

0.1. *Лузьянин А.В.* Методы оптимизации и хранения больших объёмов данных

В настоящее время одной из актуальных проблем, требующих решения, является проблема хранения больших данных и работа с ними. Традиционно для хранения больших объёмов данных используется реляционная база данных, использующая язык структурированных запросов SQL (Structured Query Language). Такая организация данных позволяет относительно легко и быстро оперировать огромными объёмами данных. Однако существуют задачи, для которых организация данных в реляционной базе данных не является оптимальной: в случае, когда она используется для хранения однотипных данных, таких как параметры устройств. Среди этих данных может присутствовать большое количество дубликатов, хотя каждая структурная единица этих данных будет уникальной и предназначена для конкретного устройства. Таким образом, база данных разрастается за счёт дубликатов и с течением времени становится всё тяжелее управлять этим объёмом данных, растут затраты на их хранение.

В данном докладе исследуемая система представляет собой реляционную базу данных, содержащую в себе среди прочего данные о том, какие параметры должны быть применены к конечному устройству, запрашивающему с сервера необходимое для своей работы приложение и параметры в виде модели данных (XML).

Проблема существующей структуры хранения данных заключается в том, что она порождает огромное количество фактических дубликатов данных, хранящихся в системе. Например, существует 2 уникальных значения параметра в системе: «On» и «Off». Если количество устройств, использующих данный параметр, равно 1000, то в базе будет храниться 1000 экземпляров данных значений. Таким образом, вместо хранения 2 уникальных значений параметра, каждый параметр хранится как отдельная сущность. Существуют шаблоны параметров устройств, которые позволяют снизить количество хранимых данных, однако при изменении параметра одного конкретного устройства требуется хранить отдельный экземпляр той же сущности. Очевидно, что решение неоптимально и влечёт за собой высокие затраты на обслуживание, управление и хранение данных.

Данную структуру можно оптимизировать, проведя сжатие данных до экземпляров уникальных значений, а также создав способ организации связи и редактирования хранимых данных. Это решение можно реализовать с использованием как SQL, так и NoSQL баз данных. Поскольку структура данных является иерархической, было принято решение использовать для их хранения графоориентированную СУБД [1]. Для этого была проведена оптими-

зация хранимых данных, осуществлён их перенос из реляционной базы данных в графоориентированную, а также создана стандартная система для манипуляции [2] этими данными. Таким образом, удалось снизить хранимый объём данных примерно на 85% (с 4 Гбайт до 600 Мбайт), сохранив базовую логику работы системы и снизив затраты на её обслуживание.

Список литературы

- [1] Alabdullah B., Beloff N., White M. Rise of Big Data — Issues and challenges // 2018 21st Saudi Computer Society National Computer Conference (NCC). Riyadh, Saudi Arabia, 2018. Vol. 1. P. 1–6.
- [2] Gyssens M., Paredaens J., van den Bussche J. A graph-oriented object database model // IEEE Transactions on Knowledge and Data Engineering. 1994. Vol. 6. P. 572–586.