УДК 004.65

ВЫБОР БАЗЫ ДАННЫХ ДЛЯ ХРАНЕНИЯ ДАННЫХ ВРЕМЕННЫХ РЯДОВ

Люкина С.Ю., студент гр. ПИм-211, I курс

Научный руководитель: Гиниятуллина О.Л., к.т.н., доцент

Кузбасский государственный технический университет

имени Т.Ф. Горбачёва

г. Кемерово

В настоящее время широкий спектр задач подразумевает необходимость сбора и хранения больших объемов данных, использующихся для аналитики и прогнозирования. К ним относятся, например, данные с различных сенсоров, значения курса акций или курса валюты, данные о посещаемости сайтов, об объемах продаж и т.д. Такая последовательность статистических данных, собранных в разные моменты времени, о значении каких-либо параметров исследуемого процесса называется временным рядом или рядом динамики. Благодаря анализу таких данных можно узнавать исторические тенденции, обнаруживать аномалии, выполнять прогнозное моделирование изучаемого явления и т.д. Показатели для составления временных рядов могут быть техническими, экономическими, социальными и даже природными [3].

Основная цель анализа временного ряда – построить прогноз его значений на будущие периоды, поэтому при анализе временного ряда исследователей интересуют не только его статические характеристики, но и взаимосвязь измерений со временем [1]. Как правило, такие данные поступают и хранятся в строгой очередности и имеют метку времени, а их сохранение выполняется как вставка, а не обновление. Это позволяет отображать события в порядке их возникновения и сохранить их естественную хронологию, что, в свою очередь, позволяет анализировать прошедшие периоды и прогнозировать изменения на будущие. Например, данные телеметрии с датчиков промышленного оборудования можно использовать для определения возможных сбоев и уведомления, данные курса акций для определения дальнейшего направления развития и т.д. Таким образом, время является самой значимой осью для просмотра и анализа данных временных рядов. Хранение данных в порядке их возникновения - главное отличие данных временных рядов [2].

Так как данные временных рядов меняются с учетом времени, то зачастую они отличаются очень большими объемами, особенно при сборе из множества разных источников, таких как датчики. Так, например, хранение

данных, собранных устройствами Интернета вещей, их индексирование, выполнение запросов и визуализация могут представлять определенные трудности, а задержки в анализе могут повлечь простои и повлиять на бизнес. В связи с этим необходимо правильно подбирать высокоскоростное хранилище и использовать мощные вычислительные операции для анализа данных в реальном времени, что может оказаться сложной задачей. Однако это позволит сократить время выхода на рынок и уменьшить общую стоимость инвестиций [2].

Рассмотрим различные виды баз данных, используемых для хранения данных временных рядов. Ключевыми характеристиками при анализе инструментов для работы с данными временных рядов будут скорость приема данных, организация их хранения, время обработки запросов и скорость ответа на них. Так как данные факторы в дальнейшем могут влиять на качество анализа и прогнозирования временных рядов [6].

Наиболее распространенными являются реляционные или SQL базы данных. Данные в них хранятся в виде набора взаимосвязанных таблиц, состоящих из строк и столбцов. В таблицах хранится информация об объектах, представленных в базе данных. Каждая строка таблицы представляет собой набор связанных значений, относящихся к одному объекту или сущности. Значения атрибутов объекта хранятся в ячейках таблицы, при этом в каждом ее столбце хранится определенный тип данных. Каждая строка в таблице может быть помечена уникальным идентификатором, называемым первичным ключом, а строки из нескольких таблиц могут быть связаны с помощью внешних ключей. Использование ключей дает возможность легко устанавливать взаимосвязь между элементами и быстро получать доступ к необходимым данным [4]. Примерами реляционных баз данных могут быть MySQL, MSSQL, PostgreSQL, Oracle и другие. SQL базы данных часто используются при разработке сайтов, в корпоративных приложениях, для отслеживания товарных запасов, обработки торговых транзакций через Интернет, управления большими объемами данных заказчика т.д. Такие базы данных подходят для обслуживания любых информационных потребностей, где элементы данных связаны между собой и необходимо обеспечивать безопасное и надежное управление ими на основе правил целостности.

Одной из фундаментальных основ реляционной модели является понятие отношения - неупорядоченного множества кортежей, в то время как принципиальным свойством временного ряда является упорядоченность его элементов. Данный факт свидетельствует о плохой совместимости основ реляционной модели и природы временного ряда и является одной из причин неэффективной работы таких баз данных с временными рядами. Однако некоторые производители реляционных баз данных вводят в свои продукты средства оптимизации работы с упорядоченными последовательностями. В результате этого, широкое распространение получили кластерные индексы, при построении которых записи на диске упорядочиваются в соответствии со значениями ключей индекса. Указанный подход позволяет оптимизировать доступ к данным, но при этом проблемы применения реляционной модели к временным

рядам полностью не решаются [5]. Наличие индексов в таблицах реляционных баз данных делает их медленными при росте объема данных. При добавлении новых записей в базу данных и при наличии в таблице индексов, СУБД будет многократно переиндексировать данные для быстрого и эффективного доступа к ним, что, в свою очередь, со временем приведет к снижению производительности, увеличению нагрузки, появлению трудностей при чтении данных. Таким образом, реляционные базы данных не способны в полной мере обеспечить эффективное хранение и обработку временных рядов [8].

Другим видом баз данных, который можно использовать для хранения временных рядов, являются NoSQL базы данных. Базы данных NoSQL появились значительно позже реляционных баз данных, а также существенно отличаются от них по структуре хранения и принципам работы с данными. Такие базы данных не имеют в своей основе реляционной модели и не используют язык SQL. Среди NoSQL баз данных выделяют системы «ключ - значение», документно-ориентированные СУБД, графовые СУБД и другие. NoSQL-системы отказываются от поддержки механизма транзакций, поддерживаемого в реляционных СУБД, для обеспечения масштабируемости системы и доступности данных при высоких нагрузках в распределенных системах обработки данных [7]. Среди популярных NoSQL баз данных можно выделить Cassandra, HBase, а также более специализированные решения, например, OpenTSDB, KairosDB и Acunu. На концептуальном уровне, NoSQL как подход предполагает выбор модели данных в зависимости от задачи и провозглашает отказ от единой модели данных. NoSQL базы данных применяются чаще всего не для хранения всех данных приложения, а лишь для решения специфических задач, например, кэширование, журналирование, распределённое хранение данных, очереди заданий, и поэтому менее распространены в простых проектах. Также большинство таких баз данных работает на базе инфраструктуры Hadoop, и для их нормального функционирования требуется огромное количество зависимостей, что также влияет на производительность системы [9].

В последнее время все большую популярность набирают базы данных временных рядов или Time Series Database (TSDB). База данных временных рядов – это программная система, оптимизированная для хранения и обслуживания временных рядов через связанные пары времени и значения. Она представляет собой специализированное решение для хранения и обработки данных временных рядов. Данные в них хранятся в «коллекциях», агрегированных со временем, это означает, что для каждой «точки», которую необходимо сохранить, есть связанная с ней временная метка. Базы данных временных рядов позволяют пользователям создавать, считывать, обновлять и удалять данные [10]. Простой синтаксис также является их преимуществом. Системы баз данных временных рядов построены на принципе, что им нужно быстро и эффективно принимать данные. Они оптимизированы для индексации данных, которые агрегируются во времени. Вследствие этого, скорость загрузки не уменьшается со временем и остается достаточно стабильной. Так как объем данных временных рядов может достаточно быстро увеличиваться, чтобы не хранить большое

количество старых данных, в базах данных временных рядов предусмотрена функция их удаления через определенное время. Для этого используется концепция, называемая политикой хранения. Это позволяет сократить затраты на содержание большого объема данных и не хранить данные, которые со временем уже утратили свою актуальность. Базы данных временных рядов часто используются в Интернете вещей, так как они были созданы под большое количество клиентов, что позволяет множеству сенсоров одновременно выполнять вставку данных [8]. Сейчас на рынке среди баз данных временных рядов, преобладают продукты с открытым исходным кодом: Open TSDB, InfluxDB, Geras, Druid и другие. Существуют также решения для узких спектров задач, например, инструменты для специалистов в области веб-технологий - YAWNDB и SiteWhere [10].

Таким образом, проанализировав различные варианты баз данных, можно заключить, что для хранения данных временных рядов лучше использовать специализированные решения, представляющие собой базы данных временных рядов. Их использование является более эффективным, так как базы данных временных рядов обеспечивают более высокую скорость приема данных и их обработки по сравнению с реляционными базами данных или базами данных NoSQL.

Список литературы:

1. Анализ и модели временных рядов [Электронный ресурс]. – Режим доступа: https://www.statmethods.ru/statistics-metody/modeli-vremennykh-ryadov/ (дата обращения: 27.03.2022).

2. Данные временных рядов [Электронный ресурс]. – Режим доступа: https://docs.microsoft.com/ru-ru/azure/architecture/data-guide/scenarios/time-series (дата обращения: 28.03.2022).

3. Временной ряд [Электронный ресурс]. – Режим доступа: https://blog.skillfactory.ru/glossary/vremennoj-ryad-2/ (дата обращения: 28.03.2022).

4. Что такое реляционная база данных? [Электронный ресурс]. – Режим доступа: https://aws.amazon.com/ru/relational-database/ (дата обращения: 29.03.2022).

5. Использование объектно-реляционных СУБД для хранения и анализа временных рядов [Электронный ресурс]. – Режим доступа: https://compress.ru/article.aspx?id=10917#07 (дата обращения: 29.03.2022).

6. Волкова, С. В. Сравнительный анализ возможностей реляционных, графовых и циклических СУБД для хранения и обработки данных временных рядов[Электронный ресурс] / С. В. Волкова // Вестник современных исследований. – 2020. - №1-7(31). - Режим доступа: https://elibrary.ru/item.asp?id=42946599 (дата обращения: 28.03.2022).

7. Иванова, Е. В. Обзор современных систем обработки временных рядов [Электронный ресурс] / Е. В. Иванова, М. Л. Цымблер // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика – 2020. - Режим доступа: https://cyberleninka.ru/article/n/obzor-sovremennyh-sistem-obrabotki-vremennyh-ryadov (дата обращения: 29.03.2022).

8. База данных временных рядов компании InfluxData [Электронный ресурс]. – Режим доступа: https://waksoft.susu.ru/2019/10/07/%D0%B1%D0%B0%D0%B7%D0%B0-%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85-%D0%B2%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D1%8B%D1%85-%D1%80%D1%8F%D0%B4%D0%BE%D0%B2-%D0%BA%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D0%B8%D0%B8-inf/ (дата обращения: 27.03.2022).

9. Реляционные базы данных и NoSQL - хранилища [Электронный ресурс]. – Режим доступа: https://web-creator.ru/articles/about_databases (дата обращения: 29.03.2022).

10. Что такое базы данных временных рядов (time series database) [Электронный ресурс]. – Режим доступа: https://www.xelent.ru/blog/chto-takoe-bazyi-dannyih-vremennyih-ryadov--time-series-database-/ (дата обращения: 29.03.2022).