NoSQL

"Structured storage" redirects here. For the Microsoft technology also known as structured storage, see COM Structured Storage.

A **NoSQL** (originally referring to "non SQL"[1]) database provides a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases. Such databases have existed since the late 1960s, but did not obtain the "NoSQL" moniker until during a surge in popularity in the early twenty-first century,[2] triggered by the storage needs of Web 2.0 companies such as Facebook, Google and Amazon.com.[3][4][5]

Motivations for this approach include simplicity of design; simpler "horizontal" scaling to clusters of machine, which is a problem for relational databases;[2] and finer control over availability. The data structures used by NoSQL databases (e.g. key-value, graph, or document) differ slightly from those used by default in relational databases, making some operations faster in NoSQL and others faster in relational databases. The particular suitability of a given NoSQL database depends on the problem it must solve. Sometimes the data structures used by noSQL databases are also viewed as "more flexible" than relational database tables.[6]

NoSQL databases are increasingly used in big data and real-time web applications.[7] NoSQL systems are also sometimes called "Not only SQL" to emphasize that they may support SQL-like query languages.[8][9]

Many NoSQL stores compromise consistency (in the sense of the CAP theorem) in favor of availability, partition tolerance, and speed. Barriers to the greater adoption of NoSQL stores include the use of low-level query languages (instead of SQL, for instance the lack of ability to perform ad-hoc JOIN's across tables), lack of standardized interfaces, and huge previous investments in existing relational databases.[10] Most NoSQL stores lack true ACID transactions, although a few recent systems, such as FairCom c-treeACE, Google Spanner (though technically a NewSQL database), FoundationDB, Symas LMDB and OrientDB have made them central to their designs. (See ACID and JOIN Support.) Instead they offer a concept of "eventual consistency" in which database changes are propagated to all nodes "eventually" (typically with milliseconds) so queries for data might not return updated data immediately.

Unfortunately, not all NoSQL systems live up to the promised "eventual consistency" and partition tolerance, but in experiments with network partitioning often exhibited lost writes and other forms of data loss.[11] Fortunately, some NoSQL systems provide concepts such as "Write Ahead Logging" to avoid data loss.[12] Current relational databases also "do not allow referential integrity constraints to span databases" as well.[13]

1 History

The term *NoSQL* was used by Carlo Strozzi in 1998 to name his lightweight, Strozzi NoSQL open-source relational database that did not expose the standard SQL interface, but was still relational.[14] His NoSQL RDBMS is distinct from the circa-2009 general concept of NoSQL databases. Strozzi suggests that, as the current NoSQL movement "departs from the relational model altogether; it should therefore have been called more appropriately 'NoREL'",[15] referring to 'No Relational'.

Eric Evans reintroduced the term *NoSQL* in early 2009 when Johan Oskarsson of Last.fm organized an event to discuss open-source distributed databases.[16] The name attempted to label the emergence of an increasing number of non-relational, distributed data stores. Most of the early NoSQL systems did not attempt to provide atomicity, consistency, isolation and durability guarantees, contrary to the prevailing practice among relational database systems.[17]

As of July 2015, the most popular NoSQL databases are MongoDB, Apache Cassandra, Redis, Solr, ElasticSearch, HBase, Splunk, Memcached, and Neo4j.[18]

2 Types and examples of NoSQL databases

There have been various approaches to classify NoSQL databases, each with different categories and subcategories, some of which overlap. A basic classification based on data model, with examples:

- **Column**: Accumulo, Cassandra, Druid, HBase, Vertica
- **Document**: Clusterpoint, Apache CouchDB, Couchbase, DocumentDB, HyperDex, Lotus Notes, MarkLogic, MongoDB, OrientDB, Qizx

- **Key-value**: CouchDB, Oracle NoSQL Database, Dynamo, FoundationDB, HyperDex, MemcacheDB, Redis, Riak, FairCom c-treeACE, Aerospike, OrientDB, MUMPS
- **Graph**: Allegro, Neo4J, InfiniteGraph, OrientDB, Virtuoso, Stardog
- **Multi-model**: OrientDB, FoundationDB, ArangoDB, Alchemy Database, CortexDB

A more detailed classification is the following, based on one from Stephen Yen:[19]

Correlation databases are model-independent, and instead of row-based or column-based storage, use value-based storage.

2.1 Key-value stores

Main article: Key-value database

Key-value (KV) stores use the associative array (also known as a map or dictionary) as their fundamental data model. In this model, data is represented as a collection of key-value pairs, such that each possible key appears at most once in the collection.[20][21]

The key-value model is one of the simplest non-trivial data models, and richer data models are often implemented on top of it. The key-value model can be extended to an *ordered* model that maintains keys in lexicographic order. This extension is powerful, in that it can efficiently process key *ranges*.[22]

Key-value stores can use consistency models ranging from eventual consistency to serializability. Some support ordering of keys. Some maintain data in memory (RAM), while others employ solid-state drives or rotating disks.

Examples include Oracle NoSQL Database, redis, and dbm.

2.2 Document store

Main articles: Document-oriented database and XML database

The central concept of a document store is the notion of a “document”. While each document-oriented database implementation differs on the details of this definition, in general, they all assume that documents encapsulate and encode data (or information) in some standard formats or encodings. Encodings in use include XML, YAML, and JSON as well as binary forms like BSON. Documents are addressed in the database via a unique *key* that represents that document. One of the other defining characteristics of a document-oriented database is that in addition to the key lookup performed by a key-value store, the database offers an API or query language that retrieves documents based on their contents

Different implementations offer different ways of organizing and/or grouping documents:

- Collections
- Tags
- Non-visible metadata
- Directory hierarchies

Compared to relational databases, for example, collections could be considered analogous to tables and documents analogous to records. But they are different: every record in a table has the same sequence of fields, while documents in a collection may have fields that are completely different.

2.3 Graph

Main article: Graph database

This kind of database is designed for data whose relations are well represented as a graph (elements interconnected with an undetermined number of relations between them). The kind of data could be social relations, public transport links, road maps or network topologies, for example.

Graph databases and their query language

2.4 Object database

Main article: Object database

- db4o
- GemStone/S
- InterSystems Caché
- JADE
- NeoDatis ODB
- ObjectDatabase++
- ObjectDB
- Objectivity/DB
- ObjectStore
- ODABA
- Perst

- OpenLink Virtuoso
- Versant Object Database
- ZODB

2.5 Tabular

- Apache Accumulo
- BigTable
- Apache Hbase
- Hypertable
- Mnesia
- OpenLink Virtuoso

2.6 Tuple store

- Apache River
- GigaSpaces
- Tarantool
- TIBCO ActiveSpaces
- OpenLink Virtuoso

2.7 Triple/quad store (RDF) database

- AllegroGraph
- Apache JENA (It's a framework, not a database)
- MarkLogic
- Ontotext-OWLIM
- Oracle NoSQL database
- SparkleDB
- Virtuoso Universal Server
- Stardog

2.8 Hosted

- Amazon DynamoDB
- Amazon SimpleDB
- Datastore on Google Appengine
- Clusterpoint database
- Cloudant Data Layer (CouchDB)
- Freebase
- OpenLink Virtuoso

2.9 Multivalue databases

- D3 Pick database
- Extensible Storage Engine (ESE/NT)
- InfinityDB
- InterSystems Caché
- Northgate Information Solutions Reality, the original Pick/MV Database
- OpenQM
- Revelation Software's OpenInsight
- Rocket U2

2.10 Multimodel database

- OrientDB
- FoundationDB

3 Performance

Ben Scofield rated different categories of NoSQL databases as follows: [23]

Performance and scalability comparisons are sometimes done with the YCSB benchmark.

See also: Comparison of structured storage software

4 Handling relational data

Since most NoSQL databases lack ability for joins in queries, the database schema generally needs to be designed differently. There are three main techniques for handling relational data in a NoSQL database. (See table Join and ACID Support for NoSQL databases that support joins.)

4.1 Multiple queries

Instead of retrieving all the data with one query, it's common to do several queries to get the desired data. NoSQL queries are often faster than traditional SQL queries so the cost of having to do additional queries may be acceptable. If an excessive number of queries would be necessary, one of the other two approaches is more appropriate.

4.2 Caching/replication/non-normalized data

Instead of only storing foreign keys, it's common to store actual foreign values along with the model's data. For example, each blog comment might include the username in addition to a user id, thus providing easy access to the username without requiring another lookup. When a username changes however, this will now need to be changed in many places in the database. Thus this approach works better when reads are much more common than writes.[24]

4.3 Nesting data

With document databases like MongoDB it's common to put more data in a smaller number of collections. For example in a blogging application, one might choose to store comments within the blog post document so that with a single retrieval one gets all the comments. Thus in this approach a single document contains all the data you need for a specific task.

5 ACID and JOIN Support

If a database is marked as supporting ACID or joins, then the documentation for the database makes that claim. The degree to which the capability is fully supported in a manner similar to most SQL databases or the degree to which it meets the needs of a specific application is left up to the reader to assess.

6 See also

- CAP theorem
- Comparison of object database management systems
- Comparison of structured storage software
- Correlation database
- Distributed cache
- Faceted search
- MultiValue database
- Multi-model database
- Triplestore

7 References

[1] http://www.eventbrite.com/e/nosql-meetup-tickets-341739151 especially "Don't mind trading a query language for scalability? "

[2] Leavitt, Neal (2010). "Will NoSQL Databases Live Up to Their Promise?" (PDF). *IEEE Computer*.

[3] Mohan, C. (2013). *History Repeats Itself: Sensible and NonsenSQL Aspects of the NoSQL Hoopla* (PDF). Proc. 16th Int'l Conf. on Extending Database Technology.

[4] http://www.eventbrite.com/e/nosql-meetup-tickets-341739151 "Dynamo clones and BigTables"

[5] http://www.wired.com/2012/01/amazon-dynamodb/ "Amazon helped start the "NoSQL" movement."

[6] http://www.allthingsdistributed.com/2012/01/amazon-dynamodb.html "Customers like SimpleDB's table interface and its flexible data model. Not having to update their schemas when their systems evolve makes life much easier"

[7] "RDBMS dominate the database market, but NoSQL systems are catching up". DB-Engines.com. 21 Nov 2013. Retrieved 24 Nov 2013.

[8] "NoSQL (Not Only SQL)". NoSQL database, also called Not Only SQL

[9] Fowler, Martin. "NosqlDefinition". many advocates of NoSQL say that it does not mean a "no" to SQL, rather it means Not Only SQL

[10] Grolinger, K.; Higashino, W. A.; Tiwari, A.; Capretz, M. A. M. (2013). "Data management in cloud environments: NoSQL and NewSQL data stores" (PDF). JoCCASA, Springer. Retrieved 8 Jan 2014.

[11] Martin Zapletal: Large volume data analysis on the Typesafe Reactive Platform, ScalaDays 2015, Slides

[12] http://www.dummies.com/how-to/content/10-nosql-misconceptions.html "NoSQL databases lose data" section

[13] https://iggyfernandez.wordpress.com/2013/07/28/no-to-sql-and-no-to-nosql/

[14] Lith, Adam; Mattson, Jakob (2010). "Investigating storage solutions for large data: A comparison of well performing and scalable data storage solutions for real time extraction and batch insertion of data" (PDF). Göteborg: Department of Computer Science and Engineering, Chalmers University of Technology. p. 70. Retrieved 12 May 2011. Carlo Strozzi first used the term NoSQL in 1998 as a name for his open source relational database that did not offer a SQL interface[...]

[15] "NoSQL Relational Database Management System: Home Page". Strozzi.it. 2 October 2007. Retrieved 29 March 2010.

[16] "NoSQL 2009". Blog.sym-link.com. 12 May 2009. Retrieved 29 March 2010.

[17] Chapple, Mike. "The ACID Model".

[18] "DB-Engines Ranking". Retrieved 2015-07-31.

[19] Yen, Stephen. "NoSQL is a Horseless Carriage" (PDF). NorthScale. Retrieved 2014-06-26..

[20] Sandy (14 January 2011). "Key Value stores and the NoSQL movement". http://dba.stackexchange.com/questions/607/what-is-a-key-value-store-database: Stackexchange. Retrieved 1 January 2012. Key-value stores allow the application developer to store schema-less data. This data usually consists of a string that represents the key, and the actual data that is considered the value in the "key-value" relationship. The data itself is usually some kind of primitive of the programming language (a string, an integer, or an array) or an object that is being marshaled by the programming language's bindings to the key-value store. This structure replaces the need for a fixed data model and allows proper formatting.

[21] Seeger, Marc (21 September 2009). "Key-Value Stores: a practical overview" (PDF). http://blog.marc-seeger.de/2009/09/21/key-value-stores-a-practical-overview/: Marc Seeger. Retrieved 1 January 2012. Key-value stores provide a high-performance alternative to relational database systems with respect to storing and accessing data. This paper provides a short overview of some of the currently available key-value stores and their interface to the Ruby programming language.

[22] Katsov, Ilya (1 March 2012). "NoSQL Data Modeling Techniques". Ilya Katsov. Retrieved 8 May 2014.

[23] Scofield, Ben (2010-01-14). "NoSQL - Death to Relational Databases(?)". Retrieved 2014-06-26.

[24] "Making the Shift from Relational to NoSQL" (PDF). *Couchbase.com*. Retrieved December 5, 2014.

8 Further reading

- Sadalage, Pramod; Fowler, Martin (2012). *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Addison-Wesley. ISBN 0-321-82662-0.
- McCreary, Dan; Kelly, Ann (2013). *Making Sense of NoSQL: A guide for managers and the rest of us*. ISBN 9781617291074.
- Strauch, Christof (2012). "NoSQL Databases" (PDF).
- Moniruzzaman, A. B.; Hossain, S. A. (2013). "NoSQL Database: New Era of Databases for Big data Analytics - Classification, Characteristics and Comparison". arXiv:1307.0191.
- Orend, Kai (2013). "Analysis and Classification of NoSQL Databases and Evaluation of their Ability to Replace an Object-relational Persistence Layer". CiteSeerX: 10.1.1.184.483.
- Krishnan, Ganesh; Kulkarni, Sarang; Dadbhawala, Dharmesh Kirit. "Method and system for versioned sharing, consolidating and reporting information".

9 External links

- Strauch, Christoph. "NoSQL whitepaper" (PDF). Stuttgart: Hochschule der Medien.
- Edlich, Stefan. "NoSQL database List".
- Neubauer, Peter (2010). "Graph Databases, NOSQL and Neo4j".
- Bushik, Sergey (2012). "A vendor-independent comparison of NoSQL databases: Cassandra, HBase, MongoDB, Riak". NetworkWorld.
- Zicari, Roberto V. (2014). "NoSQL Data Stores – Articles, Papers, Presentations". *odbms.org*.

10 Text and image sources, contributors, and licenses

10.1 Text

- **NoSQL** *Source:* https://en.wikipedia.org/wiki/NoSQL?oldid=676116205 *Contributors:* AxelBoldt, Maury Markowitz, Jose Icaza, Pnm, Kku, Komap, Phoe6, Ronz, Ntoll, Ehn, Timwi, Furrykef, Phil Boswell, Bearcat, Peak, Dilbert, (:Julien:), Tagishsimon, Gadfium, Coldacid, Alexf, Beland, Euphoria, Clemwang, Rfl, MMSequeira, Smyth, Leigh Honeywell, Russss, Stephen Bain, Thüringer, Walter Görlitz, Markito~enwiki, Bhaskar, PatrickFisher, YPavan, Eno~enwiki, Crosbiesmith, Marasmusine, Woohookitty, Linas, Tshanky, Barrylb, Dm~enwiki, Tabletop, MacTed, Nileshbansal, BD2412, Qwertyus, Koavf, Ceefour, Strait, Amire80, Seraphimblade, ErikHaugen, Professionalsql, Vegaswikian, Jubalkessler, ElKevbo, Dmccreary, AlisonW, RobertG, Sstrader, Intgr, Tedder, Benbovee, Wavelength, Hairy Dude, Bovineone, Morphh, SamJohnston, Mbonaci, Rjlabs, Leotohill, Poohneat, GraemeL, Volt42, HereToHelp, Jonasfagundes, JLaTondre, Shepard, Matt Heard, Benhoyt, A bit iffy, SmackBot, Fulldecent, Anastrophe, Mauls, Drttm, Gorman, Somewherepurple, KiloByte, Thumperward, Jstplace, Jerome Charles Potts, Милан Јелисавчић, Frap, DavidSol, Cybercobra, Plustgarten, Looris, ThomasMueller, Trbdavies, NickPenguin, Eedeebee, ThurnerRupert, Petr Kopač, Zaxius, Lguzenda, Heelmijnlevenlang, Omidnoorani, Mauro Bieg, Benatkin, Mjresin, Hu12, Charbelgereige, Dancrumb, Gpierre, Arto B, Raysonho, Sanspeur, Ostrolphant, ProfessorBaltasar, Netmesh, Omer-Mor, Neustradamus, ColdShine, Mydoghasworms, Viper007Bond, Headbomb, CharlesHoffman, Peter Gulutzan, Davidhorman, Philu, Bramante~enwiki, Nick Number, Sorenriise, Polymorph self, Widefox, QuiteUnusual, Lfstevens, Gstein, Syaskin, Dericofilho, Joolean, Orenfalkowitz, Kgfleischmann, Philg88, Mitpradeep, Adtadt, GimliDotNet, Lmxspice, Stimpy77, Mikek999, DatabACE, JohnPritchard, Ansh.prat, McSly, Atropos235, Lamp90, Jottinger, Anoop K Nayak, Bbulkow, Tonyrogerson, Robert1947, Rogerdpack, Billinghurst, Quiark, Kbrose, ManikSurtani, TJRC, Dawn Bard, Whimsley, DavidBourguignon, Flyer22, Ctxppc, Mesut.ayata, Legacypath, AndrewBass, Edlich, Drq123, CaptTofu, Stevedekorte, Rossturk, Niceguyedc, Cnorvell, Pointillist, Excirial, Zapher67, PixelBot, Dredwolff, Robhughadams, Arjayay, Razorflame, StanContributor, Irmatov, Shijucv, The-verver, Tgrall, Miami33139, XLinkBot, Phoenix720, Duncan, Fiskbil, Whooym, Techsaint, Addbot, Fmorstatter, Mortense, Mabdul, MrOllie, LaaknorBot, Chrismcnab, Alexrakia, Getmoreatp, Luckasbot, Yobot, Amirobot, Pcap, Ebalter, Ma7dy, AnomieBOT, Angry bee, Fraktalek, White gecko, Materialscientist, Xtremejames183, Cyril Wack, Jabawack81, El33th4x0r, Gkorland, Tomdo08, ChristianGruen, FontOfSomeKnowledge, Rtweed1955, Omnipaedista, Sduplooy, Shadowjams, Ciges, Cekli829, Sdrkyj, FrescoBot, Nawroth, Ashtango, Sae1962, Thegreeneman5, David Paniz, Ertugka, Chenopodiaceous, Winterst, I dream of horses, Leegee23, Hoo man, Natishalom, Michael Minh, Seancribbs, Jandalhandler, Craigbeveridge, Cnwilliams, Colemala, Argv0, Justinsheehy, AdityaKishore, Javalangstring, Voodootikigod, JnRouvignac, Svesterli, Violaaa, Hoelzro, Magnuschr, Extrovrt101, Wyverald, Jeffdexter77, Uhbif19, Zond, Asafdapper, Ptab, Tobiasivarsson, Alexandre.Morgaut, Steve03Mills, Phunehehe, R39132, EmausBot, Biofinderplus, WikitanvirBot, FalseAxiom, Bdijkstra, Dewritech, GoingBatty, RA0808, Ledhed2222, MrWerewolf, EricBloch, Hloeung, ZéroBot, Weimanm, Al3xpopescu, Theandrewdavis, Mhegi, Sagarjhobalia, Mtrencseni, Phillips-Martin, Dmitri.grigoriev, H3llBot, Jnaranjo86, DamarisC, Dstainer, Bulwersator, Eco schranzer, Thomas.uhl, Lyoshenka, Inmortalnet, Really Enthusiastic, Germanviscuso, Stephen E Browne, ClueBot NG, Rabihnassar, Ki2010, Randl, Luisramos22, Fxsjy, Korrawit, Tylerskf, Castncoot, ScottConroy, Jrudisin, Mshefer, Ashtango5, Helpful Pixie Bot, Pereb, William greenly, Rpk512, GlobalsDB, DBigXray, Tuvrotya, BG19bot, Nawk, Gonim, Freshnfruity, Vychtrle, Gaborcselle, Kkbhumana, Frze, AvocatoBot, Mark Arsten, Compfreak7, Anne.naimoli, Matspca, Boshomi, Fceller, Dshelby, Brocsima, BigButterfly, Winston Chuen-Shih Yang, Griswolf, Socialuser, Ugurbost, BattyBot, Khiladi 2010, Noah Slater, Farvartish, Knudmoeller, Electricmuffin11, Mbarrenecheajr, Corrector623, Sandy.toast, F331491, Luebbert42, Anujsahni, Tsvljuchsh, Makecat-bot, Fitzchak, Toopathfind, Msalvadores, Cloud-dev, Sasindar, Zhanghaohit, CJGarner, Crosstantine, Stevenguttman, Razibot, DallasClarke, Altered Walter, Rediosoft, Tsm32, François Robere, Harpreet dandeon, LeeAMitchell, Mbroberg, Virendervermamca, Anilkumar1129, Mhgrove, FranzKraun, Jasonhpang, Nanolat, Nosql.analyst, Rzicari, Ginsuloft, Sugamsha, K0zka, Tshuva, Dodi 8238, Dabron, Mongochang, Natan.puzis, Webtrill, CafeNoName, Yasinaktimur, Monkbot, Itamar.haber, User db, Columbus240, Textractor, Maykurutu, Kamaci, Mongodbheavy, RedOctober13, Nathan194, Jjrenzel, Azanebrain, Annmkelly1, Sunnyeyre, Danny996, Thomas4019, Dr. Testificate, M.D., Johnbweeks, Magic-carpet-pilot, IdlePlayground, Datadatadatadata, Kevin at aerospike, Anywhichway, Nawazdhandala, The King Breaker, Mohammad.rafigh, Mgentz and Anonymous: 540

10.2 Images

- **File:Unbalanced_scales.svg** *Source:* https://upload.wikimedia.org/wikipedia/commons/f/fe/Unbalanced_scales.svg *License:* Public domain *Contributors:* ? *Original artist:* ?

10.3 Content license