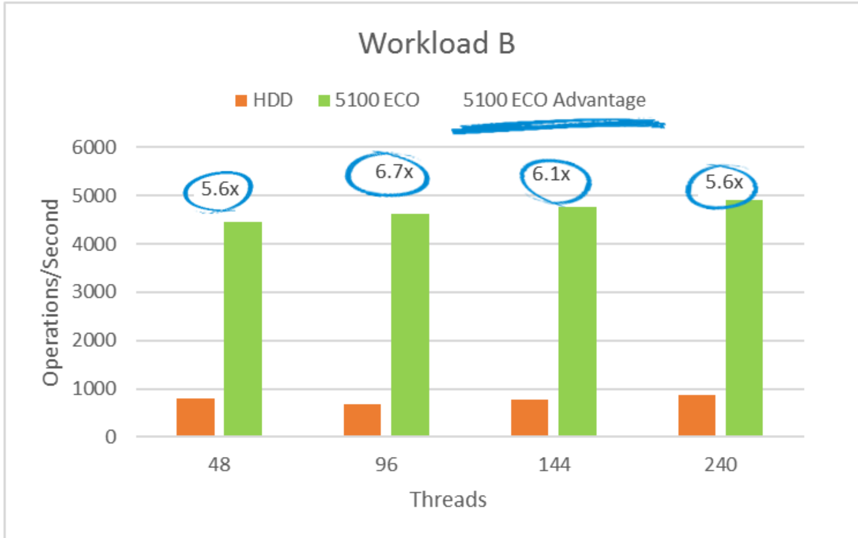

Figure 2: Workload B

Static Data Cache: 6.4x Better

Workload C is a read-only workload (100% of the total I/Os read data; there is no write traffic). At the application level, this workload is similar to [reading user profiles or other static data where profiles are constructed elsewhere].

The 5100 ECO configuration completed between 4,800 and slightly more than 5,300 database operations per second. The OPS ratio ranged between 5.0x and 7.3x, averaging 6.4x.

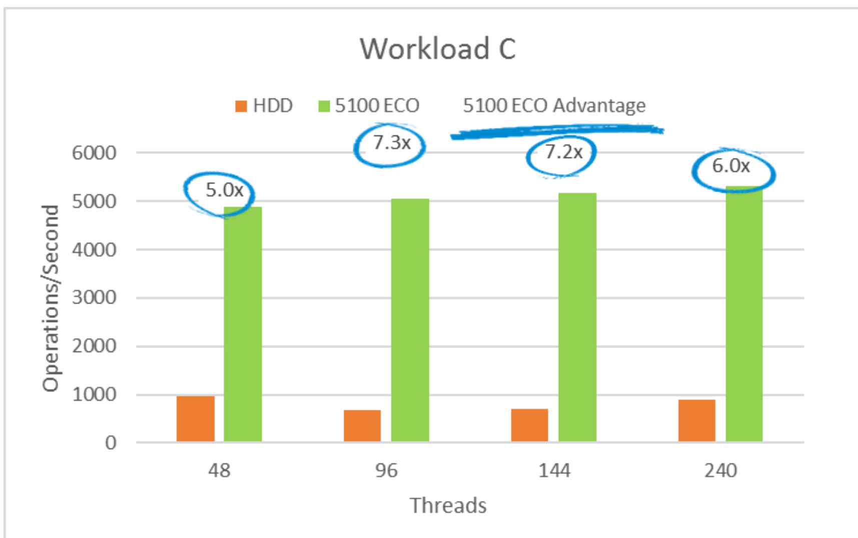

Figure 3: Workload C

Recent Statistics Tracking: 9.1x Better

Workload D reads the latest entries (most recent records are the most popular). At the application level, this workload is similar to reading user status updates (where users are likely to read the most recent entries). Examples of this workload include social media, frequently changing or updated product literature, or software development repositories.

The 5100 ECO cluster completed between 15,400 and 17,700 database operations per second. The OPS ratio ranged between 8.1x and 9.8x, averaging 9.1x.

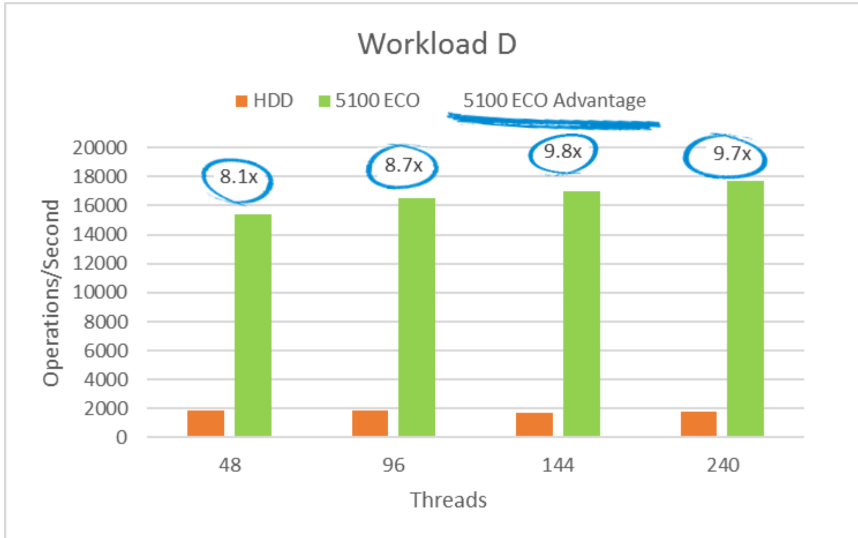

Figure 4: Workload D

User Record Changes: 5.6x Better

Workload F is a read/modify/write workload in which records are read, changed and written back. At the application level, this workload is similar to users reading and changing data or tracking user activity.

The 5100 ECO cluster completed between 4900 and 5300 database operations per second. The OPS ratio ranged between 5.0x and 6.2x, averaging 5.6x.

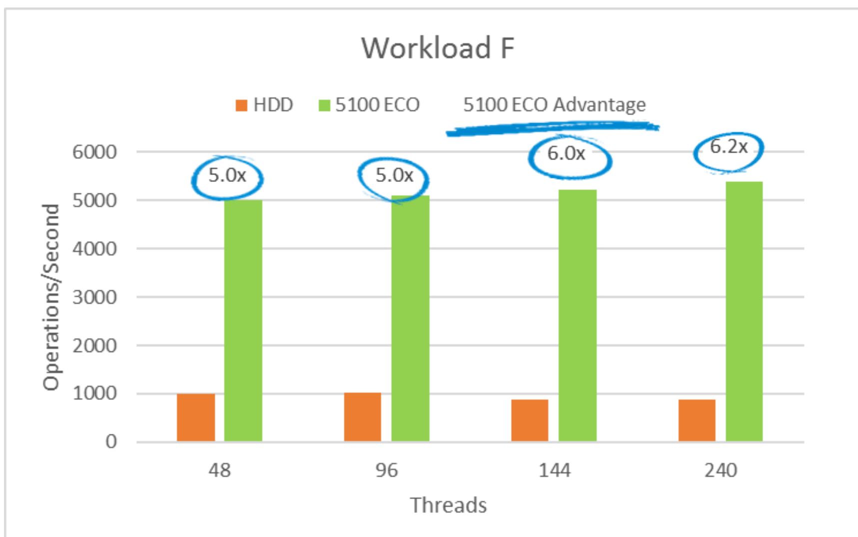

Figure 5: Workload F

Implications

Workload OPS comparisons are impressive, with the 5100 ECO configuration exceeding the HDD configuration several times over for each workload and thread count tested. While the raw OPS performance of the 5100 ECO configuration is especially surprising given that typical Cassandra accesses are well-aligned to HDD ideals (large transfers), it can have real world implications.

More Detailed Iterative Analysis

When our cluster executes more OPS, we can complete first pass, simple analysis faster, freeing up cluster resources so we can take a second, deeper look at the areas of most interest. Alternatively, we can increase our first pass analysis depth and complexity for (potentially) better results first time.

Reduce Cluster Sprawl

For large, busy clusters, better OPS can have a real impact on cluster size. For example, if we have a data set to process within a certain timeframe (such as a report), we can see how OPS per node can affect the size of a cluster.

If we knew we could process the data in time with an existing cluster the size of one rack and we increase the OPS of each node, then we could process the same data in the same time with fewer nodes. Increasing the OPS per node by 2x should meet our processing needs with a cluster that's only half a rack. Increasing OPS per node by 6x should process the same data in a cluster that's only 1/6th of a rack.

This is only an example and actual results would be affected by other inputs — but it is a good example of how better OPS per node can have a real effect on our footprint.

The Bottom Line

When we looked at the standard benchmark performance advantages across multiple workloads and thread counts, we saw that a 5100 ECO-based 4-node cluster eclipsed the capability of a similar HDD-based configuration.

The higher OPS of the 5100 ECO configuration can have a real impact: we can get more detailed first-pass analysis or we can use iterative, multi-pass analysis on the same dataset with better results. We can even contemplate processing data sets on smaller clusters despite huge data growth and complexity.

Because Cassandra supports massive scale, the extra capabilities of the 5100 ECO can help you manage skyrocketing data grown and ever-increasing demand.