

INSIDER

## DataStax review: Cassandra made faster and easier

DataStax Enterprise 6 offers greater throughput than Cassandra, lower latency, and useful extras that make it easier to run

By **Martin Heller**

Contributing Editor, InfoWorld | AUG 20, 2018

### AT A GLANCE

DataStax Enterprise 6



As I discussed in my review of Google Cloud Bigtable in 2016, Google's 2006 Bigtable paper inspired several large-scale distributed open source NoSQL databases, including Apache HBase and Apache Cassandra. I went on to explain that Cassandra was born at Facebook using ideas from Bigtable and the key-value store Amazon Dynamo, and that while Cassandra is a bit more popular than HBase, has a SQL-like query language (CQL), and is easier to get up and running than HBase, it is still complicated and has a significant learning curve.

Google Cloud Bigtable is one good managed alternative to running your own Cassandra clusters. Others include DataStax Enterprise (DSE) and DataStax Managed Cloud.

Why DataStax? Essentially, DataStax is the supported enterprise version of Cassandra, with improved performance and security, vastly improved management, advanced replication, in-memory OLTP, a bulk loader, tiered storage, search, analytics, and a developer studio. Not coincidentally, DataStax employees have contributed roughly 85 percent of the code in the Apache Cassandra project.

Like Bigtable and Cassandra, DataStax is best suited for large databases—terabytes to petabytes—and is best used with a denormalized schema that has many columns per row. DataStax and Cassandra users tend to use it for very large-scale applications. For example, eBay uses DataStax Enterprise to store 250 TB of auction data with 6 billion writes and 5 billion reads daily. Apple has (or had) more than 75,000 Cassandra nodes storing more than 10 PB of data.

### What is Cassandra?

At its heart, Apache Cassandra is a highly available distributed datastore that values availability and partition tolerance over consistency. The design of Cassandra combines the partitioning and replication of the Amazon Dynamo key-value store with the log-structured, column-family data model of Google Bigtable. Cassandra scales linearly as

you add nodes.

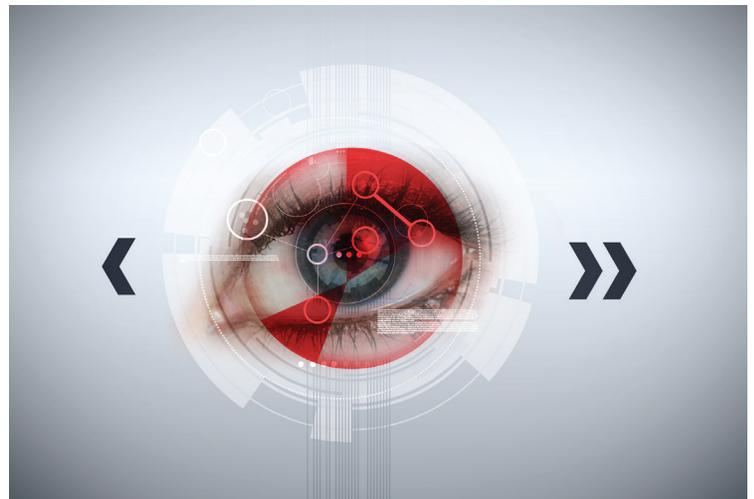
Consistency is not completely lost in Cassandra; it's a trade-off against latency. The user can specify the consistency level of each read and write, ranging from requiring only one node, through requiring a cluster quorum, to requiring all nodes. Another intermediate option is to require a local quorum, which is a way to attain consistency within a data center without waiting for remote nodes to update.

At the physical level, Cassandra architecture organizes nodes into rings of peers, called a data center, and allows multiple data centers to be connected into a cluster. All data is written first to the commit log for durability, then indexed and added to a memtable. When a memtable is full, it is flushed to disk as an SSTable (sorted string table) data file. All writes are automatically partitioned and replicated throughout the cluster up to the configured number of replicas, and according to the configured replication strategy.

At the logical level, Cassandra is a partitioned row store database, where rows are organized into tables with a required primary key; secondary keys are optional. Any authorized user can connect to any node in any data center and access data using the CQL language, which uses a SQL-like syntax and works with table data. Developers can access CQL through the CQL shell (cqlsh), DevCenter, and via drivers for application languages.

Typically, a cluster has one keyspace per application composed of many different tables. Replication is declared at the keyspace level. You can configure consistency for a client session or for an individual read or write operation.

From the beginning, Cassandra supported key-



iStock

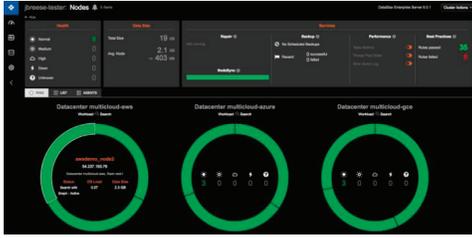
value and columnar data. Later, support for JSON was added, allowing Cassandra to be used as a document database, although CQL schemas are still enforced in JSON inserts, unlike MongoDB, which allows schema-less documents. Columns may be omitted in individual Cassandra table rows without violating the schema—it's a sparse database, like Bigtable.

### What is DataStax Enterprise?

DataStax adds both features and performance to Cassandra. Exactly which features are supported depends on whether you are using the Basic or Enterprise version. Each cluster must be one or the other.

The Basic subscription adds a production-certified Cassandra release; NodeSync and TrafficControl (automated repair and backpressure mechanisms to simplify operations); higher throughput and lower latency; a fast bulk loader; the DataStax Studio developer tool; and support and services. The Enterprise subscription additionally includes automatic lifecycle management services; advanced replication; tiered storage; integrated search based on Apache Solr; integrated analytics based on Apache Spark (but faster); and

OpsCenter (see screenshot below). DSE can be deployed on-premises, in the cloud, or in a hybrid cloud. A graph database layer, DSE Graph, can be added to DataStax Enterprise.



DataStax's OpsCenter can monitor and control your DataStax Enterprise installations across multiple data centers, when in the cloud or on premises.

## What is DataStax Managed Cloud?

The DataStax Managed Cloud is a “white glove” installation of DSE in your AWS or Azure cloud estate. According to DataStax, the managed cloud is a good choice for companies that need help with administration or want to free their DBAs for more interesting work; companies that want to have real-time and geo-distributed data; and when DBA training is becoming a burden.

While Google Cloud Bigtable is a low-touch service that just works, DataStax Managed Cloud is a high-touch service. That suggests that DataStax might not be as robust and mature as Bigtable. Given Bigtable’s history as the database for many of Google’s large-scale services, including web search, that isn’t too surprising.

## What’s new in DSE 6?

Apache Cassandra has been notorious for needing manual repairs to keep its clusters operational and its nodes synced. DSE NodeSync eliminates the need to run repair scripts and eliminates the cluster outages that can occur when manual repairs fail.

DSE 6 boasts advanced performance. According to DataStax, that breaks down into a thread-per-core architecture that improves throughput up to 2x for both read and write operations compared to open source Cassandra; storage engine optimizations that reduce latency by half compared to open source Cassandra; a bulk loader that is up to 4x faster than previous data loading and unloading utilities; and continuous paging that improves DSE Analytics read performance by up to 3x over open source Cassandra and Apache Spark.

DSE TrafficControl automatically keeps DSE nodes from overloading with client or replica requests. The DSE Upgrade Service, part of OpsCenter LifeCycle Manager, significantly reduces manual involvement in patch upgrades.

There are several new features in DSE Analytics, DSE Graph, and DSE Search for DSE 6, along with finer-grained security settings. Improvements to DataStax Studio track the improvements in DSE Analytics, such as support for Spark SQL, and expanded IDE support for DSE Graph with interactive graphs.

## DSE 6 installation

Depending on whether you are installing for production or for development and test, you can install DSE 6 a number of different ways. One way is to install OpsCenter, then use its Lifecycle Manager GUI (see side-by-side screenshots below) to install DSE using RHEL or Debian packages and DataStax agents, assuming that the targets already

have SSH and Python installed.



You can create DataStax Enterprise clusters and their nodes from Lifecycle Manager, within OpsCenter.

Another method is to use Yum (RHEL) or APT (Debian), assuming that you have root permissions—this is often done for production. A third option, and the one promoted by the documentation, is to install from tarballs. Installing on a cloud VM is basically a matter of creating an instance from a supported OS image and then running an installer for that OS from the instance.

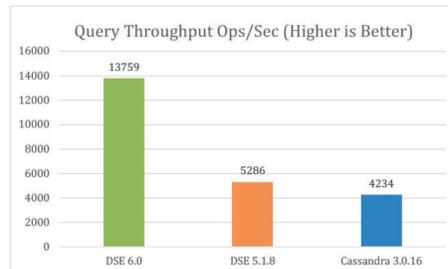
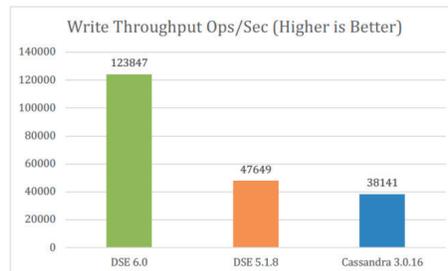
Installing DSE with Docker images is strictly for development and test at this point. It may eventually be certified for production.

DataStax will do all of the installation and configuration for you if you are willing to pay for DataStax Managed Cloud.

## DSE benchmarks

In May 2018 zData benchmarked DataStax Enterprise 6.0 and DataStax Enterprise 5.1 against Apache Cassandra 3.0 on five AWS i3.8xlarge instances as database nodes, and 10 smaller nodes as clients running the cassandra-stress application. zData tested a wide range of read/write ratios and pushed the database to about 75 percent of its capacity, with about 2.5 TB of total data, which is small by Cassandra standards but big enough to benchmark. There is enough information in the benchmark paper to reproduce the tests, although I did not attempt to do so.

For the 90 percent write and 10 percent read workload, designed to be typical of an IoT use case, DSE 6 showed a roughly 3x improvement in throughput over Cassandra, and a 2x improvement over DSE 5:



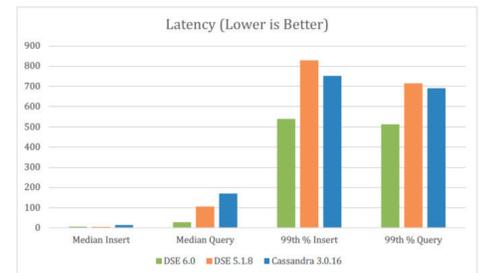
zData’s throughput benchmark results for the case of 90 percent writes and 10 percent reads show DSE 6 multiples faster than DSE 5 and Cassandra 3.

Note that DSE 6 achieved 124K write operations per second for this workload, amounting to roughly 25K write operations per second per node for this

choice of instance size. AWS i3.8xlarge instances are storage-optimized, with 32 vCPUs, 244 GiB of RAM, and four 1900 GB NVMe SSD, and cost \$2.496 per node per hour. That’s about \$60 per day—not including the cost of DSE.

By comparison, for SSD clusters, Google specifies that the typical Cloud Bigtable steady state performance is 10K queries per second per node for both reads and writes, with 6 milliseconds latency. Cloud Bigtable nodes cost \$0.65 per node per hour plus \$0.17 per GB per month. With a comparable volume of SSD storage (4 x 1900 GB), Cloud Bigtable would cost slightly less than \$60 per node per day.

DSE 6 also showed lower latency for this test:



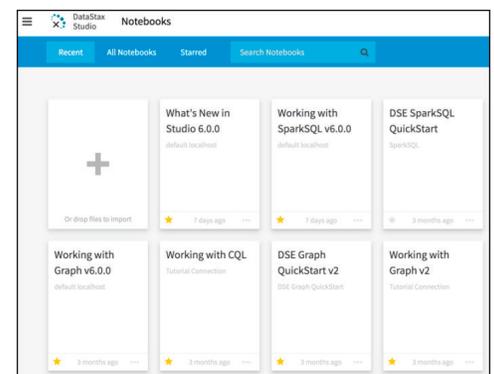
For the 90 percent write and 10 percent read workload, DSE 6 also showed reduced latency.

The performance improvements shown by DSE 6 were even more pronounced for the 50 percent read / 50 percent write and the 10 percent write / 90 percent read workloads.

## Using DSE 6

DSE 6 has language drivers for C/C++, C#, Java, Node.js, PHP, Python, and Ruby; these all support CQL and DSE-specific features. There are an additional nine community-supported language drivers for Cassandra. Graph extensions supporting Apache TinkerPop are available for C#, Java, Node.js, and Python. There are ODBC and JDBC drivers for connecting to Cassandra and DSE from BI tools and other systems that use SQL, and a Spark connector that both exposes tables as Spark RDDs and writes RDDs to tables.

DSE 6 includes both a CQL shell command line and an integrated development environment. DataStax Studio (shown below) is an interactive notebook-oriented tool for CQL (Cassandra Query Language), Spark SQL, and DSE Graph.



DataStax Studio has a Jupyter notebook interface and comes with seven tutorial notebooks. It supports CQL, Spark SQL, and TinkerPop.

For a basic local smoke test, I downloaded and unpacked the DSE 6.0.2 tarball onto an iMac that already had Java 8 and Python installed. The tarball

