

Підтримка механізму транзакцій програмною платформою Spring

Г.В. Олійник, С.В. Грибков

Національний університет харчових технологій

Для більшості сучасних корпоративних інформаційних систем характерними особливостями є реалізація складних бізнес-процесів з визначеною логікою послідовності виконання операцій, а також наявність великої кількості користувачів, які здійснюють паралельний доступ до функцій системи. При створенні такої системи існує проблема забезпечення її консистентного стану у будь-яких проміжках часу та підтримка одночасної роботи усіх її користувачів на належному рівні, що особливо стає актуальним при створенні web - орієнтованих інформаційних систем.

Для операцій користувачів, виконання яких вимагає роботу декількох незалежних частин, прийнято використовувати транзакції. Транзакцією називається одиниця виконання роботи у програмному додатку, яка представляє собою єдине ціле, і або виконується повністю, або не виконується взагалі, а також має наступні властивості: атомарність (atomicity) – усі складові транзакції розглядаються нероздільно; консистентність (consistency) – кожна транзакція переводить систему з одного несуперечливого стану в інший; ізолюваність (isolation) – зміни, які вносяться в результаті виконання транзакції, не мають бути видимі до моменту її успішного завершення; надійність (durability) – внесені зміни повинні надійно зберігатися системою. Однією з важливих задач для розробника системи є застосування механізму роботи з транзакціями у відповідності з наявними бізнес-вимогами для забезпечення загальної коректності функціонування.

Для управління транзакціями у web - орієнтованих інформаційних системах авторами обрана та рекомендується програмна платформа Spring, що надає простий, гнучкий та функціональний набір компонентів для програмного і декларативного управління глобальними та локальними транзакціями. Крім цього, модель транзакцій Spring легко інтегрується з програмними платформами для роботи з різними базами та сховищами даних.

Для використання механізму транзакцій програмна платформа Spring надає менеджер транзакцій, який дозволяє визначати стан поточної транзакції, здійснити її виконання або відміну, а також дозволяє проводити гнучке налаштування параметрів транзакції: необхідність створення нової транзакції в залежності від моменту роботи системи, рівень ізоляції транзакції, максимальний час, за який транзакція повинна бути виконана тощо.

Перевагою програмного управління транзакціями є те, що межі транзакції можуть обиратися розробником довільним чином і не співпадати з формальними межами окремого методу, що дозволяє реалізовувати складні критерії відміни усіх змін, внесених транзакцією, а не тільки за умови виникнення виключення на будь-якому етапі її виконання, та повернення до початкового стану. Недоліками даного підходу є: перевантаження програмного

коду, а, як наслідок, ускладнення його сприйняття і подальшої підтримки; зв'язування бізнес-функціональності і службового програмного коду для управління транзакціями. Зазначені недоліки є можливістю частково усунути за рахунок використання додаткової функціональності програмної платформи Spring, а саме модуля управління шаблонами транзакцій. У рамках даного підходу надається окремий службовий компонент, що забезпечує виконання послідовності дій у рамках транзакції. Проте, програмний спосіб управління транзакціями неефективний при наявності великих об'ємів програмного коду та типової бізнес-логіки.

Суть декларативного управління транзакціями полягає у тому, що одиницею управління транзакцією є метод класу. Декларативне управління використовується також у стандартному Java Enterprise Edition (набір специфікацій мови Java, що описує архітектуру серверної платформи) під назвою CMT (Container-Managed Transactions).

Перевагою моделі транзакцій Spring над моделлю CMT є те, що вона не прив'язана до конкретної технології, а інкапсулює дані про неї в реалізації менеджера транзакцій для конкретної платформи. Також, у Spring використання транзакцій доступне для будь-яких класів при наявності відповідної конфігурації, а в CMT тільки для особливих класів, таких як EJB (Enterprise Java Beans – серверні компоненти, що містять бізнес-логіку). Основою декларативного управління в Spring є підтримка аспектно-орієнтованого програмування. Коли викликається метод компонента, для якого налаштовано декларативні транзакції, спеціалізовані служби перехоплюють виклик та передають його сервісам управління транзакціями. Після виконання методу управління передається назад клієнту.

Spring забезпечує роботу з локальними та глобальними транзакціями, без прив'язки до технологій їх реалізації. Spring надає досить широку підтримку реалізації декларативних транзакцій, як за допомогою XML-конфігурації, так і за допомогою анотацій. Окрім цього є можливість управляти транзакціями програмно, що зумовлює гнучкість при налаштуванні в залежності від конкретних вимог до функціональності системи.

Література

1. Rod Johnson Spring Framework Reference Documentation [Електронний ресурс] – режим доступу: <http://docs.spring.io/spring-framework/docs/4.2.x/spring-framework-reference/html/transaction.html>
2. *Schaefer C., Ho C., Harrop R. Pro Spring 4th ed. Edition / Chris Schaefer, Clarence Ho, Rob Harrop - Apress, 2014 - 728 p.*
3. *Mularien, P. Spring Security / P. Mularien – PACT Publications, 2010 – 396 с.*
4. *Уоллс К. Spring в действии. / К. Уоллс – М.: ДМК Пресс, 2013. – 752 с.*