

3. Количественные методы финансового анализа / Под ред. Стивена Дж. Брауна и Марка П. Кричмена: Пер. с англ. – М.: ИНФРА-М, 1996. – 352 с.
4. Наталуха И. Г. Моделирование спекулятивного бума на финансовом рынке с учетом психологии инвесторов // Материалы VI Всеросс. симпозиума «Математическое моделирование и компьютерные технологии». – Кисловодск, 2004. – Т. 2. – С. 7–8.
5. Христановский В.В., Щербина В. П. Функция полезности: теория и анализ: Учебное пособие.– Х.: ИД «ИНЖЕК», 2006. – 120 с.
6. Черкасова В. А., Батенкова А. А. Влияние стратегических рисков на финансовые результаты компании // Корпоративные финансы. – 2007. – № 3. – С. 64–76.
7. Четыркин Е. М. Финансовая математика. – М.: Дело, 2002. – 540 с.
8. Шарп У., Александер Г., Бейли Д. Инвестиционный менеджмент. – М.: ИНФРА-М, 2003. – 257 с.
9. Шапки А. С. Экономические и финансовые риски. Оценка, управление, портфель инвестиций: Монография. – М.: Издательско-торговая корпорация «Дашков и К^о», 2003. – 544 с.
10. Эддоус М., Стэнфилд Р. Методы принятия решений. – М.: Мир, 2003. – 123 с.
11. Якушков В. К. Функция полезности для инвестирования. – М.: Дело, 2002. – 167 с.

Надійшла 25.04.2008

Розглянуто основні складові оптимального портфеля (очікуваний прибуток портфеля і стандартне відхилення як міра ризику), які дозволяють агенту фінансового ринку неперервно реструктурувати портфель (роблячи максимальною користь проміжного використання і (або) остаточного капіталу) у відповідності з інвестиційними можливостями, які змінюються стохастично. Проаналізовані методи оптимізації (метод Марковіца і Шарпа), а також розглянута роль функції корисності для формування інвестиційного портфеля.

The basic components of optimal portfolio were viewed: expected portfolio's yield and standard deviation as the risk's measure. These components allow the agent of the financial market to restructure portfolio uninterruptedly (maximizing profit of intermediate consumption and (or) final capital) according to stochastic varying investment opportunities. Methods of optimization (H. Markowitz's and W. Sharpe's method) were analysed, and also the role of utility function in forming investment portfolio was viewed.

УДК 004.658.3

А. Б. Кунгурцев, С. Л. Зиноватная

ИМИТАЦИОННАЯ МОДЕЛЬ ДЛЯ ИССЛЕДОВАНИЯ ЭФФЕКТИВНОСТИ ДЕНОРМАЛИЗАЦИИ РЕЛЯЦИОННОЙ БАЗЫ ДАНЫХ В ИНФОРМАЦИОННОЙ СИСТЕМЕ

Рекомендуется выполнять денормализацию схемы отношений после тщательного тестирования производительности информационной системы. Предложена имитационная модель для тестирования поведения системы в условиях применения различных вариантов денормализации. На основе модели реализован программный инструмент, позволяющий исследовать базы данных, созданные с использованием различных систем управления базами данных.

ВВЕДЕНИЕ

Под денормализацией понимается процесс намеренного введения избыточности в нормализованных таблицах в целях увеличения производительности информационной системы (ИС). Производительность может быть повышена благодаря уменьшению времени выполнения запроса к БД, вызванному устранением операции соединения между отношениями либо сокращением размера отношений, участвующих в запросе. Необходимость такого изменения структуры базы данных (БД) становится очевидной лишь на этапе проектирования приложений [1]. Однако при определенных

обстоятельствах изменение структуры отношений не приводит к ожидаемому повышению производительности ИС. Например, в [2] описаны случаи, в которых устранение операции соединения в запросе, соответствующее методу нисходящей денормализации, не приводит к уменьшению времени выполнения запроса. Кроме того, сокращение времени выполнения одних запросов путем внедрения денормализации может быть нивелировано увеличением времени выполнения других запросов, использующих отношения, которые были реструктурированы, так что в целом производительность ИС может ухудшиться.

ПОСТАНОВКА ЗАДАЧИ

Существуют программные продукты, позволяющие анализировать эффективность запросов к БД. Например, Query Analyzer [3] позволяет просматривать план выполнения запроса, сгенерированный оптимизатором запросов MS SQL Server. Однако не существует универсальных инструментов, которые дают возмож-

© Кунгурцев А. Б., Зиноватная С. Л., 2008

ность проанализировать поведение системы после выполнения денормализации, в частности проверить влияние произведенных изменений в структуре отношений БД на время выполнения запросов различного типа.

Для решения данной задачи предложена имитационная модель для экспериментального исследования ИС в случае применения различных методов денормализации.

Применение модели производится после исследования ИС на предмет выявления возможных эффективных вариантов денормализации [4]. Однако реальная эффективность предложенных вариантов зависит от многих факторов, например, от используемой системы управления БД, от аппаратного обеспечения ИС, от загрузки системы, поэтому следует уточнить полученное значение прогнозируемой эффективности реструктуризации БД путем проведения эксперимента в условиях реальной ИС.

На основе модели для исследования эффективности денормализации БД реализован программный продукт, который позволяет решить следующие задачи:

1) сформировать условия для проведения эксперимента: создавать требуемую структуру отношений; задавать исходные данные для проведения эксперимента; заполнять отношения данными; изменять структуру таблиц в процессе проведения эксперимента;

2) выполнить набор запросов для определения изменений времени их выполнения для проведения сравнительного анализа поведения системы при различных исходных условиях: сканирования отношения до и после изменений в его структуре; устранения соединения отношений; соединения денормализованного отношения с другими отношениями;

3) просмотреть результатов работы программы: в табличном виде; в графическом виде.

Параметры модели, позволяющие изменять размер исследуемых отношений, включают следующее:

– исходное количество кортежей в исследуемых отношениях pC_0^R (pC_0^1 в родительском и pC_0^2 в подчиненном отношении соответственно);

– множества коэффициентов pKs (pKs^1 для изменения количества кортежей в родительском отношении, pKs^2 для изменения количества кортежей в подчиненном отношении и pKs^h для изменения размера кортежа реструктурируемого отношения соответственно).

Для проведения исследования эффективности денормализации необходимо указать значение входного параметра PCS , который задает текст строки соединения с исследуемой.

Указанная пользователем строка соединения используется для подключения к исследуемой БД, получения метаданных БД, формирования структуры тестовых отношений, изменения их состояния и выполнения запросов для проверки их поведения в различных условиях.

В исследуемой БД создаются отношения, предназначенные для проведения эксперимента. Структура отношений (количество, тип и размер атрибутов) зависит от рассматриваемых вариантов денормализации.

Созданные для проведения исследования отношения заполняются данными в количестве, указанном в исходных параметрах. Данные формируются в соответствии с типом заполняемого атрибута отношения. Тип атрибута определяется автоматически на основании метаданных. Содержимое экспериментальных отношений может быть сформировано с учетом характеристик данных в отношениях, эффективность денормализации которых исследуется.

К заполненным таким образом отношениям отправляется запрос на сканирование единственного отношения и/или соединение отношений. Запрос выполняется многократно, чтобы устранить возможность влияния случайных факторов, например, выполнения других задач в системе, на результат эксперимента. Полученные результаты усредняются.

Результаты эксперимента фиксируются в специально разработанной служебной БД. Это позволяет в дальнейшем пользователю просмотреть и проанализировать полученные данные.

В процессе работы программы пользователь может проводить любое количество экспериментов, а также работать с различными БД. Модель предоставляет возможность изменять параметры эксперимента, сохраняя при этом различные версии набора параметров.

СТРУКТУРА ЭКСПЕРИМЕНТАЛЬНЫХ ОТНОШЕНИЙ

Родительское отношение T1. Имеет начальный размер кортежа h_0^1 , совпадающий с размером кортежа исследуемого отношения.

Атрибутами отношения T1 являются:

– ключевой атрибут id числового типа;

– набор неключевых атрибутов $nonkey1_1, \dots, nonkey1_NT1$ строкового типа фиксированного размера, что позволяет обеспечить необходимый размер кортежа отношения.

Подчиненное отношение T2. Имеет начальный размер кортежа h_0^2 , совпадающий с размером кортежа исследуемого отношения.

Атрибутами отношения T2 являются:

– ключевой атрибут id числового типа;

– внешний ключевой атрибут $fkt1$ числового типа, служит для обеспечения связи с родительским отношением T1;

– набор неключевых атрибутов $nonkey1_1, \dots, nonkey1_NT2$ строкового типа фиксированного размера, что позволяет обеспечить необходимый размер кортежа отношения.

Родительское отношение T1D. В начале исследования имеет структуру, совпадающую со структурой отношения T1. В процессе проведения эксперимента размер кортежа этого отношения h_i^1 может изменяться в соответствии с условиями проведения эксперимента.

Атрибутами отношения T1D являются:

- ключевой атрибут id числового типа;
- набор неключевых атрибутов nonkeyt1_1, ..., nonkeyt1_NT1;

- добавляемый в отношение во время проведения эксперимента набор неключевых атрибутов AddP1, ..., AddPNT1D строкового типа фиксированного размера, что позволяет обеспечить необходимый размер кортежа отношения h_i^1 ; размер каждого такого атрибута и их общее количество NT1D изменяется в процессе проведения эксперимента в соответствии с заданными параметрами.

Подчиненное отношение T2D. В начале исследования имеет структуру, совпадающую со структурой отношения T2. В процессе проведения эксперимента размер кортежа этого отношения h_j^2 может изменяться в соответствии с условиями проведения эксперимента.

Атрибутами отношения T2D являются:

- ключевой атрибут id числового типа;
- внешний ключевой атрибут fkt1 числового типа, служит для обеспечения связи с родительским отношением T1;
- набор неключевых атрибутов nonkeyt2_1, ..., nonkeyt2_NT2 строкового типа фиксированного размера;

- добавляемый в отношение во время проведения эксперимента набор неключевых атрибутов AddP1, ..., AddPNT2D строкового типа фиксированного размера; размер каждого такого атрибута и их общее количество NT2D изменяется в процессе проведения эксперимента в соответствии с заданными параметрами.

На рис. 1 приведена схема данных для экспериментального множества отношений.

СТРУКТУРА СЛУЖЕБНОЙ БД

В результате проведения эксперимента необходимо сохранять следующую информацию:

- идентификатор сеанса эксперимента (под сеансом понимается один цикл прохождения по всем коэффициентам изменения количества кортежей родительского отношения, подчиненного отношения и размера кортежа реструктурируемого отношения);
- строку соединения с исследуемой БД;
- дату и время проведения эксперимента;
- параметры проведения эксперимента (начальные значения для количества кортежей, коэффициенты их изменения, коэффициенты изменения размера кортежа исследуемого отношения, режим проведения эксперимента);
- время выполнения запросов к экспериментальным отношениям.

В процессе проведения эксперимента в течение одного сеанса происходит выполнение следующих вложенных циклов.

1) Цикл по количеству кортежей в родительском отношении:

$$i = C1_Start \text{ to } C1_End \text{ step } k1,$$

где $C1_Start$ – исходное количество кортежей в отношении T1, соответствует pC_0^1 ; $C1_End$ – конечное значение количества кортежей в отношении T1, определяется в зависимости от содержимого массива коэффициентов увеличения количества кортежей в родительском отношении; $k1$ – шаг изменения, предполагает увеличение текущего i -го значения количества кортежей в отношении в pKs_i^1 раз, где pKs_i^1 – i -е значение соответствующего множества коэффициентов pKs_i^1 ; определяется параметрами проведения эксперимента; может быть различным в разных итерациях цикла.

Количество кортежей в родительском отношении на i -й итерации цикла определяется следующим образом:

$$C_i^1 = pC_0^1 \cdot pKs_i^1.$$

2) Цикл по количеству кортежей в подчиненном отношении, выполняется на каждой итерации цикла для родительского отношения:

$$j = C2_Start \text{ to } C2_End \text{ step } k2,$$

где $C2_Start$ – исходное количество кортежей в отношении T2, соответствует pC_0^2 ; $C2_End$ – конечное значение количества кортежей в отношении T1, определя-

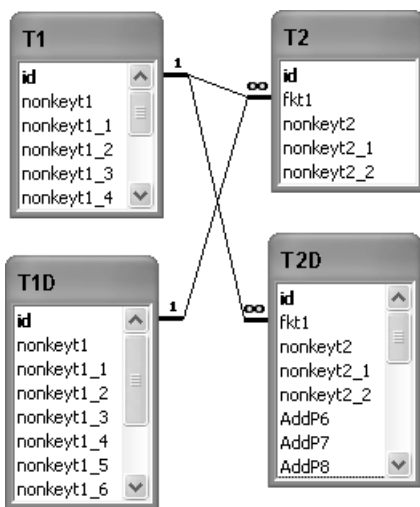


Рисунок 1 – Схема данных экспериментальных отношений

ется в зависимости от содержимого множества коэффициентов увеличения количества кортежей в подчиненном отношении; $k2$ – шаг увеличения, предполагает увеличение текущего j -го значения количества кортежей в отношении в pKs_i^2 раз, где pKs_i^2 – j -е значение соответствующего множества коэффициентов pKs^2 , которое определяется параметрами проведения эксперимента; может быть различным в разных итерациях цикла.

Количество кортежей в подчиненном отношении на j -й итерации цикла определяется следующим образом:

$$C_j^2 = pC_0^2 \cdot pKs_j^2.$$

3) Цикл, задающий увеличение размера кортежа исследуемого отношения, выполняется на каждой итерации цикла для исследуемого отношения:

$$k = k_Start \text{ to } k_End,$$

где k_Start – исходное значение коэффициента изменения размера кортежа pKs_0^h ; k_End – конечное значение коэффициента изменения размера кортежа.

Текущее значение pKs_k^h определяет, во сколько раз размер добавляемых атрибутов исследуемого отношения изменяется по сравнению с начальным значением (определенным при создании отношения).

Размер кортежа h_k^R реструктурируемого отношения R на k -й итерации цикла определяется по формуле:

$$h_k^R = h_0^R \cdot (1 + pKs_k^h),$$

где h_0^R – начальный размер кортежа.

На каждой итерации цикла измеряется время выполнения запросов к исследуемой БД.

Предложенная модель включает следующие варианты сравнения полученных в результате исследования показателей:

TTD – соотношение времени сканирования исходного отношения и этого же отношения с измененным размером кортежа;

T12T1D – соотношение времени сканирования соединения родительского и подчиненного отношений и родительского денормализованного отношения.

T12T2D – соотношение времени сканирования соединения родительского и подчиненного отношений и подчиненного денормализованного отношения.

Для хранения информации, использованной и полученной в результате исследования, необходимы следующие объекты:

O_CEAHC – для хранения информации о сеансе эксперимента;

O_C2 – для хранения данных о значениях количества кортежей в подчиненном отношении во время кон-

кретной итерации цикла по увеличению количества кортежей в родительском отношении;

O_C1 – для хранения данных о значениях количества кортежей в родительском отношении во время конкретного сеанса;

O_K – для хранения данных о значениях коэффициента изменения размера кортежа исследуемого отношения и времени выполнения соответствующих запросов на конкретной итерации цикла по увеличению количества кортежей в подчиненном отношении.

Между перечисленными объектами существуют следующие взаимосвязи:

– объект O_CEAHC находится во взаимосвязи «один ко многим» с объектом O_C2 , так как во время эксперимента может быть получено несколько значений O_C2 для одного сеанса;

– объект O_C2 находится во взаимосвязи «один ко многим» с объектом O_C1 , так как во время эксперимента может быть получено несколько значений O_C1 для одного O_C2 ;

– объект O_C1 находится во взаимосвязи «один ко многим» с объектом O_K , так как во время эксперимента может быть получено несколько значений O_K для одного O_C1 .

На основании приведенных отношений между объектами создана служебная БД dbExp.

Отношение Seans соответствует объекту O_CEAHC , имеет ключевой атрибут, являющийся уникальным идентификатором сеанса.

Отношение SeansC2 соответствует объекту O_C2 , имеет ключевой атрибут; внешний ключ для связи с отношением Seans; атрибут qc2 хранит текущее значение количества кортежей в отношении T2; атрибут tm2 хранит время сканирования отношения T2 при текущем количестве кортежей qc2.

Отношение SeansC1 соответствует объекту O_C1 , имеет ключевой атрибут; внешний ключ для связи с отношением SeansC2; атрибут qc1 хранит текущее значение количества кортежей в отношении T1; атрибут tm1 хранит время сканирования отношения T1 при текущем количестве кортежей qc1.

Отношение SeansK соответствует объекту O_K , имеет ключевой атрибут; внешний ключ для связи с отношением SeansC1; атрибут vk хранит текущее значение коэффициента изменения размера кортежа реструктурируемого отношения; атрибут tm2d хранит время сканирования отношения T2D при текущем количестве кортежей qc2 и текущем значении размера кортежа отношения T2D; атрибут tm12 хранит время соединения отношений T1 и T2 при текущем количестве кортежей qc2 и qc1; атрибут tm1d хранит время сканирования отношения T1 при текущем количестве кортежей qc1 и текущем значении размера кортежа отношения T1.

Схема данных dbExp приведена на рис. 2.

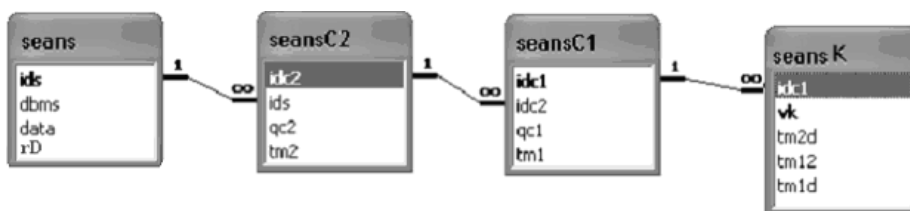


Рисунок 2 – Схема данных для служебной БД

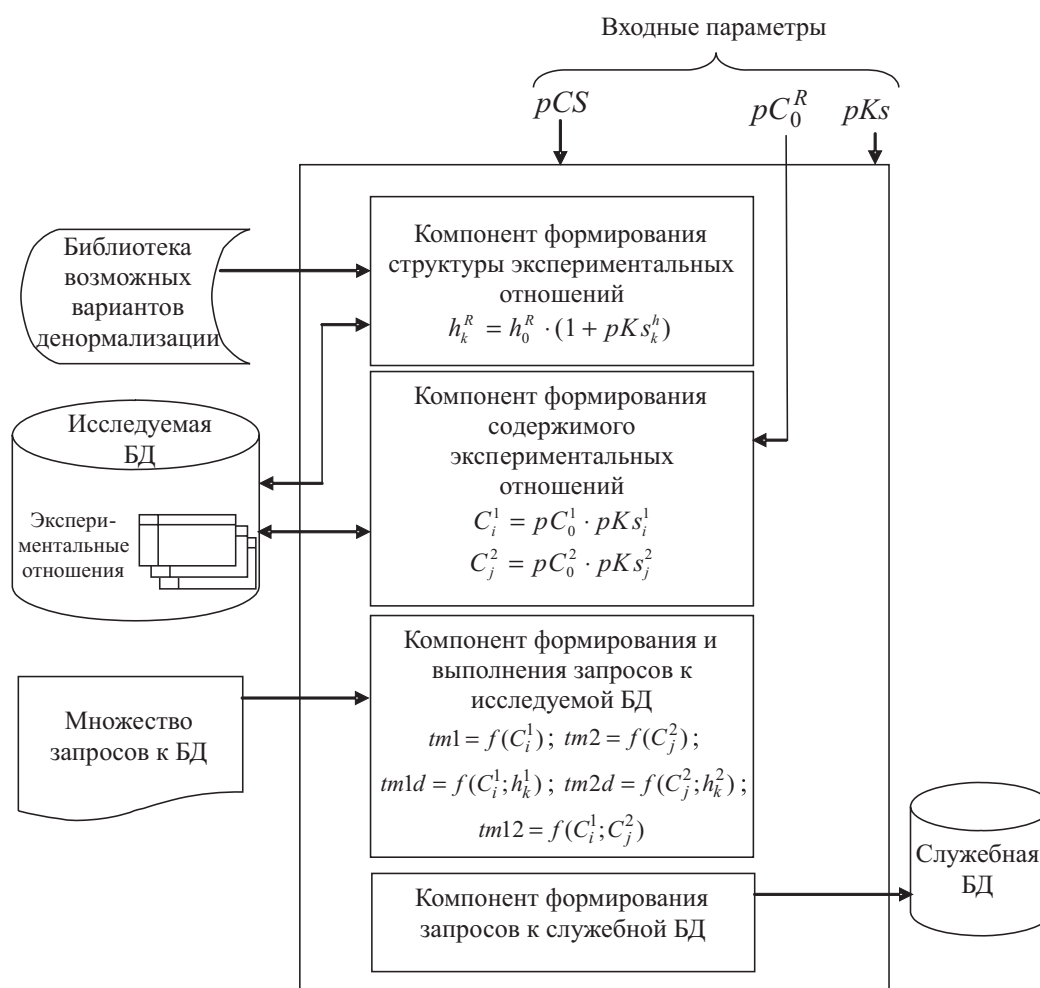


Рисунок 3 – Общая схема имитационной модели

Общая схема имитационной модели для исследования эффективности денормализации реляционной БД приведена на рис. 3.

СТРУКТУРА ФАЙЛА С ПАРАМЕТРАМИ ПРОВЕДЕНИЯ ЭКСПЕРИМЕНТА

Файл eff.ini (по умолчанию) является обычным текстовым файлом и может быть создан с использованием интерфейса программы или в текстовом редакторе.

Файл должен иметь следующую структуру. Для каждого входного параметра модели должны присутствовать пары строк. Первая строка в каждой паре содержит имя параметра в квадратных скобках. Определены следующие имена параметров:

- C1 – исходное количество кортежей в родительском отношении;
- C2 – исходное количество кортежей в подчиненном отношении;
- K1 – множество коэффициентов увеличения количества кортежей в родительском отношении;



Рисунок 4 – Заполненный файл параметров эксперимента

K2 – множество коэффициентов увеличения количества кортежей в отношении подчиненном;

K – множество коэффициентов изменения размера кортежа исследуемого отношения.

Вторая строка содержит одно или несколько числовых значений, соответствующих набору параметров. Коэффициенты должны располагаться в порядке возрастания значений. Количество коэффициентов в каждом множестве не ограничено.

Пары строк могут быть расположены в файле в произвольном порядке.

На рис. 4 приведен образец заполненного файла параметров проведения эксперимента. Из рисунка видно, что эксперимент проводится при двух значениях количества кортежей в отношении T1 и двух значениях количества кортежей в отношении T2, то есть рассматриваются четыре комбинации размера исследуемых отношений T1 и T2 соответственно: 10^5-10^5 , 10^5-10^6 , 10^6-10^5 , 10^6-10^6 . Для каждой комбинации выполняет-

ся изменение размера кортежа исследуемого отношения в 1,1 раза, в 2 раза, в 3 раза, в 6 раз.

АНАЛИЗ РЕЗУЛЬТАТОВ ЭКСПЕРИМЕНТА С ПРИМЕНЕНИЕМ ПРЕДЛОЖЕННОЙ ИМИТАЦИОННОЙ МОДЕЛИ

Ниже приведено описание эксперимента для различных методов денормализации.

Можно выделить следующие методы денормализации: нисходящая денормализация; восходящая денормализация; ввод повторяющихся групп; горизонтальная денормализация; вертикальная денормализация; ввод производных данных; ввод суррогатного ключа; устранение связи 1:1 [5].

Требования к определению входных параметров и режим работы программы для различных методов денормализации приведены в табл. 1.

В зависимости от выбранного режима работы программы происходит изменение размера кортежа исходного или подчиненного отношения для методов денормализации, предусматривающих устранение операции соединения между отношениями, или изменение размера кортежа единственного отношения.

1. Нисходящая денормализация. Уменьшение времени выполнения запросов может быть получено за счет устранения операции соединения между отношениями. При этом увеличивается размер кортежа подчиненного отношения, что приводит к увеличению общего объема памяти, занимаемого этим отношением. Следовательно, вероятно, увеличивается время сканирования этого отношения (зависит от размера добавляемого атрибута и количества кортежей в отношении).

Эксперимент позволяет измерить время сканирования исходного подчиненного отношения tm_2 и денор-

Таблица 1 – Правила определения входных параметров для различных методов денормализации

Метод денормализации	Режим	C1	C2	K1	K2	K
Нисходящая	T12T2D	#1	#2	#5	#5	#5
Восходящая	T12T1D	#1	#2	#5	#5	#5
Ввод повторяющихся групп	T12T1D	#1	#2	#5	#5	#5
Горизонтальная	TTD	#3	#4	#6	не задается	#5
Вертикальная	TTD	#3	#4	#6	не задается	#5
Ввод производных значений	TTD	#3	#4	#5	не задается	#5
Ввод суррогатного ключа	T12T2D	#1	#2	#5	#5	#5
Устранение связи «1:1»	T12T1D	#1	#2	#5	#5	#5
<p>Ссылки:</p> <p>#1 выбирается, исходя из существующего или возможного значения количества кортежей в родительском отношении</p> <p>#2 выбирается, исходя из существующего или возможного значения количества кортежей в подчиненном отношении</p> <p>#3 выбирается, исходя из существующего или возможного значения количества кортежей в исследуемом отношении</p> <p>#4 выбирается произвольно (не влияет на результат эксперимента)</p> <p>#5 может содержать различные коэффициенты</p> <p>#6 может содержать различные коэффициенты, позволяющие проверить поведение исследуемого отношения при разном размере его фрагментов</p>						

мализованного подчиненного отношения $tm2d$, а также время соединения отношений $tm12$.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения нисходящей денормализации от количества кортежей в родительском и подчиненном отношениях и от размера добавляемого в подчиненное отношение атрибута.

2. Восходящая денормализация. Уменьшение времени выполнения запросов может быть получено за счет устранения операции соединения между отношениями, подобно нисходящей денормализации. При этом увеличивается размер кортежа родительского отношения, что приводит к увеличению общего объема памяти, занимаемого этим отношением. Следовательно, вероятно, увеличивается время сканирования этого отношения (зависит от размера добавляемого атрибута и количества кортежей в отношении).

Эксперимент позволяет измерить время сканирования исходного родительского отношения $tm1$ и денормализованного подчиненного отношения $tm1d$, а также время соединения отношений $tm12$.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения восходящей денормализации от количества кортежей в родительском и подчиненном отношениях и от размера добавляемого в родительское отношение атрибута.

3. Ввод повторяющихся групп. Уменьшение времени выполнения запросов может быть получено за счет устранения операции соединения между отношениями, подобно нисходящей и восходящей денормализации. При этом увеличивается размер кортежа родительского отношения, что приводит к увеличению общего объема памяти, занимаемого этим отношением. Следовательно, вероятно, увеличивается время сканирования этого отношения (зависит от размера добавляемых повторяющихся групп и количества кортежей в отношении).

Эксперимент позволяет измерить время сканирования исходного родительского отношения $tm1$ и денормализованного родительского отношения $tm1d$, а также время соединения таблиц $tm12$.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения денормализации вводом повторяющихся групп от количества кортежей в родительском и подчиненном отношениях и от размера добавляемых в родительское отношение атрибутов.

4. Горизонтальная денормализация. Уменьшение времени выполнения запросов может быть получено за счет уменьшения объема памяти, занимаемого каждой частью денормализуемого отношения.

Эксперимент позволяет измерить время сканирования исходного отношения $tm1$ при различных значениях количества кортежей в отношении.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения горизонтальной денормализации от количества кортежей в отношении.

5. Вертикальная денормализация. Уменьшение времени выполнения запросов может быть получено за счет уменьшения объема памяти, занимаемого каждой частью денормализуемого отношения.

Эксперимент позволяет измерить время сканирования исходного отношения $tm1$ и денормализованного отношения $tm1d$ при различных значениях размера кортежа в отношении и различных значениях количества кортежей.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения вертикальной денормализации от размера кортежа в отношении и от общего размера отношения.

6. Ввод производных значений. Уменьшение времени выполнения запросов может быть получено за счет сокращения времени расчета производного значения в момент выполнения запроса, но может произойти увеличение времени выполнения запроса из-за увеличения объема памяти, занимаемого отношением после денормализации.

Эксперимент позволяет измерить время сканирования исходного отношения $tm1$ и денормализованного отношения $tm1d$ при различных значениях размера кортежа в отношении.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения производных значений от размера кортежа в отношении (зависит от размера добавляемого атрибута) и от общего размера отношения.

7. Ввод суррогатного ключа. Уменьшение времени выполнения запросов может быть получено за счет уменьшения объема памяти, занимаемого подчиненным отношением после денормализации.

Эксперимент позволяет измерить время сканирования исходного подчиненного отношения $tm1$ и денормализованного отношения $tm1d$, а также время соединения отношений $tm12$.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности внедрения ввода суррогатного ключа от размера кортежа в отношении и от общего размера отношений.

8. Устранение связи «один к одному». Уменьшение времени выполнения запросов может быть получено за счет устранения операции соединения между отношениями.

Эксперимент позволяет измерить время сканирования исходного родительского отношения $tm1$ и денормализованного отношения $tm1d$ и время соединения отношений $tm12$.

Используя полученные в ходе эксперимента показатели, можно получить функцию зависимости эффективности устранения связи «один к одному» от размера кортежа в отношениях и от общего размера отношений.

Предложенная имитационная модель исследования эффективности денормализации реляционной БД позволяет проверить поведение ИС при внедрении основных методов денормализации. Это обеспечивается за счет гибкости структуры входных параметров. Режимы работы реализованной на основе данной модели программы отличаются незначительно и обеспечиваются использованием сходных алгоритмов. В результате проведения эксперимента можно получить зависимости в табличном и графическом виде. Необходимые для получения зависимостей данные могут быть получены за счет формулировки различных запросов к служебной БД.

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТА

В табл. 2 представлены частичные данные о поведении БД, созданной в системе управления базами данных MS Access, полученные в результате проведения серии экспериментов в режиме T12T2D.

Эксперимент проводился многократно при различных количествах кортежей в исходном и подчиненном отношении и при различном размере добавляемого

атрибута. Табл. 2 содержит сведения о количестве кортежей в исходном отношении qc1 и в подчиненном отношении qc2 соответственно, о коэффициенте vk, показывающем, во сколько раз размер добавленного атрибута больше размера кортежа исходного отношения, а также усредненные значения для времени сканирования денормализованного подчиненного отношения tm2d, времени сканирования исходного подчиненного отношения tm2 и времени выполнения операции соединения между исходными отношениями и tm12. Данные получены с помощью запроса к БД dbExp.

```
SELECT qc2, qc1, vk, Avg (tm2), Avg (tm2d), Avg (tm12)
FROM (seans INNER JOIN (seansC2 INNER JOIN
seansC1 ON seansC2.idc2 = seansC1.idc2)
ON seans.ids = seansC2.ids) INNER JOIN seansK ON
seansC1.idc1 = seansK.idc1
GROUP BY qc2, qc1, vk
```

В качестве примера результата анализа поведения ИС после реструктуризации отношений БД представлены графики (см. рис. 5), построенные на основе данных из табл. 2 для исследования эффективности нисходящей денормализации.

Введем понятие эффективности устранения операции соединения в единственном запросе E_{join} как отношение времени выполнения запроса с операцией соединения ко времени выполнения запроса со сканированием

Таблица 2 – Результаты эксперимента

.qc1	qc2	vk	Ср. знач. tm2d	Ср. знач. tm2	Ср. знач. tm12	qc1	qc2	vk	Ср. знач. tm2d	Ср. знач. tm2	Ср. знач. tm12
1000	1000	0.25	0.0132	0.0155	0.0141	10000	50000	1.25	1.5852	0.7110	8.8610
1000	1000	0.5	0.0141	0.0155	0.0133	10000	50000	1.5	1.5852	0.7110	8.8610
1000	1000	0.75	0.0156	0.0155	0.0141	10000	50000	1.75	1.5844	0.7110	8.1882
1000	1000	1	0.0157	0.0155	0.0164	10000	50000	2	1.5890	0.7110	9.2336
1000	1000	1.25	0.0140	0.0155	0.0179	10000	100000	0.25	2.5016	1.5160	18.4320
1000	1000	1.5	0.0140	0.0155	0.0204	10000	100000	0.5	4.3391	1.4380	19.7679
1000	1000	1.75	0.0133	0.0155	0.0172	10000	100000	0.75	5.5125	1.5160	16.6859
1000	1000	2	0.0172	0.0155	0.0195	10000	100000	1	8.2703	1.4327	22.7860
1000	10000	0.25	0.1547	0.1250	0.8590	10000	100000	1.25	8.4898	1.5160	22.1125
1000	10000	0.5	0.1984	0.1250	1.0310	10000	100000	1.5	8.7898	1.4380	20.0258
1000	10000	0.75	0.1953	0.1250	0.7970	10000	100000	1.75	8.7703	1.5160	18.4008
1000	10000	1	0.3109	0.1250	0.9530	10000	100000	2	8.2797	1.4380	19.6336
1000	10000	1.25	0.3509	0.1250	0.7190	50000	100000	0.25	2.9640	1.7627	105.7517
1000	10000	1.5	0.3725	0.1250	0.6410	50000	100000	0.5	4.3750	1.7369	118.5603
1000	10000	1.75	0.3809	0.1250	0.7660	50000	100000	0.75	4.4805	1.7762	99.6186
1000	10000	2	0.3410	0.1250	0.9680	50000	100000	1	6.3235	1.7521	114.1672
10000	50000	0.25	0.8586	0.7110	10.3102	50000	100000	1.25	7.5875	1.7875	98.0344
10000	50000	0.5	0.9436	0.7110	10.3102	50000	100000	1.5	7.9870	1.7003	115.3261
10000	50000	0.75	1.0500	0.7110	8.3633	50000	100000	1.75	8.0757	1.6719	104.0023
10000	50000	1	1.2500	0.7110	8.3633	50000	100000	2	8.1937	1.6973	113.3766

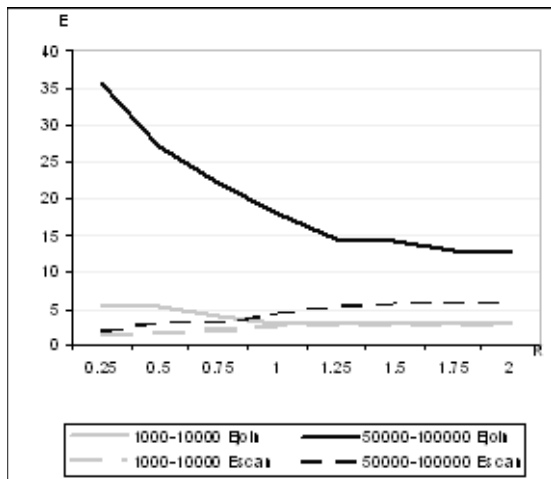


Рисунок 5 – Соотношение времени выполнения запросов

реструктуризованного підчиненого отношения

$$E_{\text{join}} = \frac{tm12}{tm2d}$$

Введем также понятие эффективности изменения структуры отношения для единственного запроса E_{scan} как отношение времени выполнения запроса на сканирование исходного отношения ко времени выполнения запроса со сканированием реструктуризованного отношения $E_{\text{scan}} = \frac{tm2d}{tm2}$.

Графики на рис. 5 представляют зависимость E_{join} (сплошная линия) и E_{scan} (пунктирная линия) от значения коэффициента K , определяющего во сколько раз размер (в байтах) добавленных атрибутов больше размера исходного кортежа, при различных количествах кортежей в исследуемых отношениях (10^3 в родительском отношении и 10^4 в подчиненном отношении – светлая линия, $5 \cdot 10^4$ в родительском отношении и 10^5 в подчиненном отношении – темная линия).

Результаты проведения эксперимента с использованием предложенной имитационной модели показывают, что эффект от применения нисходящей денормализации в значительной степени зависит от размера отношения определяется количеством записей и размером кортежа. Если размер отношений сопоставим с размером оперативной памяти компьютера, на котором выполняются запросы с использованием операции соединения, можно получить значительный эффект повышения производительности ИС. Однако эффективность снижается при увеличении размера добавляемых в подчиненное отношение атрибутов.

Из графика видно, что при малых значениях количества кортежей в отношениях и при размере добавленных атрибутов, близком к исходному размеру кор-

тежа, величина сокращения времени выполнения запросов путем устранения операции соединения, становится сопоставимой с величиной увеличения времени выполнения запросов на сканирование реструктуризованного отношения.

Реальный эффект от проведения денормализации с использованием исследуемого варианта определяется при учете количества запросов с использованием соединения отношений и запросов со сканированием денормализованного отношения за период исследования ИС. Так, например, при малом количестве кортежей в отношениях, если размер добавленного атрибута составляет половину размера подчиненного отношения, то уже при двукратном превышении количества запросов, требующих сканирования подчиненного отношения, по сравнению с количеством запросов, использующих операцию соединения с данным отношением, в системе наблюдается снижение производительности.

Имея данные о частоте использования разных типов запросов, можно провести дальнейшее экспериментальное исследование для получения конкретного результата, соответствующего текущему состоянию БД.

ВЫВОДЫ

Предложенная имитационная модель для исследования эффективности денормализации БД позволяет разработчику получить информацию о целесообразности проведения реструктуризации отношений БД. Универсальность разработанной на основе этой модели программной системы состоит в том, что она позволяет проводить эксперименты с различными системами управления базами данных, с использованием различного аппаратного обеспечения, а также для различных методов денормализации. Накапливаемая в процессе проведения эксперимента информация дает возможность путем формирования соответствующих запросов к служебной БД сопоставить измеренные величины для получения оценок поведения системы в случае применения различных методов денормализации.

ПЕРЕЧЕНЬ ССЫЛОК

1. Принятие решения о денормализации. – Режим доступа: <http://lavsoft.webhost.ru/oracle/book/dezdb/04/04.html>. – Название с экрана.
2. Кунгурцев А. Б. Анализ целесообразности реструктуризации базы данных методом введения нисходящей денормализации / Кунгурцев А. Б., Зиноватная С. Л. – Тр. Одесск. политехн. ун-та. Одесса, 2006. – 1(25). – С. 104–108.
3. М. Ф. Гарсиа. Справочник администратор Microsoft SQL Server 2000. 2-е изд. / Пер с англ. / М. Ф. Гарсиа, Дж. Реддинг, Э. Уолен, С. А. ДеЛюк. – М.: Издательство «СП ЭКОМ», 2004. – 976 с.
4. Зиноватная С. Л. Анализ информационной системы для принятия решения о выполнении восходящей денормализации // Холодильная техника и технологии. Сборник научных трудов IV семинара «Информационные системы и технологии». – Одесса, 2006. – С. 14–18.

5. Кунгурцев А. Б. Модель реструктуризації реляційної бази даних путем денормалізації схеми / Кунгурцев А. Б., Зиноватная С. А. – Тр. Одесск. политехн. ун-та. – Одесса, 2006. – 2(26). – С. 105–111.

Надійшла 27.11.2007
Після доробки 19.03.2008

Рекомендується виконувати денормалізацію схеми відносин після ретельного тестування продуктивності інформаційної системи. Запропоновано імітаційну модель для тестування поведінки системи в умовах застосування різних варіантів денормалізації. На основі моделі реалізований програмний інструмент, що дозволяє дослід-

жувати бази даних, створені з використанням різних систем керування базами даних.

It is recommended to execute denormalization of relations scheme after the careful testing of the information system productivity. A simulation model for testing of the system conduct in the conditions of application of different denormalization variants is offered. On the basis of this model a program instrument is realized to allow to probe databases, created with the use of the different database management systems.

681.51:622.788

В. Й. Лобов, С. А. Рубан

РОЗРОБКА ПРИНЦИПІВ КЕРУВАННЯ ТЕМПЕРАТУРНИМ РЕЖИМОМ ПРОЦЕСУ ВИПАЛЮВАННЯ КОТУНІВ З ВИКОРИСТАННЯМ ПРОГНОЗУЮЧИХ ANFIS-МОДЕЛЕЙ

Запропоновано принцип керування температурним режимом процесу випалювання котунів на конвеєрній машині з використанням прогнозуючої ANFIS-моделі.

ВСТУП

Відомим є факт, що збільшення продуктивності, підвищення якості готової продукції та зменшення питомої витрати енергоносіїв можна досягти за рахунок удосконалення автоматизованих систем управління технологічними процесами (АСУ ТП). Одним з найперспективніших напрямків розвитку АСУ ТП гірничо-збагачувальних комбінатів є використання теорії керування на основі прогнозуючих моделей (Model Predictive Control, MPC), що забезпечують високі адаптивні властивості та можливість квазіоптимального керування нелінійними об'єктами в режимі реального часу з урахуванням обмежень на керуючі і вихідні змінні [1, 2]. В базовому варіанті MPC в якості моделі об'єкту керування використовується лінійна дискретна модель у просторі станів [2]. Однак відомо, що для більшості технологічних процесів гірничо-збагачувальних комбінатів, зокрема і для процесу температурної обробки котунів на конвеєрній випалювальній машині, достатньо складно отримати адекватну математичну модель. Це пов'язано з багатомірністю та стохастичними властивостями об'єктів керування, нестационарністю, наявністю великих часових запізнь, неможливістю безперервного контролю окремих технологічних показників [3, 4]. Тому для вирішення цієї проблеми можуть бути використані нелінійні багатомірні математичні моделі, зокрема нейромережеві та нечіткі.

Так, в роботі [1] в схемі MPC використовується лінійна дискретна модель у просторі станів, яка знаходиться з наслідків нечітких правил типу Такаґі – Сугено прямим способом. В роботі [5] запропоновано для прогнозування показників процесу збагачення використовувати нейромережеві моделі NNARX (нейромережева авторегресійна модель, екзогенний тип сигналів) та NNOE (нейромережева модель помилки виходу).

Для отримання високих показників якості котунів при забезпеченні ресурсозберігаючої технології необхідно забезпечити бажаний розподіл температур уздовж випалювальної машини шляхом стабілізації температури в кожній газоповітряній камері. Регулювання температури котунів в різних зонах випалювальної машини здійснюється шляхом змінення кількості теплоносія, що фільтрується через шар. Наприклад, розглянемо зону сушки 2. В цю зону газ-теплоносії з температурою 900–1000 °С надходить прямим перетоком із зони охолодження 1. Задане значення температури на вході в шар (350–450 °С) підтримується за рахунок розбавлення теплоносія холодним повітрям із зони охолодження 2, що подається через бічні форкамери. Змінення витрати холодного повітря дозволяє також компенсувати збурення, пов'язані зі змінами висоти шару котунів, швидкості візків випалювальної машини, температури та тиску теплоносія, що надходить в зону сушки 2 із зони охолодження прямим перетоком [6].

В існуючих системах автоматизації стабілізація температури в газоповітряних камерах зони сушки 2 здійснюється шляхом змінення витрати холодного по-