

УДК 004.6
М79

Рецензент
кандидат технических наук, доцент *А.С. Кожаринов*

Морозов Е.А.

М79 Проектирование транзакций для приложений информационных систем: Учеб. пособие. – М.: МИСиС, 2004. – 62 с.

В учебном пособии систематизируются основные понятия теории транзакций, их свойства, модели и структура. Основное внимание уделено задачам, связанным с проектированием транзакций при разработке приложений информационных систем. Краткая теория по каждой задаче сопровождается примером и рекомендациями по практическому проектированию.

Содержание пособия соответствует программе курса «Проектирование информационных систем».

Пособие может использоваться как при изучении курса «Проектирование информационных систем», так и при освоении дисциплины «Базы данных».

Предназначено для студентов, обучающихся по специальностям 3514э «Прикладная информатика» и 200200 «Автоматизированные системы обработки информации и управления».

ОГЛАВЛЕНИЕ

Введение	4
1. Понятие транзакции.....	5
2. Свойства транзакций	6
3. Модели транзакций.....	7
4. Типовая структура транзакций	9
5. Возможные случаи завершения транзакций.....	11
6. Блокирование информационных объектов базы данных	12
7. Явное и неявное блокирование.....	16
8. Параллельное исполнение транзакций	20
9. Проблемы параллелизма	23
10. Возникновение тупиковых ситуаций и их устранение.....	29
11. Задание уровней изоляции транзакций.....	31
12. Управление параллелизмом с помощью временных меток	33
13. Оптимистические технологии управления параллелизмом.....	35
14. Усиление параллелизма посредством вложенных подтранзакций ..	39
15. Регистрация и завершение транзакций	43
16. Задачи проектирования транзакций	45
17. Определение состава прикладных программ, реализующих транзакции	46
18. Выделение и анализ конфликтующих транзакций	51
19. Уменьшение времени исполнения длинных транзакций	54
Заключение	58
Контрольные вопросы и задания.....	59
Библиографический список	61

ВВЕДЕНИЕ

Современным крупным информационным системам (ИС) присущ, как правило, многопользовательский режим работы, при котором с базой данных (БД), используемой информационной системой, одновременно может работать большое число пользователей. При этом в общем случае предполагается, что эти пользователи могут не только выбирать данные из БД, но и модифицировать их. Последнее допущение (естественное для большинства современных систем) неизбежно приводит к необходимости разработки транзакций для приложений этих систем.

Как известно, механизмы транзакций поддерживаются системами управления базами данных (СУБД). Это, пожалуй, наиболее сложный механизм любой СУБД и с точки зрения его реализации самой СУБД, и с точки зрения его использования.

Поэтому, несмотря на то, что многие люди, хоть в какой-то мере причастные к информационным системам или к базам данных, знают или, по крайней мере, слышали термин «транзакция», далеко не многие используют его правильно, а тем более в состоянии успешно реализовать транзакции в своих программах.

Следствием этого являются серьезные огрехи при разработке информационных систем, ведущие к снижению их эффективности и надежности. При этом парадокс (или комизм) ситуации состоит в том, что правильно разработанные приложения подчас отличаются от неправильных всего-навсего несколькими операторами («пустячком»). Причина этого состоит в том, что, как будет показано далее, главное при разработке транзакций – понять их роль и место в прикладной программе, а определить их формально несложно (с помощью немногочисленных операторов).

Учебное пособие посвящено проектированию транзакций для приложений информационных систем с централизованной базой данных. Проектирование транзакций рассматривается как один из этапов проектирования информационных систем, а именно как этап, следующий за этапом логического проектирования БД и предшествующий этапу непосредственного программирования приложений.

Для простоты изложения предполагается, что приложения разрабатываются с использованием встроенного SQL (хотя все это также относится и к приложениям с прикладным API).

1. ПОНЯТИЕ ТРАНЗАКЦИИ

Изучение транзакций начнем с рассмотрения двух простых примеров:

- 1) бронирование билета на авиарейс;
- 2) перевод денег с одного банковского счета на другой.

Чем характерны эти примеры? Они представляют собой некие вычислительные работы. Причем и в том, и в другом случае эти работы связаны с выполнением двух операций.

Бронирование билета предполагает запись о занятии определенного места и уменьшение числа свободных мест на единицу. Перевод денег означает снятие некоторой суммы с одного счета и зачисление этой суммы на другой счет. Иными словами, выполнение каждой из этих работ связано с выполнением в базе данных двух обновлений (под обновлениями здесь понимаются операции INSERT, DELETE и непосредственно UPDATE).

Чем еще характерны эти работы? Тем, что каждая из них должна быть выполнена полностью (от начала и до конца). Нельзя выполнить только часть работы, например, сделать запись о занятии места, но не уменьшить число свободных мест. Это противоречит логике выполняемой работы и нарушает базу данных (делает ее содержимое противоречивым). Более того, в БД между этими двумя обновлениями временно нарушается согласованность данных (они становятся противоречивыми).

Таким образом, транзакция – не просто одиночная операция системы баз данных, а, скорее, согласование нескольких таких операций. В общем, это преобразование одного согласованного состояния базы данных в другое, причем в промежуточных точках БД находится в несогласованном состоянии.

Транзакция – логически неделимая вычислительная работа над базой данных, переводящая ее из одного непротиворечивого состояния в другое непротиворечивое состояние [1].

Это понятие транзакции возникло в 1970–1972 гг. и достаточно долго использовалось (хотя справедливости ради надо отметить, что были и другие определения, в которых также достаточно часто транзакция рассматривалась как единица восстановления). Тем не менее это определение транзакции так окончательно и не утвердилось. Фактически с 1972 по 1990-е годы понятие транзакции во многих научных работах часто искажалось. Существовала страшная путаница не только в советской литературе, но и у американцев, которые называли

транзакции как угодно. В нашей стране из-за неясности термина часто использовался перевод «запись о событии» и подчас было очень трудно понять, о какой записи и о каком событии идет речь.

Конец этой неразберихи был положен предложением американского института по стандартизации ANSI/ISO, который дал формальное определение транзакции. В настоящее время термин *транзакция* устоялся.

По ANSI/ISO транзакция – это некоторая последовательность операторов SQL, заканчивающаяся оператором COMMIT – завершить транзакцию (этот оператор также является оператором SQL).

2. СВОЙСТВА ТРАНЗАКЦИЙ

Определение транзакции, по ANSI/ISO, кратко, но совершенно неинформативно. Поэтому его принято пояснять следующими свойствами транзакций (ACID):

1) **атомарность (неделимость)**. Означает, что транзакция является логически неделимой вычислительной работой, исполняемой по принципу: *все или ничего*;

2) **непротиворечивость**. Означает, что в результате действия транзакции БД должна переводиться из одного непротиворечивого состояния в другое непротиворечивое состояние;

3) **изолированность**. Транзакции отделены друг от друга. Это означает, что, если даже будет запущено множество конкурирующих между собой транзакций, любое обновление определенной транзакции будет скрыто от остальных до тех пор, пока эта транзакция выполняется. Другими словами, для любых двух отдаленных транзакций T_1 и T_2 справедливо следующее утверждение: T_1 сможет увидеть обновление T_2 только после выполнения T_2 , а T_2 сможет увидеть обновление T_1 только после выполнения T_1 .

Следствием этого свойства является то, что при параллельном исполнении нескольких транзакций результат их воздействия на БД должен быть такой, как будто эти транзакции выполнялись последовательно одна за другой;

4) **долговечность**. Когда транзакция выполнена, ее обновления сохраняются, даже если в следующий момент произойдет сбой системы.

В случае возникновения сбоя (отказа в системе) БД должна быть восстановлена в одном из согласованных состояний, которые она имела на момент начала аварийно завершившейся транзакции, причем это восстановление может быть выполнено по истечении сколь угодно большого интервала времени после отказа.

3. МОДЕЛИ ТРАНЗАКЦИЙ

Определение транзакции по ANSI/ISO молчаливо допускает наличие в ней любых операторов любого алгоритмического языка. Кроме того, в определении не сказано о том, что считать началом транзакции, поэтому согласно ANSI/ISO началом транзакции считается любой оператор SQL, в том числе и SELECT (оператор выборки).

Эта идеология ANSI/ISO получила название *модели транзакции ANSI/ISO*.

Существует еще *расширенная модель*, иногда называемая в литературе *моделью SYBASE*. Хотя, если быть точным, оба названия не отражают истины, так как эта модель появилась не в SYBASE и при этом задолго до модели ANSI/ISO. Фактически эта вторая модель была первой. Именно она появилась в 1970–1972 гг.

Согласно определению по второй модели транзакция – это последовательность (совокупность) операторов SQL, начинающихся с оператора BEGIN TRANSACTION, заканчивающаяся оператором COMMIT и допускающая использование специальных операторов SAVE TRANSACTION (рис. 3.1).

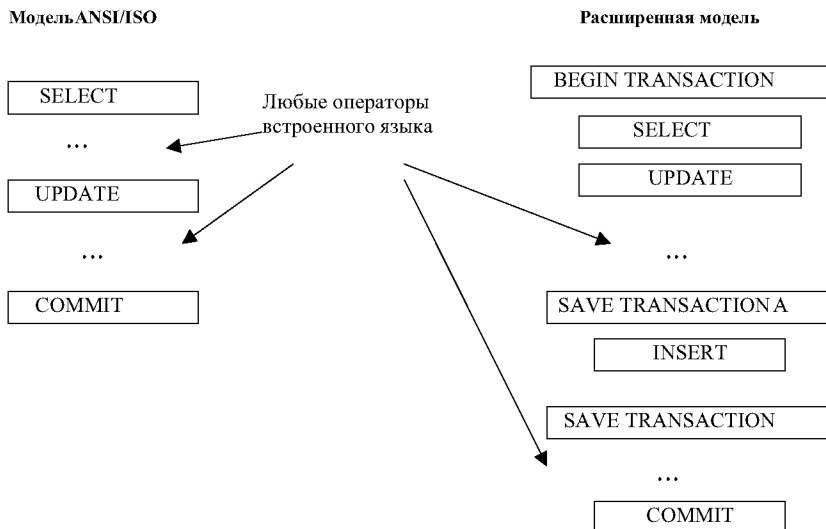


Рис. 3.1. Модели транзакций