

К.Ю. Островська, Є.В. Островський, С.В. Волков
**РОЗРОБКА АЛГОРИТМУ АСИНХРОННОЇ РЕПЛІКАЦІЇ
ДЛЯ ПОБУДОВИ РБД З ЗАДАНОЮ ТОПОЛОГІЄЮ**

Анотація. Розглянуто моделі синхронної та асинхронної реплікації розподілених баз даних з заданою топологією. Розроблено алгоритм асинхронної реплікації для побудови РБД з заданою топологією.

Ключові слова: алгоритм, реплікація, тиражування, топологія, база даних.

Сучасні інформаційні системи пред'являють досить високі вимоги до швидкості обробки інформації за умови одночасної роботи великої кількості клієнтів. Крім того, розвиваючись, такі системи повинні легко масштабуватись без шкоди для швидкісних характеристик системи.

Один із способів задоволення цієї потреби - створення розподіленої бази даних БД, що підтримує механізм асинхронної реплікації даних. У цьому випадку замість однієї БД, з якою повинні працювати всі клієнти інформаційної системи, створюється кілька однакових серверів БД на різних машинах і / або вузлах мережі. Клієнти мають доступ до деякого розподіляє пристрою (реалізованому апаратно або програмним методом), яке при підключенні нового клієнта оцінює завантаження кожного сервера БД і направляє клієнта до найменш завантаженому сервера, з яким він (клієнт) і буде працювати до від'єднання.

Питання побудови розподіленої бази даних єдиної інформаційної системи виникають і при розвитку компанії, коли створюються віддалені філії, магазини і склади. Кожна віддалена інформаційна система з метою підвищення стійкості повинна працювати самостійно, періодично відправляючи в Центральний офіс консолідовану інформацію. Для виключення людського фактора в питанні періодичної синхронізації інформації бази даних повинні бути включені в загальну систему реплікації.

Реплікація даних між серверами баз даних може виконуватися за допомогою вбудованих засобів СУБД або може бути реалізована в рамках бізнес-логіки додатків. Реплікація за допомогою вбудованих засобів СУБД передбачає наявність надійних каналів зв'язку. Пропускна здатність цих каналів повинна бути досить високою, щоб встигати передавати всю інформацію, що реплікується в realtime-режимі. Процес реплікації в СУБД заснований на поняттях "видавець", "передплатник", "стаття". Налаштування реплікації зводиться до установки відносин між видавцем і передплатником. Недоліком даної реплікації є односпрямованість. Тобто стаття (фактично це таблиця) може передаватися від видавця передплатнику. При цьому передплатник не може її змінювати.

Реалізації процесів реплікації на рівні бізнес-логіки значно ускладнює життя розробнику, але дозволяє значно оптимізувати сам процес передачі інформації. Проблема реплікації інформації являє собою досить нетривіальну задачу з вельми неоднозначним рішенням. Приступаючи до вирішення завдання реплікації даних, необхідно брати до уваги, що доведеться зіткнутися з конфліктами репліцируемой даних, яких для баз даних, що працюють в єдиній мережі прямого конекту до сервера бази даних, не виникає в принципі. Особливо складний перехід від єдиної бази до розподіленої, коли доводиться підлаштовувати алгоритм реплікації під вже існуючу структуру працюючої БД. При розробці нової інформаційної системи необхідно враховувати технологічні нюанси майбутньої розподіленої бази даних.

Реплікація (англ. Replication) - механізм синхронізації вмісту декількох копій об'єкта (наприклад, вмісту бази даних).

Для підтримки цілісності розподіленої БД в СУБД Лінтер використовується механізм асинхронного тиражування (далі по тексту - реплікації) транзакцій.

Суть механізму асинхронного тиражування полягає в тому, що обробка даних виконується локально, а розподілені дані копіюються на той сервер, де вони повинні використовуватися. При такому методі підтримки логічної цілісності розподіленої БД має місце деяка рясинхронізація стану локальних БД в часі, тобто зміна стану однієї локальної бази даних відстає від зміни іншої локальної бази даних у часі.

Якщо один з серверів системи, що вимагають оновлення тиражованих даних, виходить з ладу, то система продовжує працювати з іншими, при цьому оновлення даних на сервері після його ремонту відбудеться автоматично, тобто помилка на одному вузлі глобальної мережі не вплине на роботу інших вузлів.

Механізм асинхронного тиражування транзакцій гарантує доставку змінених даних на вторинні сервери безпосередньо після завершення транзакції, якщо сервер доступний, або відразу після підключення сервера до мережі. Такий підхід передбачає зберігання дублюючої інформації в різних вузлах мережі і може забезпечити, в порівнянні з іншими підходами до реплікації, зниження трафіку, поліпшення часу відповіді системи, а також дозволяє оптимізувати навантаження на сервери.

Асинхронна реплікація, на відміну від двофазної синхронізації, не забезпечує повною синхронності інформації на всіх серверах в будь-який момент часу. Синхронізація відбувається через деякий, зазвичай невеликий, інтервал часу, величина якого визначається швидкістю відповідного каналу зв'язку. Для більшості завдань короткочасне наявність застарілих даних у віддалених вузлах цілком припустимо.

При синхронній реплікації, дані копіюються в межах однієї транзакції. Або іншими словами можна сказати, копіювання відбувається транзакція до транзакції. Зазвичай, цю модель використовує протокол two-phase commit. Цей протокол гарантує, що транзакція пройде послідовно по всіх базах даних. Оскільки зміни відбуваються в межах однієї транзакції, синхронна реплікація гарантує високу надійність і цілісність даних. Така транзакція або буде прийнята у всіх базах, або буде відкинута всюди.

Асинхронна реплікації копіює дані поза області транзакції. Копіювання може відбуватися протягом секунд, хвилин, годин і навіть днів, в залежності від вимог бізнесу. Хоча реплікація виконується запис в запис, вона може відбуватися і за транзакціями. Тобто, якщо відбуваються множинні зміни в межах однієї транзакції, вони можуть скопіювати як одна транзакція.

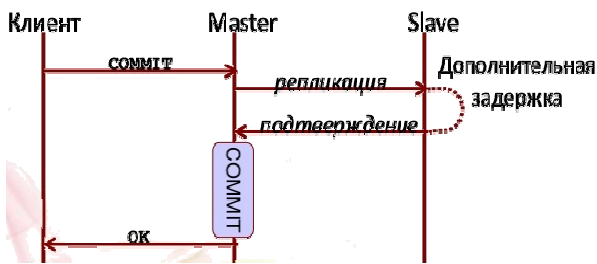


Рисунок 1 - Модель синхронної реплікації



Рисунок 2 - Модель асинхронної реплікації

Моделі зв'язків бази даних. Моделі зв'язків баз даних визначають, яким чином зміни в одній базі впливають на зміни інших баз в мережі. Нижче описані три такі моделі: розподілу даних, консолідації даних і однорангова модель, а також як вони пов'язані з реплікацією.

Модель розподілу даних. У розподіленій моделі, тільки одна база даних є основною. Всі зміни відбуваються саме в цій базі. Дані копіюються з неї в інші бази в мережі і маю статус «тільки для читання», тобто заборонено вносити зміни в них. З точки зору реплікації, головна перевага цієї моделі то, що це суттєво зменшує можливість конфліктів при зміні записів, тому що зміни відбуваються тільки в одній базі даних.

Виконання централізованої реплікації.

При централізованій реплікації, центральна база даних копіюється на безліч цільових баз даних. Централізована реплікація дозволяє наступне:

- Створення резервних баз даних в реальному часі.
- Гнучка можливості контролю часу реплікації
- Прозорий спосіб віддаленого резервного копіювання первинної бази даних.
- Реплікація здійснюється засобами after-image файлів і резервної копії центральної бази.

Розглянемо процес передачі змін:

1. При зміні даних в таблиці, що реплікується нові дані через тригер записуються в журнал змін. Крім того, туди заноситься ім'я таблиці, код зробленого зміни і первинний ключ зміненої записи.

2. При виникненні в базі певного події (наприклад при великій кількості записів в журналі змін) або в певний момент часу комунікаційний сервіс запускає процес реплікації.

3. Процес реплікації встановлює з'єднання з сервером підписки і починає синхронізацію даних.

4. Сервер підписки приймає змінену запис і модифікує відповідним чином таблицю на своєму боці.

5. Якщо в процесі зміни записи був згенерований новий ключ, то він передається на сервер реплікації.

6. Сервер реплікації замінює первинний ключ репліцируемой записи на ключ, що повертається з сервера підписки і видаляє відповідний запис з журналу змін.

На рисунку 3 представлено схему реплікації даних.

При передачі змін комунікаційним сервісом використовується протокол двофазної фіксації транзакцій (Two-phase commit transactions), що дозволяє застрахуватися від помилок.

При синхронізації даних подібним методом процес реплікації може бути перерваний у будь-який момент часу і продовжений пізніше з тієї ж точки. Дана особливість дозволяє використовувати таку схему тиражування навіть на дуже поганих каналах зв'язку.



Рисунок 3 - Детальна схема моделі реплікації даних

ЛІТЕРАТУРА

1. К. Дейт. Введение в системы баз данных, 6-е изд. Диалектика, 1998
2. А. Фаритов, А. Луковенко Практическая репликация, «Открытые системы», № 12, 2001