



NoSQL in the Enterprise

A Guide for Technology Leaders
and Decision-Makers

WHITE PAPER

NoSQL in the Enterprise

A Guide for Technology Leaders
and Decision-Makers

By DataStax Corporation
August 2011



Table of Contents

Introduction.....	3
An Overview of NoSQL	3
The Rise and Momentum of NoSQL in the Enterprise.....	4
Is NoSQL Replacing the RDBMS in the Enterprise?	5
What Constitutes an Enterprise NoSQL Solution?	6
Technical Characteristics of an Enterprise-Class NoSQL Solution.....	6
Primary and Analytic Data Source Capable.....	6
Mixed-Workload Isolation Within a Single Database	6
Safe for Critical Data	7
Fault-Tolerant (No Single Point of Failure)	7
Multi-Data Center Capable	7
Easy Replication for Distributed Read and Write Anywhere Capabilities	7
No Need for Separate Caching Layer	8
Cloud-Ready	8
Big Data Capable.....	8
High Performance with Linear Scalability	9
Flexible Schema Support.....	9
Support Key Developer Languages and Platforms.....	9
Easy to Implement, Maintain, and Grow	9
Thriving Open Source Community.....	9
Business Considerations for a NoSQL Enterprise Solution	10
Backed by a Commercial Entity	10
Enterprise Support and Services	10
Professional Documentation	10
Referenceable Customers Across Different Industries	10
Cost-Effective.....	11
Accepted By All Major Stakeholders.....	11
A Recommended Enterprise NoSQL Checklist	12
An Overview of DataStax	13
What is Apache Cassandra?.....	13
What is DataStax Enterprise?	13
Industries Served by DataStax.....	15
Conclusions	16
About DataStax	16

Introduction

The information processing demands of many of today's businesses long ago outgrew the legacy RDBMS software that first appeared in the mid-1980s with IBM, and then continued into the 1990s with Oracle, Sybase, Microsoft SQL Server, and MySQL. The Web's explosive growth since the 1990s has only amplified the need for businesses to manage increasingly large volumes of data – data that must be made available across a distributed (geographically or otherwise) system and does not fit neatly into a relational data model.

While Internet giants such as Amazon, Facebook and Google may have been the first to truly struggle with the “big data problem,” enterprises across industries – and not just Web-based organizations – are now struggling to manage massive quantities of data, or data entering systems at a high velocity, or both. As an example, according to a recent report from consulting giant McKinsey & Company, the average investment firm with fewer than 1,000 employees has 3.8 petabytes of data stored, experiences a data growth rate of 40 percent per year, and stores structured, semi-structured, and unstructured data.¹

As pressing dilemmas typically give rise to innovation, it wasn't long before data scientists and engineers delivered a new and advanced set of software designed to meet 21st century data management demands. The term “NoSQL” was introduced to describe the progressive data management engines that contained some RDBMS-like qualities, but went beyond the limits that currently shackle traditional SQL-based databases.

There hasn't been such a rapid shift to a new method for storing data since the move from hierarchical to relational data stores. Conferences devoted to addressing modern data management challenges have been sold out – and most have focused agendas on NoSQL topics. Technology leaders are no longer addressing the question of *if* they'll have a NoSQL strategy, but rather *when* their NoSQL strategy will roll out – and more importantly, what it will be comprised of.

That last question is not easy to answer, as the NoSQL ecosystem has been one of rapid change, with numerous software offerings appearing under the NoSQL umbrella. However, as more enterprises have implemented NoSQL solutions, a distinctive set of criteria has emerged that can help today's IT professional more easily identify the NoSQL solutions built for enterprise-wide deployment.

This paper outlines these characteristics in detail so that those implementing a NoSQL strategy can make more informed decisions when (1) choosing a particular set of NoSQL software, and (2) deciding which vendors to target.

An Overview of NoSQL

What exactly is NoSQL? Some think NoSQL and Hadoop (a batch analytic infrastructure used to process large volumes of data) are synonymous. Others believe NoSQL always equates to data warehousing. But the characteristics that constitute a NoSQL database extend beyond these narrow definitions. Today's NoSQL databases can:

¹ http://www.mckinsey.com/mgi/publications/big_data/index.asp

- Serve as an online processing database, so that it becomes the primary datasource/operational datastore for online applications, or what is sometimes called the “system of record”
- Use data stored in primary source systems for either real-time or batch analytics
- Comfortably store and quickly process data volumes that range anywhere from gigabytes to petabytes
- Excel at distributed database operations (some better than others)
- Offer a flexible schema design that can be changed without downtime or service disruption
- Accommodate structured, semi-structured, and non-structured data
- Easily operate in the cloud and exploit the benefits of cloud computing

Clearly, a NoSQL database is capable of doing much more than some think. The “No” part of the NoSQL label can be thought of as “not only SQL,” which communicates the fact that a NoSQL database doesn’t completely discard all features/functions that define a relational database. In fact, a few NoSQL databases provide an SQL-like query language that helps ease the transition from the RDBMS world.

What is true about most – if not all – NoSQL databases is that they don’t conform to the standard Codd-Date relational model², where data is normalized to a third logical form. Such data structures often require resource-intensive join operations to satisfy end user requests. Instead, data in a NoSQL database is greatly denormalized and resides in structures organized in a variety of formats (e.g., columnar, document, key/value, graph).

Whereas such data is either impossible to store properly in an RDBMS or performs very poorly when accessed in a relational manner, NoSQL databases are defined by how well they handle such data and the speed at which they do so. For example, a standard RDBMS does not handle “wide” rows (rows consisting of many columns) very well, but a NoSQL database such as Cassandra can have data structures that each consist of thousands of columns and both write and read such data at speeds that quickly outdistance its RDBMS predecessors.

The Rise and Momentum of NoSQL in the Enterprise

The capabilities of NoSQL databases are fast becoming well known to IT leaders. For example, a recent Evans Data survey revealed that corporate enterprise developers in North America are rapidly accepting NoSQL. The study also showed that NoSQL databases already are being used in 56 percent of organizations surveyed, and 63 percent of respondents said they plan to use NoSQL in the next two years.

² http://en.wikipedia.org/wiki/Edgar_F._Codd

Percent of Planned Enterprise NoSQL Usage

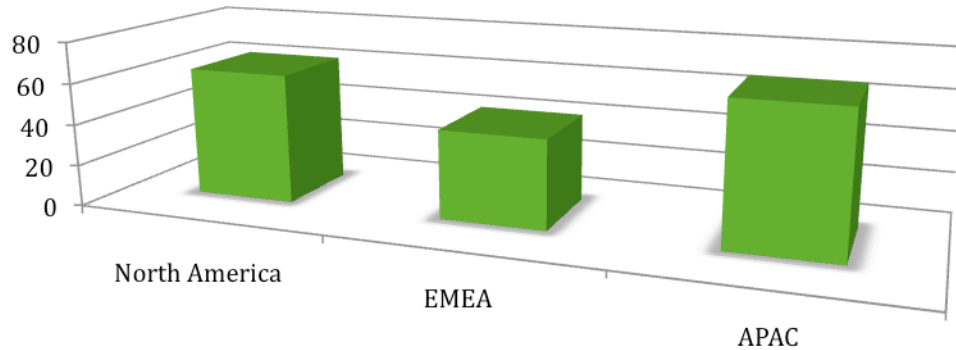


Figure 1 - NoSQL Momentum, Evans Data, 2011

*"The advent of 'Big Data' is driving adoption of NoSQL, **and this is especially true in the corporate enterprise.** While it may have gotten its start on the Web with innovations like BigTable and MapReduce, **it's the enterprise that can most benefit from NoSQL,** and developers realize this across all geographical regions."*

—Janel Garvin, CEO of Evans Data

An interesting note about the Evans survey's findings is that the NoSQL movement is much stronger in the enterprise segment than within the general developer population (where 43 percent of respondents said they expect to use NoSQL). Such a statistic demonstrates that NoSQL databases are meeting real corporate data management needs versus just being another niche, albeit interesting, technology.

Evans Data also found that NoSQL is showing strong growth in the EMEA (Europe, Middle East and Africa) region, where about 40 percent of enterprises are undertaking NoSQL projects. The rise of NoSQL is even higher in the Asia-Pacific region, where nearly 70 percent of Evans Data's responders report that they are planning NoSQL implementations.

Is NoSQL Replacing the RDBMS in the Enterprise?

Such large percentage indicators of NoSQL usage naturally raise the question of whether NoSQL is replacing the traditional relational database in the enterprise. The answer is both yes and no.

Many enterprises are choosing to leave some legacy RDBMS systems in place, while directing new development towards NoSQL databases. This is especially the case when the applications in question demand high write throughput, need flexible schema designs, process large volumes of data, and are distributed in nature.

However, some businesses are choosing to replace existing relational systems with NoSQL solutions. As an example, Netflix, the world's leading Internet subscription service for movies and

TV shows, has replaced a number of its existing Oracle systems with Cassandra running in the cloud.³

Technology aside, another reason many new development and/or migration efforts are being directed towards NoSQL databases is the high cost of legacy RDBMS vendors versus NoSQL software. In general, NoSQL software is a fraction of what vendors such as IBM and Oracle charge for their databases.

What Constitutes an Enterprise NoSQL Solution?

What should a technology leader or decision-maker look for in a NoSQL offering that defines it as truly being “enterprise ready”? To help answer this question, the following sections outline enterprise-class characteristics to look for in a NoSQL solution targeted for widespread usage. The technical attributes are outlined first, followed by a detailed overview of key business considerations.

Technical Characteristics of an Enterprise-Class NoSQL Solution

Following are the desirable technical attributes of an enterprise-capable NoSQL solution.

Primary and Analytic Data Source Capable

The first consideration of an enterprise-class NoSQL solution is that it is capable of serving as both a primary or operational datasource (sometimes called the “system of record”) that accepts data from various application/customer points of entry, and also can act as an analytic database (or secondary datasource) that powers business intelligence applications.

From a system of record perspective, the NoSQL database should be able to assimilate all types of data – structured, semi-structured, and unstructured – in a very rapid fashion. It also should offer high-performance query capabilities.

Once data is in the database, decision-makers naturally want to analyze it, both in real time and in map/reduce form for heavy analytic operations. An enterprise-class NoSQL database should be able to handle such requests on the same database without having to duplicate the data into another, separate analytic datastore.

Mixed-Workload Isolation Within a Single Database

Industry analyst Gartner Group identifies mixed-workload management (e.g., OLTP and analytics, batch/real-time analytics) among the top challenges data management professionals have been facing for a number of years. In addition, Gartner identifies mixed-workload as a continuing issue for 2011-2012.⁴

Mixed-workload situations raise two key questions for today’s IT professional:

- How to avoid constant ETL operations and multiple databases to serve different workloads.
- How to isolate workloads “smartly,” so they don’t compete with one another for resources.

³ <http://blip.tv/datastax/replacing-datacenter-oracle-with-global-apache-cassandra-on-aws-5395633>

⁴ <http://www.gartner.com/it/page.jsp?id=1542914>

An enterprise-class NoSQL solution will deliver methods for handling these and other similar workload issues. A basic strategy to tackle this involves marking certain nodes in a cluster as being for real-time data and other nodes as being analytic in nature. Once that's accomplished, the database then smartly manages each workload on each set of nodes, ensuring they don't compete with each other.

Safe for Critical Data

One criticism that's been aimed at NoSQL databases is their "eventual consistency" model of dealing with data. NoSQL databases typically strive to deliver strong availability and partition tolerance in a database cluster, but to do so data consistency sometimes is sacrificed. The concern has been, as a result, that NoSQL databases don't provide a satisfactory level of protection for critical data.

However, this isn't true for all NoSQL solutions. Cassandra, for instance, offers a "tunable consistency" model where a developer/architect can choose the degree of consistency desired on a global or per-operation basis. They can decide between strong and eventual consistency depending on the situation. This provides for great flexibility and choice; Cassandra can behave much like a typical RDBMS – when needed – where data consistency is concerned, or it can deliver eventual consistency when the use case permits it.

Fault-Tolerant (No Single Point of Failure)

For a NoSQL database to be considered enterprise-capable, it needs to offer strong high availability (HA) where the configuration preferably has no single point of failure. Moreover, rather than having to construct an HA configuration outside of the software, the NoSQL solution should deliver HA in an out-of-the-box fashion.

Key things to look for include:

- All nodes in a cluster being able to serve in the same capacity (i.e., no "master" node), which equates to operational simplicity,
- The ability to replicate and segregate data easily between different physical racks in a data center (to avoid hardware outages), and
- The ability to support data distribution schemes that are either multi-data center or on-premises and in the cloud.

Multi-Data Center Capable

Today's businesses have highly distributed databases that often span multiple data centers as well as multiple geographic regions. Although replication has been a main feature in literally every legacy RDBMS, none offer a simple method for distributing data between different data centers where performance isn't an issue. Part of the definition of "simple method" includes being able to handle n-number of data centers and not worry about where read and write operations occur.

A good enterprise-class NoSQL solution offers simply implemented, multi-data center data distribution options that provide smart and configurable compromises between performance and data consistency.

Easy Replication for Distributed Read and Write Anywhere Capabilities

One major data distribution problem facing many RDBMS and some NoSQL solutions is their reliance on a sharded or master/slave architecture, where the master eventually becomes the bottleneck for write operations, and undesirable latency issues exist with slave machines fed from the master machine.

To overcome this issue – and ensure multi-geographical sites experience excellent performance while sharing the same database – a good NoSQL solution will provide strong replication abilities. This includes not only a read-anywhere capability, but also full support for write-anywhere functionality. This allows users to write their data to any node in a cluster and automatically have that data replicate to other nodes and be available for all user accounts, no matter where they're located.

Lastly, writes on any node should be durable in nature such that if a power failure or other disruptive event occurs, data is safe.

No Need for Separate Caching Layer

Another enterprise characteristic of a good NoSQL solution is that, because it can easily use multiple nodes and smartly distribute data among all participating nodes, it eliminates the need for a special caching layer. Instead, the memory caches of all participating nodes are used to store data for quick I/O access.

An additional benefit of this capability is that it eliminates irregularities between the cache and the persistent database layer, which equates to having simple scalability with fewer management headaches.

Cloud-Ready

As of 2011, cloud computing accounts for only 2 percent of IT spending, but that's quickly changing. Analyst group IDC predicts that by 2015, close to 20 percent of all information will be attached to cloud services in some way, and as much as 10 percent will reside in an internal cloud infrastructure.⁵

Therefore, it's critical for an enterprise-class NoSQL solution to be cloud-ready. This means being able to easily spin up/take down a NoSQL database cluster in a cloud setting such as Amazon EC2, expand and contract a cluster at will, and more.

Further, advanced functionality for the NoSQL database includes being able to support a hybrid solution where part of the database is contained in an on-premise fashion and another part is hosted in the cloud.

Big Data Capable

Each day, 4 billion pieces of information are shared on Facebook alone. But handling big data is not just a problem for companies like Facebook. To put things into perspective, the U.S. Library of Congress, as of April 2011, had collected 235 terabytes of data. McKinsey Global Institute says that 15 out of the 17 main sectors in the marketplace already have more data per company than the Library of Congress – and that data is predicted to grow at 40 percent per year.⁶

Although a NoSQL database is not restricted to working only with “big data,” one of the hallmarks of an enterprise-ready NoSQL solution is that it can – when asked – scale to manage anywhere from terabytes to petabytes of data. This capability goes beyond simply being able to store large volumes of data; it also means delivering high performance for both read and write operations no matter the size of the data.

⁵ <http://www.asiacloudforum.com/content/study-cloud-process-20-worlds-data-2015>

⁶ http://www.mckinsey.com/mgi/publications/big_data/index.asp

High Performance with Linear Scalability

Piggybacking on the big data requirement, an enterprise NoSQL database should offer the ability to increase performance through adding nodes to a cluster. Whereas some database systems actually experience performance degradation when additional boxes are added to a configuration, a good NoSQL solution delivers the exact opposite: adding nodes should increase performance for both read and write operations. Additionally, those performance gains should be mostly linear in nature.

Flexible Schema Support

A key characteristic of an enterprise NoSQL database is its ability to offer a flexible, or dynamic, schema design able to consume structured, semi-structured, and non-structured data. This ability negates the need to have many different vendors for the types of data that must be supported throughout the organization. Keep in mind that different NoSQL databases support different schema formats (e.g. columnar/BigTable, Document, etc.) so some will match various application needs better than others.

Additionally, flexible/dynamic schema support means schema changes can be made to a structure without that structure going offline. With many applications requiring near-zero downtime and around-the-clock availability, this support is critical.

Support Key Developer Languages and Platforms

Naturally, an enterprise-class NoSQL solution should support all key operating systems in use today. It also should be able to run on commodity hardware that needs no special hardware tweaks or other proprietary additions.

The NoSQL database also should provide client interfaces and drivers for all popular developer languages. Lastly, given that many developers are coming from one or more legacy RDBMS, the NoSQL solution should offer an SQL-like language that helps ease the transition into storing and accessing data in a NoSQL database.

Easy to Implement, Maintain, and Grow

“Complex” and “difficult to use” should not describe a NoSQL solution that is a candidate for wide enterprise-scale rollout. Instead, a NoSQL database should be “simple” software – but not “simplistic” software. In short, it should be easy to implement and use, but offer strong and deep functionality capable of handling enterprise applications.

Moreover, the NoSQL provider should supply good management tools that assist the data professional in managing, monitoring, and performing various administrative tasks such as adding capacity to a cluster, running various utility tasks, and more.

Lastly, because successful businesses oftentimes have no idea where they will be 6-12 months from the present, the NoSQL database should allow for easy growth without requiring any change to the front-end application.

Thriving Open Source Community

If the NoSQL database is open source in nature, then it's important to have a vibrant community behind it – one that's growing, active, and contributes regularly to making the core software better. In addition, a strong open source community provides excellent quality assurance (QA) testing that often far exceeds the ability of most commercial software companies to hire, train, and retain professional QA staff.

A number of indicators can be used to validate a thriving open source community, including activity on mailing lists and technical forums, growing numbers of local user groups, and healthy attendance at large-scale conferences.

Business Considerations for a NoSQL Enterprise Solution

A NoSQL solution may have excellent technical attributes, but there's more to consider than just pure technology when evaluating NoSQL databases for a modern enterprise. Various business and nontechnical considerations should be weighed as well when deciding whether to roll out a particular NoSQL solution on an enterprise-wide scale.

Following are some of the key business must-have's for an enterprise-class NoSQL database.

Backed by a Commercial Entity

While it's important to have a strong open source community behind a NoSQL database (if the database in question originated in the open source world), equally important is that the NoSQL solution be backed by a viable commercial entity that marries the benefits of open source with the advantages that come from doing business with commercial software vendors.

Enterprise Support and Services

One major benefit of having a commercial company behind a NoSQL database is the full range of support and services provided by such an entity. If a particular technical issue arises in a production NoSQL system, the absolute last thing an IT manager wants to do is post a cry for help on some community forum and hope someone, somewhere responds in a timely fashion with advice that hits the mark.

An enterprise-class NoSQL solution should include complete access to professional, experienced production support – around the clock, if needed. Such support should include service level agreements where response times are concerned, as well as other expected services such as consultative support.

On the consulting front, the commercial entity should provide a range of professional services that can be used in both pre- and post-production so that an organization can jump-start its progress with the new NoSQL database. The ability to follow-up after implementing a NoSQL application to ensure things are running smoothly and that future capacity needs are being taken into account should be available as well.

Lastly, the commercial vendor should provide a series of training courses designed to take both developers and system architects from beginning to end where the NoSQL database is concerned. Good training courses should offer both classroom discussion and real-world lab exercises so the concepts being taught are solidified through actual practice.

Professional Documentation

One often overlooked aspect of a quality NoSQL solution is professional documentation that's always accessible online. Such documentation should cover the basic concepts of the NoSQL database, describe how to architect, develop, manage, and monitor applications targeting the NoSQL database, and also provide quick/jump-start guides to assist in an evaluation of the software.

Referenceable Customers Across Different Industries

Another key characteristic of an enterprise-ready NoSQL solution: referenceable customers successfully using the NoSQL database in production. Having customers in a variety of different

industries also indicates that the NoSQL database under consideration is not a niche software product, but a solution that addresses a wide range of needs across a wide array of use cases and application settings.

Cost-Effective

The high cost of commercial RDBMS software is well known, with products from Oracle, IBM, and Microsoft often requiring a seven-figure investment just to get the project under way – and a yearly 20 percent minimum maintenance charge to retain the assistance of support personnel and software updates.

By contrast, a good NoSQL offering will have a disruptive pricing strategy that usually makes the software available and affordable to everyone.

Accepted By All Major Stakeholders

The issues we've addressed primarily come from four key stakeholders in today's organizations:

1. "The Business" – More than ever, increasing demands are being placed on IT by the business side of the organization. Any solution must be able to adapt and grow to meet these challenges to help gain a competitive advantage in the market place.
2. Developers – Backend systems must allow flexibility for changes to the application, and scalability that they do not need to manually manage.
3. Operators/Administrators – Once the system is in production, it must meet the rigorous demands of a mission-critical application and must be easy to manage and provision for the operations teams.
4. IT Executives – Need solutions that provide all these things, while at the same time, reducing overall IT costs by less expensive TCO and fewer resources to manage the systems.

It is critical that each of these stakeholders needs is taken into account throughout the planning and decision process.

A Recommended Enterprise NoSQL Checklist

Below are technical and business criteria for an enterprise-class NoSQL solution combined into a single checklist:

Technical Considerations

- Can the NoSQL database serve as a primary data source (a “system of record”)?
- Can the NoSQL database operate as an analytic data source?
- Can the NoSQL database provide workload isolation in a single database?
- Is the NoSQL database safe where storing critical data is concerned?
- Is the NoSQL database fault tolerant (has no single point of failure)?
- Can the NoSQL database easily replicate data between the same and multiple data centers?
- Does the NoSQL database offer read/write anywhere capabilities?
- Are writes durable in nature such that data is safe?
- Does the NoSQL database remove the need for special caching layers?
- Is the NoSQL database cloud-ready?
- Is the NoSQL database capable of managing “big data” and delivering high performance results regardless of data size?
- Does the NoSQL database offer linear scalability where adding new nodes is concerned?
- Does the NoSQL database offer flexible schema support?
- Does the NoSQL database support key platforms/developer languages?
- Can the NoSQL database run on commodity hardware with no special hardware requirements being needed?
- Is the NoSQL database easy to implement and maintain?
- If open source, does the NoSQL database have a thriving open source community?

Business Requirements

- Is the NoSQL solution backed by a commercial entity?
- Does the commercial entity provide enterprise support and services?
- Does the NoSQL solution have professional online documentation?
- Does the NoSQL solution have referenceable customers across a wide range of industries?
- Does the NoSQL database have an attractive cost/pricing structure?

With these criteria in mind, let’s see how well Apache Cassandra and offerings from DataStax meet the requirements for an enterprise-class NoSQL solution.

An Overview of DataStax

DataStax is the leading provider of enterprise NoSQL software products and services based on Apache Cassandra™. Through its offerings, DataStax supports businesses that need a progressive data management system able to serve as a primary system of record/operational datastore for critical production applications, and also deliver built-in analytic capabilities for analyzing that data once it's in Cassandra.

What is Apache Cassandra?

Apache Cassandra is a highly scalable and high performance distributed database management system. Cassandra is able to manage the distribution of data across multiple data centers and offers incremental scalability with no single point of failure. Cassandra is a logical choice for enterprises that need high degrees of uptime, reliability, and very fast performance. Many leading companies, including Cisco, HP, Motorola, Netflix, Ooyala, Openwave, Rackspace, and Twitter, rely upon Cassandra to manage the data needs of their critical production applications.

Cassandra was originally incubated at Facebook and is based upon Google's BigTable and Amazon's Dynamo software. The end result is an extremely scalable and fault-tolerant data infrastructure that solves small to big data problems, handles write intensive workloads, delivers sub-millisecond caching layer reads, and supports batch analytical map/reduce workloads involving petabytes of data.

What is DataStax Enterprise?

DataStax Enterprise is an enterprise-class NoSQL solution that has Cassandra as its foundation. However, with DataStax Enterprise, DataStax also provides advanced data management functionality above the community Cassandra product, as well as complete production support, visual management tools, and services to ensure every customer is successful with the software.

As can be seen below, DataStax Enterprise nicely fulfills the requirements of an enterprise NoSQL solution:

Requirement	DataStax Enterprise	Notes
Technical Requirements		
Serve as primary data source	Yes	System of record capable
Serve as analytic data source	Yes	Supports Hadoop analytics with Hive and Pig support on Cassandra data
Workload isolation in single database	Yes	Isolates Cassandra real-time and Hadoop operations on different nodes
Safe for critical data	Yes	Provides tunable data consistency and durable writes
Fault tolerant (no single point of failure)	Yes	Peer-to-peer architecture
Multi-data center aware	Yes	Easy to configure multi-data center replication
Easy replication (read/write anywhere)	Yes	One configuration option controls how many copies of data are replicated among nodes
No need for caching layer	Yes	Easy distribution of data and use of multiple machine's memory removes need for caching software
Cloud ready	Yes	Can run fully in the cloud or in a hybrid mode of part-cloud/part-on premises
Big data capable	Yes	Petabyte capable
High performance/linear scalability	Yes	Fastest NoSQL solution for writes and extremely fast reads
Flexible schema support	Yes	Based on Google BigTable
Support for key platforms/developer languages	Yes	Available for all popular platforms and languages. Also includes CQL language that is very similar to SQL
Easy to implement and maintain	Yes	Visual management tool – OpsCenter – included, that manages and monitors performance across a database cluster
Thriving open source community	Yes	Numerous committers, developers, and user groups

Business Requirements

Backed by commercial entity	Yes	DataStax
Enterprise support and services	Yes	24x7 production support, consultative services, and professional training
Professional documentation	Yes	All available online
Referenceable customers	Yes	Many customers across nearly every industry
Cost-effective	Yes	Available as a subscription per node

Industries Served by DataStax

The industries currently using DataStax to support key applications include:

- Consulting
- Consumer Electronics
- E-Commerce
- Entertainment
- Energy
- Financial
- Government
- Healthcare
- Hosting
- Marketing/Advertising
- Messaging
- Mobile Applications
- Online Gaming
- Retail
- Security
- Social Media
- Social Networking
- Software
- Travel

“Customers turn to us for highly complex analysis. The best way for us to deliver the experience our users demand is to employ extremely fast, scalable distributed computing based on Cassandra.”

—Harry Schultz, Digital Reasoning

Conclusions

Businesses that have outgrown legacy relational systems are now turning to NoSQL solutions to manage their critical data needs. NoSQL databases have shown they're capable of handling both real-time/system of record applications as well as analytic and business intelligence systems. This is why many enterprises have already elevated NoSQL as a primary data provider along with traditional RDBMS's.

However, not all NoSQL databases are created alike – and some are more enterprise-ready than others. This paper has outlined the key criteria for selecting an enterprise-class NoSQL solution and has shown that the software and services offered by DataStax meet them all.

To find out more about DataStax and its products and services, or to get started today with downloads of DataStax's NoSQL solutions, please visit www.datastax.com or send an email to info@datastax.com.

About DataStax

DataStax, the commercial leader in Apache Cassandra™, offers products and services that make it easy for customers to build, deploy and operate elastically scalable and cloud-optimized applications and data services. Over 90 customers use DataStax today, including leaders such as Netflix, Cisco, Rackspace and Constant Contact, with industries served including web, financial services, telecommunications, logistics and government.

DataStax is backed by industry leading investors, including Lightspeed Venture Partners, Sequoia Capital and Rackspace Hosting, and is based in Burlingame, CA with offices in Austin, TX and Stamford, CT.

For more information, visit www.datastax.com.