

Википедия

NoSQL

Материал из Википедии — свободной энциклопедии

NoSQL (от англ. *not only SQL* — *не только SQL*) — термин, обозначающий ряд подходов, направленных на реализацию хранилищ баз данных, имеющих существенные отличия от моделей, используемых в традиционных реляционных СУБД с доступом к данным средствами языка SQL. Применяется к базам данных, в которых делается попытка решить проблемы масштабируемости и доступности за счёт атомарности (англ. *atomicity*) и согласованности данных (англ. *consistency*)^[1].

Содержание

Происхождение

- История названия

- Развитие идеи

Основные черты

Типы хранилищ данных

- Хранилище «ключ-значение»

- Хранилище семейств колонок (или Bigtable-подобные базы данных)

- Документоориентированная СУБД

- Базы данных на основе графов

UnQL

Примечания

Литература

Ссылки

Происхождение

История названия

Изначально слово NoSQL являлось акронимом из двух слов английского языка: No («Не») и SQL (сокращение от англ. *Structured Query Language* — «структурированный язык запросов»), что даёт термину смысл «отрицающий SQL». Возможно, что первые, кто стал употреблять этот термин, хотели сказать «No RDBMS» («не реляционная СУБД») или «no relational» («не реляционный»), но NoSQL звучало лучше и в итоге прижилось (в качестве альтернативы предлагалось также NonRel). Позднее для NoSQL было придумано объяснение «Not Only SQL» («не только SQL»). NoSQL стал общим термином для различных баз данных и хранилищ, но он не обозначает какую-либо одну конкретную технологию или продукт^[2].

Развитие идеи

Сама по себе идея нереляционных баз данных не нова, а использование нереляционных хранилищ началось ещё во времена первых компьютеров. Нереляционные базы данных процветали во времена мэйнфреймов, а позднее, во времена доминирования реляционных СУБД, нашли применение в специализированных

хранилищах, например, иерархических службах каталогов. Появление же нереляционных СУБД нового поколения произошло из-за необходимости создания параллельных распределённых систем для высокомасштабируемых интернет-приложений, таких как поисковые системы^[2].

В начале 2000-х годов Google построил свою высокомасштабируемую поисковую систему и приложения: GMail, Google Maps, Google Earth и т. п., решая проблемы масштабируемости и параллельной обработки больших объёмов данных. В результате была создана распределённая файловая система и распределённая система координации, хранилище семейств колонок (англ. *column family store*), среда выполнения, основанная на алгоритме MapReduce. Публикация компанией Google описаний этих технологий привела к всплеску интереса среди разработчиков открытого программного обеспечения, в результате чего был создан Hadoop и запущены связанные с ним проекты, призванные создать подобные Google технологии. Через год, в 2007 году, примеру Google последовал Amazon.com, опубликовав статьи о высокодоступной базе данных Amazon DynamoDB^[3].

Поддержка гигантов индустрии менее чем за пять лет привела к широкому распространению технологий NoSQL (и подобных) для управления «большими данными», а к делу присоединились другие большие и маленькие компании, такие как: IBM, Facebook, Netflix, eBay, Hulu, Yahoo!, со своими проприетарными и открытыми решениями^[3].

Основные черты

Традиционные СУБД ориентируются на требования ACID к транзакционной системе: атомарность (англ. *atomicity*), согласованность (англ. *consistency*), изолированность (англ. *isolation*), надёжность (англ. *durability*), тогда как в NoSQL вместо ACID может рассматриваться набор свойств BASE^[1]:

- базовая доступность (англ. *basic availability*) — каждый запрос гарантированно завершается (успешно или безуспешно).
- гибкое состояние (англ. *soft state*) — состояние системы может изменяться со временем, даже без ввода новых данных, для достижения согласования данных.
- согласованность в конечном счёте (англ. *eventual consistency*) — данные могут быть некоторое время рассогласованы, но приходят к согласованию через некоторое время.

Термин «BASE» был предложен Эриком Брюером, автором теоремы CAP, согласно которой в распределённых вычислениях можно обеспечить только два из трёх свойств: согласованность данных, доступность или устойчивость к разделению^[1].

Разумеется, системы на основе BASE не могут использоваться в любых приложениях: для функционирования биржевых и банковских систем использование транзакций является необходимостью. В то же время, свойства ACID, какими бы желанными они ни были, практически невозможно обеспечить в системах с многомиллионной веб-аудиторией, вроде amazon.com^[1]. Таким образом, проектировщики NoSQL-систем жертвуют согласованностью данных ради достижения двух других свойств из теоремы CAP^[4]. Некоторые СУБД, например, Riak, позволяют настраивать требуемые характеристики доступности-согласованности даже для отдельных запросов путём задания количества узлов, необходимых для подтверждения успеха транзакции.^[5]

Решения NoSQL отличаются не только проектированием с учётом масштабирования. Другими характерными чертами NoSQL-решений являются^{[6][7]}:

- Применение различных типов хранилищ^[6].
- Возможность разработки базы данных без задания схемы^{[6][7]}.
- Линейная масштабируемость (добавление процессоров увеличивает производительность)^[6].
- Инновационность: «не только SQL» открывает много возможностей для хранения и обработки данных^[6].

Типы хранилищ данных

Описание схемы данных в случае использования NoSQL-решений может осуществляться через использование различных структур данных: хеш-таблиц, деревьев и других.

В зависимости от модели данных и подходов к распределённости и репликации можно выделить четыре типа хранилищ: «ключ-значение» (key-value store), документно-ориентированные (document store), хранилища семейств колонок (column database), графовые базы данных (graph database).

Хранилище «ключ-значение»

Хранилище «ключ-значение» является простейшим хранилищем данных, использующим ключ для доступа к значению. Такие хранилища используются для хранения изображений, создания специализированных файловых систем, в качестве кэшей для объектов, а также в системах, спроектированных с прицелом на масштабируемость. Примеры таких хранилищ — Berkeley DB, MemcacheDB, Redis, Riak, Amazon DynamoDB^[6].

Хранилище семейств колонок (или Bigtable-подобные базы данных)

В этом хранилище данные хранятся в виде разреженной матрицы, строки и столбцы которой используются как ключи. Типичным применением этого вида СУБД является веб-индексирование, а также задачи, связанные с большими данными, с пониженными требованиями к согласованности данных. Примерами СУБД данного типа являются: Apache HBase, Apache Cassandra, Apache Accumulo, Hypertable, SimpleDB (Amazon.com)^{[6][8]}.

Хранилища семейств колонок и документно-ориентированные хранилища имеют близкие сценарии использования: системы управления содержимым, блоги, регистрация событий. Использование отметок времени (timestamp) позволяет использовать этот вид хранилища для организации счётчиков, а также регистрации и обработки различных данных, связанных со временем^[8].

Хранилища семейств колонок (англ. *column family stores*) не следует путать с колоночными хранилищами (англ. *column stores*). Последние являются реляционными СУБД с отдельным хранением колонок (в отличие от более традиционного построчного хранения данных)^{[8][9]}.

Документноориентированная СУБД

Документноориентированные СУБД служат для хранения иерархических структур данных. Находят своё применение в системах управления содержимым, издательском деле, документальном поиске и т. п. Примеры СУБД данного типа — CouchDB, Couchbase, MarkLogic, MongoDB, eXist, Berkeley DB XML^[6].

Базы данных на основе графов

Графовые базы данных применяются для задач, в которых данные имеют большое количество связей, например, социальные сети, выявление мошенничества. Примеры: Neo4j, OrientDB, AllegroGraph, Blazegraph^[10] (RDF-хранилище, ранее называлось Bigdata), InfiniteGraph, FlockDB, Titan^{[6][8]}.

Так как рёбра графа материализованы (англ. *materialized*), то есть, являются хранимыми, обход графа не требует дополнительных вычислений (как JOIN в SQL), но для нахождения начальной вершины обхода требуется наличие индексов. Графовые базы данных как правило поддерживают ACID, а также имеют различные языки запросов, вроде Gremlin и Cypher (Neo4j)^[8].

UnQL

В июле 2011 компания Couchbase, разработчик CouchDB, Memcached и Membase, анонсировала создание нового SQL-подобного языка запросов — UnQL (Unstructured Data Query Language). Работы по созданию нового языка выполнили создатель SQLite Ричард Гипп (англ. *Richard Hipp*) и основатель проекта CouchDB Дэмиен Кац (англ. *Damien Katz*). Разработка передана сообществу на правах общественного достояния^{[11][12][13]}.

Примечания

- ↑ Vaish, 2013, What NoSQL is and what it is not.
- ↑ Tiwari, 2011, Chapter 1: NoSQL: What It Is and Why You Need it > Definition and Introduction.
- ↑ Tiwari, 2011, pp. 4-6.
- ↑ Brewer, Eric A. A Certain Freedom: Thoughts on the CAP Theorem (http://portal.acm.org/ft_gateway.cfm?id=1835701&type=pdf&CFID=25475815) (англ.) // Proceeding of the IXXX ACM SIGACT-SIGOPS symposium on Principles of distributed computing. — N. Y.: ACM, 2010. — Iss. 29. — No. 1. — P. 335-336. — ISBN 978-1-60558-888-9. — DOI:10.1145/1835698.1835701 (https://dx.doi.org/10.1145%2F1835698.1835701).
- ↑ Zachary Kessin. Building Web Applications with Erlang. — O'Reilly Media, Inc., 2012. — P. 13. — 156 p. — ISBN 978-1-4493-0996-1.
- ↑ McCreary, Kelly, 2013, 1.1. What is NoSQL?.
- ↑ Vaish, 2013, Why NoSQL?.
- ↑ Curé, Blin, 2014.
- ↑ McCreary, Kelly, 2013, 4.3. Column family (Bigtable) stores.
- ↑ Blazegraph (Formerly Bigdata) (https://www.w3.org/2001/sw/wiki/Bigdata), w3c
- ↑ UnQL Query Language Unveiled by Couchbase and SQLite (http://www.couchbase.com/press-releases/unql-query-language)
- ↑ Welcome to the UnQL Specification home (http://www.unqlspec.org/display/UnQL/Home)
- ↑ Создатели CouchDB и SQLite представили UnQL, аналог SQL для систем NoSQL (http://www.opennet.ru/opennews/art.shtml?num=31346), новость на OpenNet

Литература

- Мартин Фаулер, Прамодкумар Дж. Садаладж.* NoSQL: новая методология разработки нереляционных баз данных = NoSQL Distilled. — М.: «Вильямс», 2013. — 192 с. — ISBN 978-5-8459-1829-1.
- Леонид Черняк* Смутное время СУБД (http://www.osp.ru/os/2012/02/13014107/) // Открытые системы. — 2012. — № 2.
- Dan McCreary, Ann Kelly.* Making Sense of NoSQL: A guide for managers and the rest of us. — Manning Publications, 2013. — 312 p. — ISBN 978-1-61729-107-4.
- Gaurav Vaish.* Getting Started with NoSQL. — Packt Publishing, 2013. — 142 p. — ISBN 978-1-84969-498-8.
- Shashank Tiwari.* Professional NoSQL. — Packt Publishing, 2011. — 384 p. — ISBN 978-0-470-94224-6.
- Olivier Curé, Guillaume Blin.* Chapter 2. Database Management Systems // RDF Database Systems: Triples Storage and SPARQL Query Processing. — Elsevier Science, 2014. — 256 p. — ISBN 978-0-12-800470-8.

Ссылки

- Matthew Aslett, Updated database landscape graphic (http://blogs.the451group.com/information_management/2012/11/02/updated-database-landscape-graphic/), November 2nd, 2012 (диаграмма)

Источник — https://ru.wikipedia.org/w/index.php?title=NoSQL&oldid=93693110

Эта страница последний раз была отредактирована 1 июля 2018 в 19:34.

Текст доступен по лицензии Creative Commons Attribution-ShareAlike; в отдельных случаях могут действовать дополнительные условия.

Wikipedia® — зарегистрированный товарный знак некоммерческой организации Wikimedia Foundation, Inc.

Свяжитесь с нами