DATASTAX

Implementing a NoSQL Strategy

White Paper

BY DATASTAX CORPORATION

JULY 2013

Table of Contents

Abstract

NoSQL database market is expected to grow at a rate three times faster than that of the SQL market during the next few years, understandably making newcomers to big data technology eager to understand why and how it fits into their organizations. The needs for speed, scale, continuous availability, location independence, ability to manage all types of data, and cost reduction are driving this increasing adoption. Barriers to adoption have certainly existed, from the technical to the non-technical, and it's important to examine those along with improvements that have developed in the NoSQL ecosystem. Furthermore, use cases are gaining speed as important indicators of when and how organization should use NoSQL technologies. This paper examines these topics and provides practical implementation strategy tips, including a selection checklist and migration pointers.

Introduction

Describing the current and expected growth of NoSQL technology, a 2013 article in Silicon Angle stated: *"According to analysis by Wikibon's David Floyer (and highlighted in the Wall Street Journal), the NoSQL database market is expected to grow at a compound annual growth rate of nearly 60% between 2011 and 2017. The SQL slice of the Big Data market, in contrast, will grow at just a 26% CAGR during that same time period."*[1]

With major enterprises now putting NoSQL solutions in key line of business applications that power major aspects of their company, many IT leaders not using NoSQL are interested in understanding why and how they can use the technology in their organizations.

This paper provides a general enterprise implementation strategy for NoSQL by exploring the key reasons why enterprises are turning to NoSQL solutions and providing examples of how modern businesses are using the technology today. It also discusses what inhibitors companies have faced in implementing NoSQL, provides advice on how to select a NoSQL database, and discusses practical ways for implementing NoSQL solutions in various application scenarios.

What is Driving NoSQL Adoption in the Enterprise?

In determining how to implement a NoSQL strategy for your business, it's helpful to first examine the top reasons why successful modern enterprises have turned to NoSQL solutions and see if those needs are present in your organization as well. Although certainly not exhaustive, the following represent the key motivations for why companies have implemented NoSQL databases in their critical line of business applications.

The Need for Speed

Nearly everyone acknowledges the fact that improving response times for external facing systems can directly impact customer satisfaction and revenue. For example, Amazon found

[1] "Oracle is in Big Trouble: Big Data is to Blame", by Maria Deutscher, Silicon Angle, March 2013: http://goo.gl/ODA7g.

that every 100ms reduction in site response time netted them 1% more in revenue; Yahoo states that they have seen site traffic increase 9% for every 400ms speedup in performance.[2] Database professionals have always dreamed of setting the universal "`fast=true`" database parameter and having their database run blindingly fast at all times. Although no such configuration setting exists, what is real is the fact that fast database response times have never been more important than today with businesses having countless competitors just a click away.

Because they don't adhere to many of the encumbrances of relational databases (RDBMS's), NoSQL databases can deliver faster performance for many use cases. A hallmark of NoSQL solutions like Apache Cassandra has been the ability to write data much faster than an RDBMS as well as deliver just as fast query speeds across large volumes of data.

eBay utilizes DataStax Enterprise, which is powered by Apache Cassandra, for this very reason. Citing the fact that legacy relational engines were too slow for key parts of their external applications, eBay replaced their traditional databases with DataStax Enterprise and now meets their response time SLA's servicing 6 billion writes and 5 billion reads per day, while also managing 250TB of data in Cassandra.[3]

The Need for Scale

Scalability and performance go hand in hand, with companies needing to maintain fast response times while accommodating increasing numbers of users and data volumes in their line of business applications. Traditional scale up architectures have proven unsuccessful at "future proofing" the scalability of such systems.

However, NoSQL databases like Apache Cassandra provide a scale out, divide-and-conquer architecture that affords proven linear scalability[4] with the addition of new nodes that can be added online and without business interruption.

One company that exemplifies this approach is Ooyala, which serves as a video distributor and data analysis provider for companies like ESPN and Rolling Stone. Ooyala tracks and analyzes literally one-quarter of all online video views each day on the Web, with that translating into billions of events that are streamed through their scale-out DataStax Enterprise / Cassandra database clusters. Ooyala says that the scale and performance they needed could not be met with a legacy scale-up database implementation, so they choose Cassandra instead.

The Need for Continuous Availability

While slow performance can drive customers away, nothing is worse than downtime. IT industry expert Gartner Group states that downtime serves as a bigger risk to companies than security breaches.[5] In terms of lost revenue, predictions range up to $6.5 million per hour for financial institutions, with the average across all industries being $5,600/minute or over $300,000/hr.[6]

There is a difference between the failover-styled high availability approach that RDBMS's offer with their master-slave architectures than the *continuous availability* that NoSQL databases like Apache Cassandra provide. Because of its scale-out, masterless design, and multi-data center

---

[2] http://www.strangeloopnetworks.com/assets/images/infographic2.jpg

[3] "EBay Leveraging Cassandra to Support Growing Multistructured Data Volumes" by Jeff Kelly, May 9, 2013, Wikibon: http://goo.gl/tLHkB.

[4] "Benchmarking Cassandra Scalability on AWS" by Adrian CockCroft, November 2011: http://goo.gl/G8NUa.

[5] "Gartner's state of cloud security: Outages are bigger risk than breaches" by Brandon Butler, November 14. 2012: http://goo.gl/mpnm3.

[6] "Confronting System Downtime", Evolven: http://goo.gl/jWQty.

and cloud availability zone support, Cassandra ensures no downtime with redundant copies of data and function being spread throughout a cluster across multiple locations.
Netflix, which has been christened as the largest cloud application in the world[7], uses Cassandra to ensure zero downtime for its customers, storing 95% of its data in Cassandra. When Amazon experienced a major outage in 2012, Netflix never missed a beat noting: "We didn't need to do anything. Cassandra routed requests around the unavailable zone and when it recovered, the ring was repaired."[8]

The Need for Location Independence

Because nearly all successful businesses have a global reach, the ability to serve data quickly to multiple locations is critical. Because of their foundational master-slave design, legacy RDBMS's struggle with providing fast reads across many locations, and they simply cannot do a key thing that many enterprises need which is allow for write-anywhere capability.

NoSQL databases like Cassandra can easily spread data across multiple data centers and cloud availability zones. Further, because of its peer-to-peer architecture, Cassandra allows for both read and write anywhere capability and thus delivers true location independence where data is concerned.

Companies like Adobe appreciate this capability in Cassandra. For its marketing cloud application, Adobe runs its DataStax Enterprise / Cassandra database cluster between two data centers to ensure its customers can both read and write data fast no matter where they're located.

The Need for Managing All Types of Data

The variety of today's data types has proven to be a challenge for traditional relational databases and is one of the primary reasons enterprises have turned to NoSQL solutions for help. NoSQL databases like Cassandra offer a much more flexible data model that easily accommodates structured, semi-structured, and unstructured data and does so in a way that is performant and efficient from a storage perspective.

NASA uses Cassandra for security applications that track all hardware and software patches around the globe for the agency, and deals with data that is both structured and unstructured. NASA found that the flexible data model of Cassandra allowed them to insert data much more naturally than their prior RDBMS, plus query response times were reduced for retrieving the data as well.

The Need for Cost Reduction

One final driver of NoSQL adoption in the enterprise is cost reduction. Price sticker-shock is still very common for RDBMS's, especially for some of the new mainframe scale-up appliances that have been introduced. NoSQL solutions like DataStax Enterprise typically cost 70-80% less than legacy relational systems and are designed to run on cost-efficient commodity hardware.

Constant Contact, a company the serves the marketing needs of many small businesses, discovered huge cost savings when it turned to DataStax two years ago. Needing to scale their systems, but unable to afford the high costs of their previous RDBMS vendor to do so, the company chose a scale-out design and DataStax Enterprise, and ended up implementing the changed system in 1/3 less the time than estimated with their old RDBMS and at 90% less cost than their prior database.

---

[7] "The biggest cloud app of all: Netflix", by Steven J. Vaughan-Nichols, ZDnet, April 2013: http://goo.gl/KQRKr.
[8] "Post-mortem of October 22,2012 AWS degradation", Netflix Tech Blog, October 2012: http://goo.gl/X20sp.

Further, even though Constant Contact runs hundreds of DataStax Enterprise nodes, they found they required no dedicated admin to manage the database clusters, but instead have personnel that tend to the database part time each week along with other systems.

While there are certainly other motives companies have for implementing NoSQL in their IT infrastructure, the above reasons are the ones most cited by DataStax customers. The next question to consider when mapping out a NoSQL strategy is what roadblocks might you encounter when moving to NoSQL.

Inhibitors to NoSQL Adoption in the Enterprise

Although NoSQL solutions contain much promise for many different use cases, those who are successful with NoSQL admit there are things IT professionals should consider upfront before embarking on an enterprise rollout of the technology. Such inhibitors can be broken out into technical and non-technical categories.

Technical Inhibitors

Some of the top technical constraints that successful companies have wrestled with where NoSQL is concerned are the following:

- **Data Model Differences**: It cannot be emphasized enough that the number one NoSQL technical issue that companies have struggled with is making the mental switch from the relational to the NoSQL data model. Projects can be made or broken on whether the IT team has correctly modeled the data for the NoSQL database to maximize its capabilities. This being true, it is crucial that database professionals be trained and become thoroughly acquainted with the new NoSQL data model in the database they choose.
- **Lack of Security**: In 2012, InformationWeek ran a special story entitled "NoSQL Equals No Security".[9] In the article, the author cited the lack of security features in NoSQL databases that could negate their use in environments that necessitate strong data protection policies. However, it should be noted that while security capabilities are absent in some NoSQL databases, DataStax Enterprise *does* contain enterprise-class security features that meet the vast majority of enterprise security requirements.[10]
- **ACID Transaction Support**: The fact that most NoSQL databases do not support ACID-level transactions troubles some IT professionals. It is true that if your target application requires complex, nested transactions that necessitate rollbacks and savepoints then a NoSQL database may not be right for that particular situation. However, it should be noted that a NoSQL database like Cassandra does offer atomicity, durability, and isolation (AID), with consistency (C) being tunable: either eventual or strong depending

[9] "Why NoSQL Equals No Security", by Michael Davis, InformationWeek, March 2012: http://goo.gl/4E5Ac.
[10] See: "What's New in DataStax Enterprise 3.0?", http://goo.gl/GW1vb.

on what the application or particular operation needs, with transaction support for batches also being available.

Non-Technical Inhibitors

The primary non-technical issues that companies have dealt with in implementing NoSQL systems include:

- **Finding Experienced Personnel**: Fortunately, this problem is becoming less of an issue these days due to NoSQL having been deployed in many more companies than it was a few years ago, and the many training classes offered by NoSQL vendors. For employees transitioning to NoSQL, the learning curve will differ depending on the chosen technology. Some NoSQL databases like Cassandra help reduce the learning curve problem because the primary language interface is nearly identical to SQL.
- **Technical "Religious" Warfare**: Many corporations have IT groups that wage a type of religious war where technology is concerned. Alliances around certain technologies form and any attempt to bring in something new is met with resistance and an insistence that the old technology is still able to handle any job thrown at it. Overcoming this problem comes down to a data-driven methodology that includes both tech and non-tech analysis, which is submitted to impartial IT managers for review.
- **Vendor Viability Concerns**: IT executives making a long term commitment to any technology want the assurance that the provider(s) of that technology will be around for the long haul. Small startups with few customers will naturally raise eyebrows and concerns, whereas companies with a meaningful and growing customer base who are equipped with a proven support organization will supply the peace of mind that IT leaders are looking for.

But is NoSQL for You?

With an understanding of why enterprises are adopting NoSQL technology and what issues companies that have implemented the technology have wrestled with now out of the way, let's turn to an important question: How do you determine if NoSQL technology is right for you? A good place to start is by examining the opportunity for NoSQL in your business by industry and then by application use case.

From a global industry perspective, Gartner group produced the following grid that supplies an interesting view of how well various industries can be served by NoSQL technology given the key characteristics that define Big Data (velocity, variety, volume, complexity, etc.):

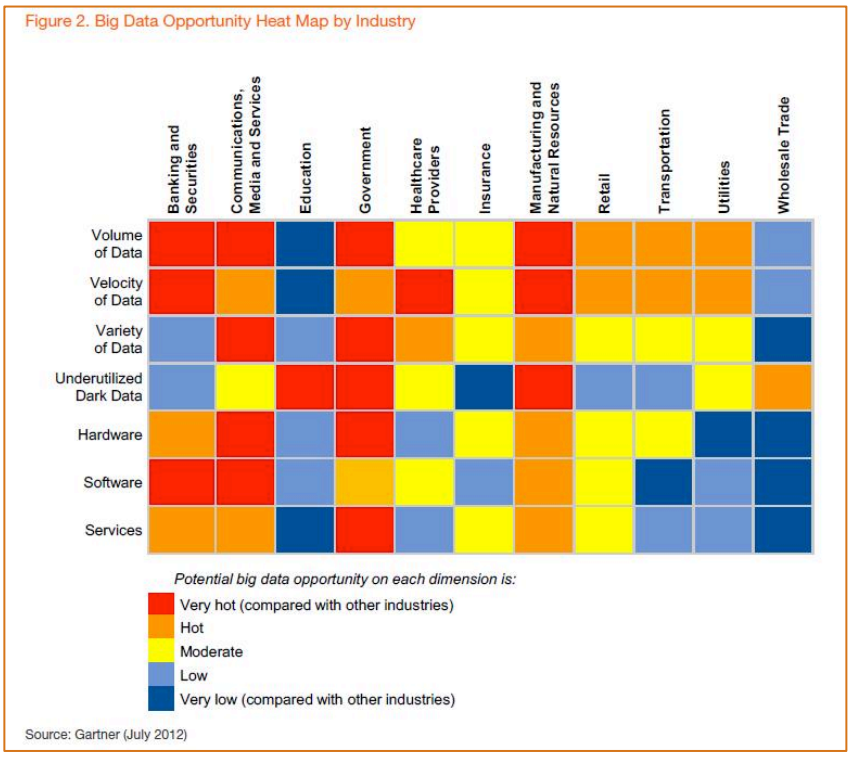

Figure 1 – Industry Heat Map for Big Data by Gartner Group

When it comes to looking at possible application use cases where NoSQL technology may be a fit, the broad use case categories of online, analytic, and search applications should first be brought into focus (with the understanding, of course, that a single application could involve a mix of all three).

Although not exhaustive, the following use cases are being supported today with NoSQL technology like Apache Cassandra, Hadoop, and Solr:

Online applications:

- Time series feeds (financial or other time-based data)
- Device/sensor/data "exhaust" systems
- Distributed transactional applications
- Media streaming
- Online web retail (e.g., transactional, shopping carts)
- Real-time data analytics
- Social media capture and analysis
- Web clickstream analysis
- Write-intensive transactional systems

Analytic applications:

- Buyer behavior analytics

- Compliance/regulatory analysis
- Customer recommendation output
- Fraud detection
- Risk analysis
- Sales program campaign analysis
- Supply chain analytics
- Batch web clickstream analysis

Enterprise search applications:

- General web search
- Web retail-faceted (categorization) search
- Search/hit prioritization and highlighting
- Application log search and analysis
- Document (e.g., PDF, MS Word) search and analysis
- Geospatial search
- Real estate location and property search
- Social media matchups

Choosing a NoSQL Database

The website nosql-database.org currently lists over 150 different NoSQL databases. How do you go about whittling down such a list into candidates that may be a fit for your application use cases?

Key Selection Criteria

There are many different features and functions that separate the different NoSQL databases, but the following criterion helps narrow the field for specific deployments:

- **The data model**: the primary consideration involves the type of data you need to store and its starting/ending format. NoSQL databases differ greatly in the data model used (e.g. wide row stores, document, graph, etc.) and a mismatch with a NoSQL solution's data model and the target application can make or break the success of a project.
- **The data scale expectations**: the next question involves how large an application is expected to grow and the data scale support that will be needed. Some NoSQL databases are main memory and do not scale out across multiple machines, whereas others like Cassandra scale linearly across many machines.
- **The data distribution model**: consideration should be given to how widely data needs to be distributed, whether to support multiple geographic regions, for disaster recovery purposes, or something else. Further, questions should be asked if both reads and writes will need to be supported in distributed locations. Some NoSQL databases use master-slave architectures (although they may term them “primary/secondary”), which

can only somewhat scale read operations vs. peer-to-peer architectures that can scale both reads and writes.

Enterprise Selection Checklist

A more detailed enterprise-ready checklist for NoSQL databases is below, and contains both technical as well as business considerations for determining the right match of a NoSQL database and an intended application use case.

Technical Considerations

- Can the NoSQL database serve as the primary data source for the intended online application?
- Can the NoSQL database operate as an analytic data source and/or easily interface with and support Hadoop operations?
- Can the NoSQL database handle or seamlessly integrate with enterprise search software?
- Can the NoSQL database provide workload isolation between online, analytic, and search operations in a single application?
- How safe is the NoSQL database where the possibility of losing critical data is concerned? Are writes durable in nature such that data is safe?
- Does the NoSQL database provide a robust security feature set?
- Is the NoSQL database fault tolerant (i.e., has no single point of failure) and does it provide continuous availability?
- Can the NoSQL database easily replicate data between the same and multiple data centers, as well as different cloud availability zones?
- Does the NoSQL database offer read/write anywhere capabilities?
- Does the NoSQL database require or remove the need for special caching layers?
- 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?
- Can new nodes be added and removed online (i.e. without business impact)?
- Does the NoSQL database support key platforms/developer languages?
- Can the NoSQL database run on commodity hardware with no special hardware requirements?
- Is the NoSQL database easy to implement and maintain for large deployments?

Business Requirements

- Is the NoSQL solution backed by a commercial entity?
- Does the commercial entity provide enterprise 24x7 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?
- If open source, does the NoSQL database have a thriving open source community?

Cloud Considerations for NoSQL Databases

The amount of information that currently resides only in the cloud is small, but that's about to change. A recent study by IT industry analyst group IDC estimates that cloud computing accounts for less than 2 percent of IT spending today, but by 2015, nearly 20 percent of all information will be "touched" (stored or processed) in a cloud.[11] Moreover, IDC predicts that by that same year, as much as 10 percent of all data will be maintained in a cloud.[12]

The cloud promises many things: transparent elasticity and scale, higher availability, simplified data distribution, easier manageability and more. However, it should be noted that while many database vendors claim their database is "cloud ready", what that oftentimes means is that you can easily install and run an instance of their database on a cloud provider. The bigger question to ask is, does the database exploit all or most of the supposed benefits of running a database in the cloud?

Whether it's a legacy RDBMS or a NoSQL database, the checklist items for truly realizing benefits from cloud computing are similar. The following are a suggested set of questions to ask any NoSQL database provider being considered for the cloud:

- Does the database provide transparent elasticity with expansion or contraction being possible without downtime?
- Can extra capacity be realized from scaling out in the cloud, and if so, how much benefit will be obtained?
- Can the database easily make use of a cloud provider's multiple availability zones so that continuous availability can be achieved in the event of one or more zone's failure?
- Does the database offer security features that protect data in the cloud?
- Does the NoSQL vendor provide management tools for managing and monitoring the database on the cloud provider?
- What is the 3-5 year cost difference in running the targeted NoSQL database on premise vs. the cloud?

Making the Move to NoSQL

From a practical perspective, how do you go about actually moving to NoSQL and implementing your first application? In general, there are three ways to go about implementing a NoSQL database:

1. **New applications**: many begin with NoSQL by choosing a new application and starting from the ground up. Such an approach mitigates the issues of application rewrites, data migrations, etc.
2. **Augmentation**: some choose to augment an existing system by adding a NoSQL component to it. This oftentimes happens with applications that have outgrown an

[11] "Extracting Value from Chaos", by John Gantz and David Reinsel, IDC, June 2011, http://idcdocserv.com/1142.
[12] Ibid.

RDBMS due to scale problems, the need for better availability, or other issues. Parts of the existing system continue to use the existing RDBMS whereas other components of the application are modified to utilize the NoSQL database.

3. **Full Rip-Replace**: for systems that simply are proving too costly from an RDBMS perspective to keep, or are breaking in major ways due to increases of user concurrency, data velocity, or data volume, a full replacement is done with a NoSQL database.

Migrating Data to NoSQL Databases

For either augmentation or rip-replace scenarios, migration of the data from the existing RDBMS to the new NoSQL database is required. Choosing how to migrate the legacy data depends on the amount of data needing to be moved:

- **Flat file loads**: most every RDBMS allows data to be exported from tables out to flat files that are delimited in some way. NoSQL databases like Cassandra have flat file loaders that take such files and load them directly into tables/column families.
- **Sqoop**: Sqoop is a utility used in Hadoop to move data from legacy databases into Hadoop. Cassandra also supports sqoop so a developer can connect to an existing RDBMS and Cassandra, and pump data straight into the new database.
- **ETL Tools**: If more sophistication is needed for a data migration, then any number of extract-transform-load (ETL) solutions can be used. Many tools from Jaspersoft, Pentaho, and Talend provide excellent transformation routines that allow source data to be manipulated in literally any way needed and then loaded into a NoSQL target. They also supply many other features such as visual, point-and-click interfaces, scheduling engines, and more. Finally, many are free to download and use.

Conclusion

Implementing an enterprise NoSQL strategy involves having a solid understanding of why successful companies are using NoSQL and deciding if those or other key characteristics of the technology can make an impact in your business. Once you have concluded that NoSQL is right for you, then it becomes a matter of smartly understanding what pitfalls to avoid, what criteria is needed to select the right NoSQL database(s) for your application use cases, and what strategy to use for rolling out the technology.

DataStax provides enterprise-class software, services, and strategies that ensure your success with NoSQL technology. With its proven and secure DataStax Enterprise solution – powered by Apache Cassandra – along with around-the-clock support, consulting, and training, the experts at DataStax can make sure your move to NoSQL is a positive and rewarding experience.

To find out more about Apache Cassandra and DataStax, and to obtain downloads of Cassandra and DataStax Enterprise software, please visit www.datastax.com or send an email to info@datastax.com. Note that DataStax Enterprise Edition is completely free to use in development environments, while production deployments require a software subscription to be purchased.

About DataStax

DataStax powers the big data applications that transform business for more than 300 customers, including startups and 20 of the Fortune 100. DataStax delivers a massively scalable, flexible and continuously available big data platform built on Apache Cassandra™. DataStax integrates enterprise-ready Cassandra, Apache Hadoop™ for analytics and Apache Solr™ for search across multi-data centers and in the cloud.

Companies such as Adobe, Healthcare Anytime, eBay and Netflix rely on DataStax to transform their businesses. Based in San Mateo, Calif., DataStax is backed by industry-leading investors: Lightspeed Venture Partners, Crosslink Capital and Meritech Capital Partners. For more information, visit DataStax.com or follow us @DataStax.