

УДК 004.7

А.В. Сви́рин

Одеський національний політехнічний університет, пр. Шевченка, 1, м. Одеса, 65044

## ИСПОЛЬЗОВАНИЕ ВЫЧИСЛИТЕЛЬНОГО СЕРВЕРА В ФЕДЕРАТИВНОЙ ГЕТЕРОГЕННОЙ НЕОДНОРОДНОЙ SQL-СИСТЕМЕ

В настоящей работе предложены методы, позволяющие улучшить характеристики гибкости и производительности федеративной гетерогенной неоднородной распределенной базы данных за счет обеспечения гетерогенных свойств узлов и оптимизации выполнения повторяющихся распределенных SQL-запросов.

**Ключевые слова:** Федеративные РБД - гетерогенные РБД - SQL-системы - Шлюзы доступа - Вычислительный сервер

In this paper we propose methods to improve the flexibility and performance characteristics of heterogeneous federated inhomogeneous distributed database by providing the nodes heterogeneous properties and optimizing the recurring distributed SQL-queries performance.

**Keywords:** Federated SRM – Heterogeneous SRM – SQL-system – Access gateways – Computing Server

### I. ВВЕДЕНИЕ

Использование неоднородной реляционной распределенной базы данных (РБД) в качестве информационной системы дает ряд преимуществ, а именно: *устойчивость* – в случае, если в РБД часть из узлов вышла из строя, то остальные узлы обеспечивают частичную работоспособность РБД; *масштабируемость* и *производительность* – РБД распределена по узлам вертикальным либо горизонтальным способом, и на каждом узле можно выполнять части распределенного запроса или локальные запросы; *гибкость* и *совместимость* – на узлах могут использоваться различные системы управления базами данных (СУБД), которые обладают гетерогенными свойствами по отношению к остальным узлам. Гетерогенное свойство РБД является ключевым свойством для использования федеративных РБД [120].

В федеративную РБД могут быть включены базы данных, в которых хранится частичная информация о глобальной схеме данных и в которых достаточно данных для полноценного их функционирования. В такой SQL-системе могут сосуществовать схемы данных, которые ранее не взаимодействовали между собой и развивались независимо друг от друга.

На рисунке 1 показано, что существует две не взаимосвязанные между собой гомогенные реляционные РБД: «Реестр сотрудников» с СУБД *MySQL* и «Реестр библиотеки» с СУБД *PostgreSQL*. В последствии появилась необходимость создания реляционной РБД «Реестр научных деятелей» с СУБД *Firebird* и, в результате, образуется федеративная гетерогенная РБД, которая включает в себя две частичные схемы данных и одну свою схему данных.



Рисунок 1 – Пример федеративной гетерогенной РБД

В гетерогенной РБД, для обеспечения необходимых качеств в узлах, требуется использование СУБД определенных производителей и версий. Использование такой СУБД на узле зависит от

наличия свойства гетерогенности у нее. К сожалению, на сегодняшний день, большинство производителей СУБД не обеспечивают такое свойство по той причине, что для этого требуется совмести-

мость не только с СУБД сторонних фирм-разработчиков, но и с версиями. Чаще всего такую проблему решают использованием на узлах программ **шлюзов** типа «точка-точка» [120], которые преобразовывают синтаксис определенных сообщений и результаты их выполнения для определенной SQL-системы в синтаксис сообщений также определенной другой SQL-системы. В результате, на узле может потребоваться использовать необходимое количество шлюзов для взаимодействия с другими узлами гетерогенной РБД. Очевидно, что при попытках предоставить полную независимость от СУБД возникают значительные технические сложности, даже если все участвующие СУБД являются системами SQL.

В РБД может потребоваться выполнять запросы конкатенации на ВС с некоторой периодичностью, с целью обновления в первый раз полученных данных. Назовем такие запросы *повторяющимися*. Решение такой задачи будет эффективнее, если с узлов на ВС каждый раз передавать не все данные, а лишь данные об изменениях. Встроенные средства большинства СУБД позволяют выполнять тиражирование данных, но недостатком такого подхода является то, что настроить ведение журнализации для тиражирования у отдельных таблиц или доменов таблиц нет возможности, а есть лишь для всей БД.

Анализ приведенных выше проблем использования федеративных гетерогенных неоднородных SQL-систем на примере РБД, позволил поставить цель данной работы – улучшение характеристик *гибкости* и *производительности* федеративной гетерогенной неоднородной РБД за счет решения задач: обеспечение гетерогенных свойств узлов в РБД; оптимизация выполнения повторяющихся распределенных SQL-запросов.

## II. ОСНОВНАЯ ЧАСТЬ

Исследования показали, что основная нагрузка в реляционной БД возникает при выполнении конкатенации данных таблиц, особенно если такие таблицы находятся на разных узлах РБД. При конкатенации данных в РБД первичной таблицы со вторичными необходимо передавать данные таблиц из вторичных узлов столько раз, сколько имеется шагов в запросе конкатенации. Это происходит по той причине, что первичный узел не может знать достоверно – менялись ли данные в таблицах на вторичных узлах за то время, пока выполняется шаг запроса конкатенации. Следует отметить, что при выполнении запросов конкатенации нет возможности использовать встроенные оптимизирующие алгоритмы кеширования данных на вторичных узлах, по той причине, что вычисления происходят на первичном узле.

### Метод обеспечения гетерогенных свойств узлов в РБД

Предложен метод использования *вычисли-*

*тельного сервера* (ВС), одной из дополнительных функций которого будет обеспечение гетерогенных свойств в РБД. Использование ВС с шлюзами дополнит структуру РБД, обеспечит гетерогенные свойства и при этом не потребует дополнительного вмешательства в функционирующую систему. Для того, чтобы ВС был способен выполнять запросы на всех СУБД в узлах РБД, требуется на нем использовать шлюзы для любых SQL-ориентированных СУБД, которые используются на узлах РБД. Таким образом в узлах появится возможность выполнять распределенные запросы к другим узлам посредством ВС с шлюзами, без необходимости дополнительной конфигурации, не заботясь о том, какая СУБД используется на вторичных узлах.

Основной функцией ВС остается выполнение вычислительных операций в РБД. Для выполнения запросов конкатенации над временно-перенесенными таблицами на ВС, следует: использовать на ВС БД под управлением SQL-ориентированной СУБД, подходящей для шлюзов; обеспечить ВС достаточным объемом вычислительных ресурсов. Использование ВС подразумевает, что пользователи всех узлов РБД могут выполнять распределенные запросы с использованием ВС, поэтому следует организовать очередь выполнения запросов, ориентированную на одновременное обслуживание простых и сложных запросов.

Каждый запрос конкатенации следует выполнять по шагам. Промежуток времени ожидания выполнения шага запроса предложено вычислять по формуле:

$$T = t + n * t$$

где  $T$  – время ожидания,  $n$  – номер шага,  $t$  – время выполнения шага (одинаковое для всех запросов на ВС). Таким образом сложные запросы смогут параллельно выполняться с простыми запросами, а в случае, если сложный запрос будет «зависать», то с каждым следующим  $n$ -ым шагом выполнения запроса время ожидания будет увеличиваться на  $n$  раз.

На рисунке 2 показано, каким образом ВС обеспечивает гетерогенные свойства в РБД. Для выполнения запроса конкатенации с использованием ВС с шлюзами в РБД предложен протокол общения с ВС, в котором описаны такие данные, как: *SQL-запрос*, в котором указано какие данные требуется перенести из таблиц-источников в таблицу-приемник; *информация об источниках*, где описаны подключения к СУБД в узлах-источниках; *информация о приемнике*, где описано подключение к СУБД в узле-приемнике.

Предполагается, что источник и приемник может быть один и тот же узел. В таком случае из узла будут переданы таблицы на ВС для выполнения запроса конкатенации и результат будет возвращен в таблицу на том же узле. Это уместно в случае, если узел не располагает достаточными вычислительными ресурсами для эффективного выполнения запроса конкатенации.



Рисунок 2 – Алгоритм работы ВС с обеспечением гетерогенных свойств узлов в федеративной гетерогенной РБД

#### Метод оптимизации выполнения повторяющихся распределенных SQL-запросов

Учитывая то, что журнализация всей БД может значительно влиять на производительность узла и самой РБД, ранее в работе [3] было описано использование механизмов триггеров для журнализации отдельно взятых таблиц и их доменов, а также метод инъекции функций триггеров и таблиц для журнализации в различные SQL-ориентированные СУБД. Такой подход более эффективен, в том случае, когда требуется частичная журнализация БД.

В случае использования ВС с шлюзом, данные журнализации востребованы лишь на ВС. Для минимизации нагрузки на узлы РБД при журнализации таблиц и их доменов предложен метод, который позволяет вести ее в 2 этапа. На первом этапе хранить таблицы журнализации на узле, а на втором этапе переносить таблицы журнализации в БД на ВС. Период обновления данных журнализации на ВС, можно определить по формуле:

$$P_{\text{обновления}} = k \frac{N_{\text{изменений}}}{N_{\text{чтений}}}$$

где  $k$  – коэффициент поправки для всего ВС,  $N_{\text{изменений}}$  – количество изменений в таблице за месяц,  $N_{\text{чтений}}$  – количество чтений с таблицы за месяц.

После того, как узел обновит первый раз полученные данные запроса, есть смысл очистить данные журнализации используемых таблиц и их доменов. Учитывая, что ВС многопользовательский, то может потребоваться, чтобы одна таблица участвовала одновременно в нескольких повторяющихся запросах конкатенации. В таком случае очистка устаревших данных журнализации зави-

сит от выполнения самого позднего запроса.

На рисунке 3 отображена схема выполнения повторяющихся запросов в федеративной гетерогенной неоднородной SQL-системе с ВС с шлюзами и журнализацией. В примере требуется выполнять два повторяющихся запроса: конкатенация данных из таблиц  $DT1$ ,  $DT2$ , сохранение результата в таблице  $DT4$  через интервал времени  $t4$ ; конкатенация данных из таблиц  $DT2$  и  $DT3$ , сохранение результата в таблице  $DT5$  через интервал времени  $t5$ . Для этого выполняется запрос первый раз и на ВУ формируются временные таблицы  $CT12$  и  $CT23$ . Затем на ВС генерируются триггеры и таблицы первого этапа журнализации:  $JT1$ ,  $JT2$ ,  $JT3$  и происходит инъекция в СУБД с БД  $DT1$ ,  $DT2$ ,  $DT3$ , и генерируются таблицы второго этапа журнализации  $JTU1$ ,  $JTU2$ ,  $JTU3$ , которые хранятся в БД на ВС. Перенос данных из таблиц журнализации первого этапа  $JT1$ ,  $JT2$ ,  $JT3$ , в таблицы журнализации второго этапа  $JTU1$ ,  $JTU2$ ,  $JTU3$  происходит через соответствующие интервалы времени  $t1$ ,  $t2$ ,  $t3$ . Через интервалы времени  $t4$  и  $t5$  на ВС выполняется генерирование запросов  $JCT12$  и  $JCT23$ , а затем обновление данных в таблицах  $DT4$  и  $DT5$  на узлах РБД. В случае, если требуется удалить повторяющийся запрос на ВС, тогда проверяется какие триггеры, таблицы журнализации на узлах и ВС не используются и могут быть удалены.

На рисунке 4 показано как будет вести себя ВС с оптимизацией выполнения повторяющегося распределенного запроса в федеративной гетерогенной неоднородной РБД. Есть два варианта выполнения запроса – первый раз и последующие с использованием таблиц журнализации таблиц-источников и их доменов.

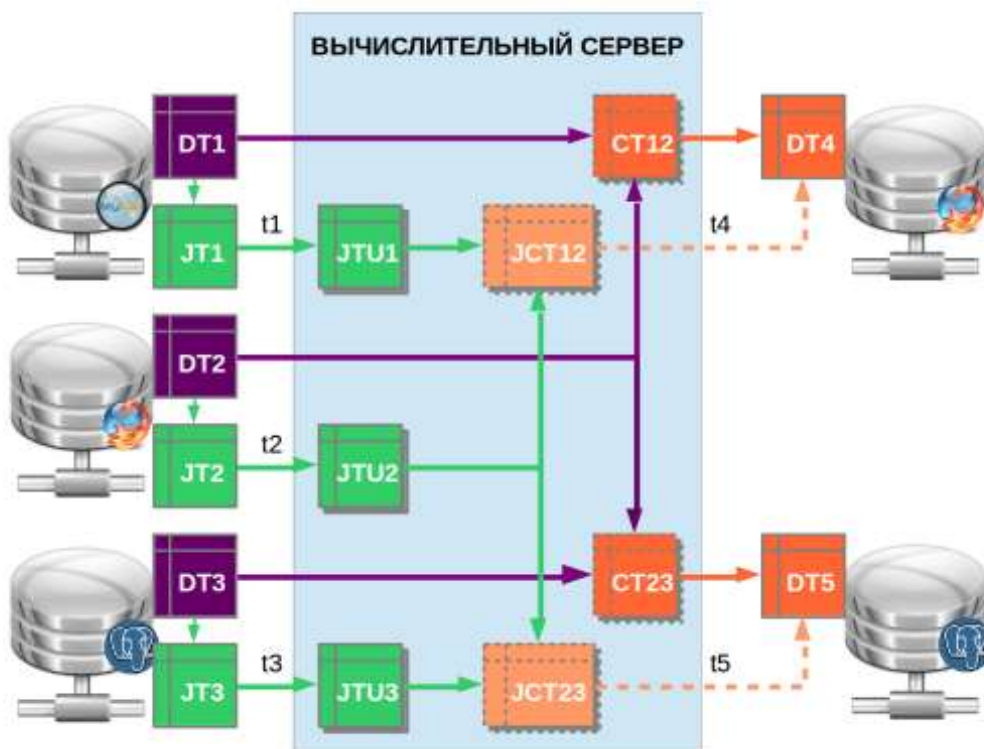


Рисунок 3 – Схема выполнения повторяющихся запросов в федеративной гетерогенной неоднородной SQL-системе с ВС с шлюзами и журнализацией



Рисунок 4 – Алгоритм работы ВС с оптимизацией выполнения повторяющегося распределенного запроса в федеративной гетерогенной неоднородной РБД

### III. ВЫВОДЫ

Появление федеративных РБД обусловлено не прогнозируемым поэтапным нарастанием количества информации, способов использования накопленной информации и непрерывным развитием технологий. Очевидно, что в федеративной РБД возникает дополнительная нагрузка на узлы и коммуникационную сеть, ранее не взаимодействующих между собой РБД. В федеративной РБД существуют дополнительные требования к политике безопасности на узлах, целью которых является ограничение доступа к данным и экономия вычислительных ресурсов. Таким образом возникает потребность в создании новых ролей и правил доступа на узлах: для локальных пользователей; для пользователей РБД; для пользователей федеративной РБД.

Использование предложенных методов улучшения характеристик гибкости и производительности федеративной гетерогенной неоднородной SQL-системы, а именно: использование на ВС шлюзов SQL-ориентированных СУБД на узлах; использование на ВС оптимизации выполнения

повторяющихся запросов, позволит пользователям такой системы развивать все более сложные проекты, использовать в них подходящие СУБД и не заботиться о совместимости с системой, использовать в проектах повторяющиеся запросы, которые выполняются достаточно эффективно как для самого проекта так и для всей системы в целом.

### ЛИТЕРАТУРА

1. **Amit P. Sheth**, Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases / Piscataway, New Jersey – 1990 – 54 pages
2. **К. Дж. Дейт**, Введение в системы баз данных / Промежуточное программное обеспечение для доступа к данным / Вильямс, Москва – 2005 – стр. 855-858
3. **А.А. Блажко, Т. Джайяб**, Тиражирование операций согласования в гетерогенных распределенных базах данных / ИСТА '08. – 2008.

---

Получена в редакции 20.03.2013, принята к печати 22.03.2013