

**НЕРЕЛЯЦИОННЫЕ СИСТЕМЫ
ХРАНЕНИЯ В УСЛОВИЯХ
ПРОБЛЕМЫ БОЛЬШИХ ДАННЫХ
И РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ****NON-RELATIONAL
STORAGE SYSTEMS FOR
BIG DATA AND DISTRIBUTED
COMPUTING**

Аннотация. Данная статья отражает проблему больших данных относительно систем управления базами данных и распределенных информационных систем, ставшую на сегодняшний день ключевой в ИТ-индустрии.

Ключевые слова: большие данные; реляционные БД; транзакционные БД; аналитические БД; распределенные системы; NoSQL; BASE; CAP; ACID; синхронизация.

Сведения об авторе: Шалтунович Анна Викторовна, ассистент кафедры информатики и методики преподавания информатики.

Место работы. Нижевартовский государственный гуманитарный университет.

Контактная информация: 628611, г. Нижневартовск, ул. Дзержинского, д. 11; тел. (3466) 454403.
E-mail: shaltunovich.anna@gmail.com

Abstract. This article reviews the problem of Big Data in regard to database management systems and distributed systems which has become a key issue in today's IT industry.

Key words: Big Data; relation databases; transaction db; analytical db; distributed computing; NoSQL; BASE; CAP; ACID; synchronization.

About the author: Anna Victorovna Shaltunovich, assistant of the Department of Informatics and its Teaching Methodology.

Place of employment: Nizhnevartovsk State University of Humanities.

Федеральная целевая программа Российской Федерации «Информационное общество 2011—2020» ориентирована на укрепление статуса информационно и телекоммуникационно развитой державы. При этом акцент долгосрочного развития делается на поддержку слаборазвитых систем связи, таких как почтовые службы, телеграфные, системы телевизионного и радиовещания, а также на дальнейшее ускоренное развитие и повсеместное внедрение компьютерных технологий в повседневную жизнь российского общества. Главной задачей является перенос всех доступных населению услуг по работе с административно-управленческими инстанциями в формат электронного предоставления услуг посредством глобальной сети Интернет из любой точки нашей страны.

Однако при этом возникает проблема перенасыщения информацией о пользователях, зарегистрированных сразу в нескольких системах (налоговая инспекция, портал государственных услуг, портал медицины и здравоохранения, образования и т.д.). Базы данных, поддерживающие работу данных информационных систем, будут дублировать одну и ту же информацию. Кроме того стоит отметить еще и масштабы нашей страны, а также популярность информационных ресурсов, что способствует возникновению проблемы по работе с так называемыми большими данными.

Проблема больших данных, ставшая на сегодняшний день ключевой в области систем управления базами данных, представляет собой огромную избыточность хранения, проявившуюся в многократном дублировании одной и той же информации на различных носителях, а также увеличение объемов устаревшей, неактуальной и недостоверной информации. Единственным и наиболее популярным способом борьбы с избыточным хранением и нерациональным использованием дискового пространства являются методы дедупликации данных. Однако дедупликация ориентирована прежде всего на работу с файловой системой, что в современном господстве реляционных баз данных не позволяет эффективно бороться с избыточностью.

Традиционные реляционные СУБД реализуют эффективное управление транзакционными и аналитическими БД. Транзакционные предназначены для поддержки оперативных

динамичных приложений (торговые системы, системы резервирования, управления бизнес-процессами и т.д.), работающих с данными, которые отражают текущее положение дел в той или иной сфере деятельности. Аналитические базы содержат исторически подтвержденные данные, поступающие из различных источников, в том числе и транзакционных БД.

С проблемой больших данных сталкиваются обе категории баз данных. При этом объемы транзакционных БД увеличиваются за счет постоянного роста количества пользователей сети Интернет, их активности и потребностей в использовании информационных ресурсов телекоммуникационных сетей. Особенно возрастают объемы БД при персонализации пользователей. Аналитические БД в свою очередь всегда лишь накапливают информацию, не уничтожая ее, осуществляя тем самым доступ как к более новым, так и к устаревшим данным.

Для транзакционных баз частный случай проблемы больших данных можно сформулировать следующим образом: «нужно обеспечить технологию относительно недорогого масштабирования СУБД и транзакционных приложений, позволяющую поддерживать требуемую скорость обработки транзакций при росте объема данных и увеличении числа одновременно выполняемых транзакций» [1. С. 22]. Для аналитических баз частный случай проблемы звучит так: «требуется обеспечить технологию относительно недорогого масштабирования СУБД и аналитических приложений, позволяющую аналитикам расширять возможности СУБД по выполнению аналитических запросов и обеспечивать эффективную оперативную аналитическую обработку данных при росте их объема» [1. С. 22].

В первом десятилетии XXI в. исследователям во главе с Майклом Стоунбрейкером удалось обозначить пути решения данных проблем, используя следующие принципы:

1. Минимизация времени обмена данными между узлами системы и, как следствие, перенос приложений баз данных на сторону сервера.
2. Распараллеливание работы СУБД и приложений, в частности, отказ от архитектуры совместных данных.
3. Распределение данных по узлам системы с возможностью их репликации на нескольких узлах, что обеспечивает эффективную параллельную обработку транзакций и поддержку оперативной аналитической обработки данных.

По мнению А.Босуорта, неизбежным является «переход от складирования данных к их распределению» [3. С. 17]. Кроме того, внедрение технологий распределенных вычислений обусловлено развитием интернет-сервисов (DNS-серверы, поисковые машины, социальные сети и т.д.), которые отвечают за обработку огромных массивов информации и пользовательских запросов.

Распределенные системы представляют собой единый комплекс информационных и вычислительных ресурсов, решающий задачи обработки и хранения данных путем объединения удаленных и разобщенных вычислительных центров.

В основу распределенных вычислений положены два принципа:

1. Способность системы поддерживать организационно и физически распределенных пользователей, одновременно работающих с общими данными.
2. Способность системы поддерживать логически и физически распределенные данные, составляющие единую базу данных. При этом необходима система управления распределенными базами данных, обеспечивающая интеграцию локальных БД к единой БД, с предоставлением пользователям системы прозрачности, целостности данных, распределенной обработки, асинхронного дублирования, фрагментации, оптимизации и выполнения распределенных запросов, средств администрирования и защиты, возможности взаимодействия.

Однако созданию современных распределенных систем препятствует теорема CAP (Consistence, Availability, Partition tolerance). Теорема, которую также называют теоремой

Брюера, определяется как невозможность одновременно в распределенной вычислительной системе выполнять требования по согласованности данных, доступности системы и устойчивости к разделению [2. С. 27], где согласованность считается мгновенной (набор операций завершается одновременно); распределенная система считается доступной, если на каждый запрос получен ответ; устойчивость к разделению означает сохранение жизнеспособности системы при отказе некоторых узлов.

Приведем нестрогое доказательство теоремы CAP. Пусть распределенная система состоит из N серверов, каждый из которых обрабатывает запросы M клиентских приложений. Обрабатывая запрос, сервер должен гарантировать актуальность предоставляемой в ответе информации, для чего необходимо синхронизировать содержимое его собственной базы с другими серверами. Таким образом, сервер может реагировать тремя возможными способами:

1. ждать полной синхронизации с остальными вычислительными узлами;
2. формировать ответ на базе несинхронизированных данных;
3. синхронизировать данные лишь с некоторыми вычислительными узлами.

В первом случае, соответственно, не выполняется требование доступности, во втором — согласованности, в третьем — устойчивости к разделению.

Требования, предъявляемые к распределенной системе, объединяются в набор BASE-свойств:

1. Basically Available — система находится в состоянии готовности, но не всегда;
2. Soft State — система может выйти из заданного состояния (мягкое состояние, противопоставляемое Hard State);
3. Eventually Consistent — узлы распределенной системы согласованы, но не всегда.

BASE-свойства используются для наиболее общего описания требований к распределенным системам, попадающим под утверждение теоремы CAP и не удовлетворяющим требованиям ACID (атомарность, согласованность, изолированность, долговечность). Кроме того, стоит отметить, что условия CAP сформулированы абсолютно в ином технологическом окружении, чем предложенные Джимом Греем в 70-е гг. XX в. ACID-свойства, когда представление о современных СУБД складывалось относительно только централизованных вычислительных систем.

По видам выполняемых CAP-требований выделяют системы:

1. CA, удовлетворяющие требованиям по согласованности и доступности;
2. CP, удовлетворяющие требованиям по согласованности и устойчивости;
3. AP, удовлетворяющие требованиям по доступности и устойчивости к разделению.

При этом ни в одном из трех случаев не будут соблюдаться ACID-свойства реляционных БД.

На сегодняшний день одной из наиболее важных характеристик информационных систем является способность к масштабированию, в том числе и распределенных систем хранения. Масштабируемость — это способность системы справляться с увеличением рабочей нагрузки путем увеличения своей производительности, при добавлении в систему новых вычислительных ресурсов. Масштабируемость — важное требование, предъявляемое к распределенным информационным системам и базам данных, если для них требуется возможность работать под большой нагрузкой и постоянном увеличении компонентов системы.

Главное достоинство реляционных баз данных состоит в технологиях, обеспечивающих надежное хранение данных с минимизацией резервирования. Однако при этом те же самые технологии становятся препятствием к масштабированию больших объемов петабайтных данных. Ведущими лидерами в сфере информационных технологий, такими как Amazon, Google, создаются собственные нереляционные системы хранения, способные справляться с данными объемами информации. Масштабирование в таких системах достигается за счет использования идентичных серверов, по которым данные распространяются

сегментированно. Возникающие изменения распространяются асинхронно, поэтому согласованность достигается, но не сразу, что соответствует одному из положений теоремы CAP — Eventually Consistent.

Переход к использованию подобных баз данных связывают с так называемым движением NoSQL. В качестве основных характеристик, отличающих системы NoSQL, можно выделить:

1. способность к горизонтальному масштабированию большого числа серверов распределенной вычислительной системы;
2. способность реплицировать и распределять данные по большому числу серверов;
3. возможность ослабления требований согласованности по сравнению с ACID;
4. эффективное использование дискового пространства;
5. способность динамически наращивать атрибуты хранимых данных.

При этом данные системы ориентированы прежде всего на работу с неструктурированными данными, поступающими от многочисленных пользователей сети Интернет в онлайн режиме. Базовым структурным взаимодействием является связь «ключ-значение». Примеры систем хранения типа NoSQL: Dynamo, BigTable, Apache Cassandra, HBase, Hypertable, Memcached. Каждая из систем демонстрирует преимущества тех или иных технологических решений, например, Memcached использует индексы в оперативной памяти (in-memory indexes) как средства для масштабирования, распределений и реплицирования данных по узлам; Dynamo была первой в согласованности типа Eventually Consistency, то есть с негарантированной актуальностью данных в текущий момент, но с гарантией, что обновление данных будет со временем распространено по узлам; BigTable дает возможность распространения неизменных записей (persistent record) по тысячам узлов [3. С. 19].

Влияние проблемы больших данных показывает неподготовленность современных информационных технологий справляться с объемами неструктурированной информации, хранящихся сегодня, в большинстве случаев, в реляционных базах данных. В качестве альтернативы реляционным базам данным ИТ-сообщество предлагает движение к NoSQL-системам хранения, позволяющее обеспечивать возможность работы на базе распределенных технологических решений.

ЛИТЕРАТУРА

1. Кузнецов С. К свободе от проблемы больших данных // Открытые системы. СУБД. 02/2012.
2. Селезнев К. От SQL к NoSQL и обратно // Открытые системы. СУБД. 02/2012.
3. Черняк Л. Смутное время СУБД // Открытые системы. СУБД. 02/2012.