

Р.А. СИМАКОВ,
В.С. ЯШКОВ

**Повышение отказоустойчивости и
актуальности баз данных с
помощью репликации**

УДК 004.658.6

Муромский институт
(филиал) ФГБОУ ВПО
«Владимирский
государственный
университет имени
А.Г. и Н.Г. Столетовых»,
г. Муром

Каждый год сферы деятельности человека переходят к электронному варианту хранения и обработки данных[3]. Для таких целей используются системы управления баз данными (СУБД). Популярными СУБД являются «Firebird»[4], «Ред База Данных», «Oracle», «MSSQL»[3], «MySQL» и множество других. Быстрая скорость работы современных СУБД позволяет обрабатывать большое количество информации за короткий промежуток времени. Базы данных заметно упростили и сделали удобным процесс контроля данных, поиск информации, распространение различных сервисов в сети интернет.

Большинство современных услуг можно произвести на сайтах компаний и государственных учреждений.

Все эти возможности реализуются с помощью непрерывно работающих систем. Важным критерием является то, что в такие системы должны обладать отказоустойчивостью в любых ситуациях. Именно базы данных являются уязвимым местом в таких системах. Чтобы система обладала отказоустойчивостью, необходимо повысить отказоустойчивость баз данных.

Один из способов повышения отказоустойчивости баз данных – использование репликации[1,4,5].

Репликация данных (от лат. *replico* - повторяю) – это процесс, под которым понимается копирование данных из одного источника на другой (или на множество других) без прерывания работы системы.

Репликация в базах данных — это тиражирование данных с главной базы данных на одну или несколько зависимых баз данных. Главная база данных называется мастером, а зависимые - подписчиками.

Репликация разделяется на синхронную и асинхронную:

- Синхронной называется репликация, которая производится при обновлении данных главной базы и всех зависимых от нее баз в одной и той же транзакции. В логическом понимании это означает, что данные не разделяются, а существует одна их копия. В большинстве случаев синхронная репликация реализуется с помощью создания триггерных процедур на изменение данных. Недостаток синхронной репликации заключается в том, что создается дополнительная нагрузка на все выполняемые транзакции, в которых обновляются какие-либо зависимые базы (также могут возникать проблемы, связанные с ограничением доступа к данным).

- Асинхронная репликация – это репликация, производящаяся спустя некоторое время, а не в той же транзакции. В результате при асинхронной репликации появляется задержка (время ожидания), которая может внести изменения в подписчиков. В общих случаях асинхронная репликация реализуется с помощью ведения журнала изменений, который используется для последующих обновлений подписчиков. Преимущество асинхронной репликации заключается в том, что дополнительные затраты репликации не связаны с зависимыми транзакциями обновлений, имеющее важное значение для функционирования, и предъявляющее высокие требования к производительности. Недостатком асинхронной репликации является возможное отличие данных на зависимых от мастер-базы подписчиках.

Изменения данных, происходящие на мастер-базе, повторяются на подписчиках. Процесс зеркалирования на одного из подписчика не влияет на работу других подписчиков, и практически не влияет на работу мастер-базы.

Репликация может выполняться в ручном режиме, либо автоматически по расписанию или при изменении журнала, который обновляется при внесении изменений в мастер-базу. В журнал сохраняются все действия с мастер-базой, вызывающие изменения данных в ней. Изменения сохраняются в виде метаданных, а не в явном виде, иначе возникает проблема избыточности данных

Репликация также подразделяется на полную и инкрементальную.

Полная репликация подразумевает полную синхронизацию подписчика и мастер-базы. Полная репликация не зависит от поколения мастер-базы.

Инкрементальная репликация позволяет производить не полную синхронизацию данных мастер-базы и подписчика, а только изменения, относительно последнего поколения, записанного в мастер-базу.

Поколение – это уникальный номер индексируемых изменений в мастер-базе. После изменений в мастер-базе, в журнал записывается информация об изменениях, которым присваивается номер.

Процесс репликации представлен на рисунке 1.

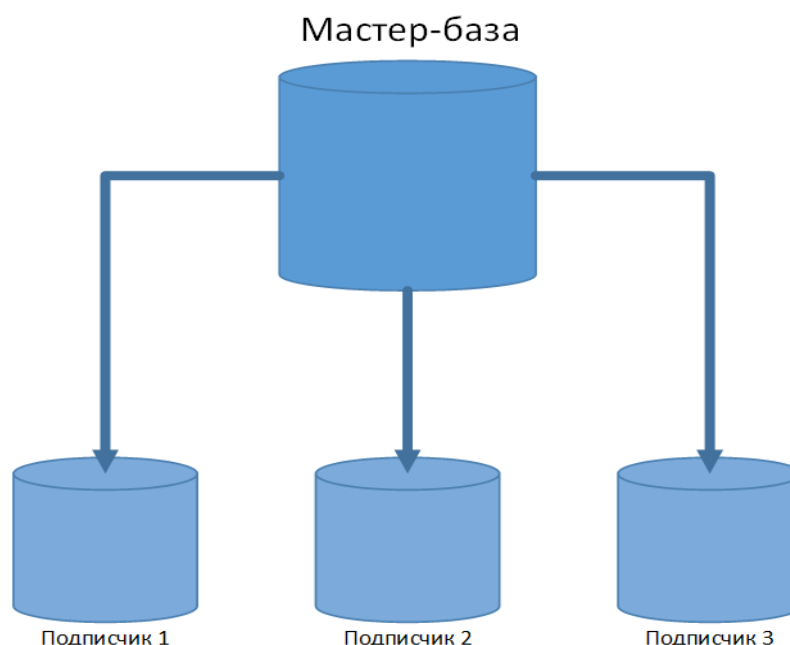


Рис. 1. Процесс репликации

Использование репликации решает несколько задач по повышению производительности и надежности системы.

- Производительность и масштабируемость. Если одна база данных подвержена высокой нагрузке, которая создается одновременными операциями записи и чтения. Запросы на чтение данных можно перенаправить на подписчиков. При этом такая система будет выгодней, если несколько операций чтения приходится на одну операцию записи.

- Отказоустойчивость. При отказе одного из подписчика, запросы можно распределить по другим подписчикам или перевести все запросы на мастер-базу, если был лишь один подписчик. В случае отказа мастер-базы, запросы можно перевести на одного из подписчика, а после того как мастер-база будет восстановлена, ее можно сделать подписчиком.

- Резервирование данных. При использовании репликации можно создать резервную базу данных, которая будет в точности повторять мастер-базу.

- Сложные вычисления. Медленные и сложные запросы[2] можно производить на резервной копии мастер-базы или на подписчике, без замедления работы всей системы.

- Модернизация или замена оборудования. Если необходимо остановить один из серверов с базой данных, то все запросы можно перевести на резервную базу данных или подписчика, без нарушения работы системы.

- Возможность переопределения структур таблиц на подписчиках. В частности, можно изменить тип полей в таблицах подписчика или добавить различные индексы.

Репликация также несет в себе ряд недостатков.

- Различие копий. Асинхронная репликация может вносить изменения в данных между мастер-базой и подписчиками. Так как при изменении данных в мастер-базе производится обновление журнала, который используется для обновления подписчиков. Разница во времени может существенно повлиять на работу системы.

- Проблемы со случайными значениями. Если в запросах использовалась функция RAND()[4], то при использовании этого запроса на подписчиках результат может быть отличным от результата в мастер-базе.

При создании системы использующей метод репликации баз данных, не критичен отказ главной мастер-базы, работоспособность системы восстанавливается за короткий промежуток времени вручную, или с помощью контроллера, автоматически восстанавливающего работоспособность системы, сделав одного из подписчика мастер-базой и изменив настройки остальных

подписчиков таким образом, чтобы они брали изменения с новой мастер-базы.

Использование репликации можно улучшить созданием дополнительной утилиты, которая будет обрабатывать все запросы к мастер-базе и следить за нагрузками по доступности подписчиков[2]. В результате, в системе нагрузка на мастер-базу будет минимальна, а запросы на выборку данных будут распределяться только по тем подписчикам, содержащие актуальные данные для запроса.

Литература

1. Р.А. Симаков, М.А. Смяткин. Особенности архитектуры распределенной массивно-реляционной СУБД// Алгоритмы, методы и системы обработки данных. 2011. № 18. С. 9.
2. Е.В. Пугин, Р.А. Симаков. Аналитический обзор способов построения процессоров запросов в СУБД// Алгоритмы, методы и системы обработки данных. 2012. № 21. С. 76-89.
3. Р.А. Симаков, М.И. Ткачук. Организация информационной системы кафедры для методического обеспечения на основе технологии MS .NET// Алгоритмы, методы и системы обработки данных. 2006. № 11. С. 27-31.
4. Борри Х. The Firebird Book. A Reference for Database Developers / Firebird. Руководство разработчика баз данных. Наиболее полное руководство – БХВ-Петербург, 2006, 1104 с.
5. Каролин Бегг, Томас Коннолли. Базы данных. Проектирование, реализация и сопровождение. Теория и практика – Вильямс, 2003, 1436 с.
6. Д.Е. Андрианов, С.В. Еремеев, С.С. Садыков. Теоретические основы описания и анализа плоских пространственно-распределенных объектов в ГИС - Владимир: Изд-во Владимир. гос. ун-та, 2007. -109 с.