УДК 519.677

МОДЕЛЬ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ БАЗ ДАННЫХ С КОЛОНОЧНОЙ СТРУКТУРОЙ

И.А. Кулик, *канд. техн. наук, доцент;*
В.В. Гриненко, *ст. преподаватель;*
С.В. Костель, *инженер;*
Е.М. Скордина, *мл. научн. сотр.,*
Сумский государственный университет,
ул. Римского-Корсакова, 2, г. Сумы, 40007, Украина

В статье рассматривается модель векторного представления реляционных баз данных с колоночной структурой. Приводится формализованное описание векторной модели и основных этапов моделирования реляционных баз данных с колоночной структурой хранения информации. Предлагаются выражения для оценки объема памяти, занимаемого данными до и после применения векторной модели.

Ключевые слова: *векторное моделирование данных, реляционные базы данных, отношение, атрибут, домен, биномиальное нумерационное сжатие.*

ВВЕДЕНИЕ

Развитие современных информационных технологий неразрывно связано с базами данных (БД). Активное использование сетевых и интернет технологий, облачных вычислений, обуславливают повышенные требования к оперативности получения и изменения информации в БД. Одним из наиболее важных параметров функционирования БД является время доступа к данным, от которого напрямую зависит время выполнения запроса к базе или хранилищу данных.

В данной работе предлагается использовать векторную модель представления данных применительно к БД, что позволит повысить скорость выполнения наиболее распространенных запросов к БД. Кроме того, векторное моделирование данных является предварительным этапом для биномиального сжатия БД [1]. Совокупность модели векторного представления БД и биномиального нумерационного сжатия уменьшает время выполнения запросов к БД за счет повышения пропускной способности каналов доступа к данным и уменьшения времени обработки данных в сжатом виде.

ВЕКТОРНОЕ МОДЕЛИРОВАНИЕ ДАННЫХ

Одним из методов моделирования повторяющихся данных является векторное моделирование [2]. Определим понятие векторной модели следующим образом.

Пусть имеется последовательность $S=\left(s_{n-1},\ldots,s_i,\ldots,s_1,s_0\right)$, состоящая из n повторяющихся сообщений $d_j \in D$, $j \in \overline{0,(m-1)}$, $m=\left|D\right|$, множества сообщений D. Векторная модель представления данных ставит в соответствие некоторому сообщению d_j в последовательности сообщений $S=\left(s_{n-1},\ldots,s_i,\ldots,s_1,s_0\right)$ двоичный вектор $Y_j=\left(y_{j(n-1)},\ldots,y_{ji},\ldots,y_{j1},y_{j0}\right)$, $y_{ji} \in \overline{0,1}$, $i \in \overline{0,(n-1)}$, содержащий единицы в тех позициях, номера которых соответствуют позициям сообщения d_j в последовательности сообщений S:

$$y_{ji} = \begin{cases} 1, s_{ji} = d_j \\ 0, s_{ji} \neq d_j \end{cases} \quad (1)$$

Результатом векторного моделирования сообщения d_j в последовательности сообщений $S = (s_{n-1}, \ldots, s_i, \ldots, s_1, s_0)$ будет само сообщение d_j и соответствующий ему n-битный двоичный вектор $Y_j = \left(y_{j(n-1)}, \ldots, y_{ji}, \ldots, y_{j1}, y_{j0}\right)$.

Например, для последовательности сообщений

$$S = [c\ f\ c\ c\ a\ c\ f\ f\ c\ f\ a\ f\ f\ f\ a\ f\ c\ c\ a\ f\ c\ f\ f\ f],$$

двоичный вектор Y, соответствующий сообщению f будет иметь вид

$$f: 0100001101011101000010111.$$

Векторную модель представления данных можно применить для всех сообщений $d_j \in D$, размещенных в последовательности сообщений S.

Модель векторного представления последовательности данных ставит в соответствие множеству сообщений $d_j \in D$, размещенных в последовательности $S = (s_{n-1}, \ldots, s_i, \ldots, s_1, s_0)$, множество двоичных векторов $Y_j = \left(y_{j(n-1)}, \ldots, y_{ji}, \ldots, y_{j1}, y_{j0}\right)$, $y_{ji} \in \overline{1,0}$, $j \in \overline{0,(m-1)}$, $i \in \overline{0,(n-1)}$, в соответствии с выражением (1).

Например, для множества сообщений $D = \{d_0, d_1, d_2, d_3\}$, входящих в последовательность S сообщений вида

$$S = [d_3\ d_0\ d_0\ d_0\ d_0\ d_1\ d_0\ d_2\ d_2\ d_0\ d_1\ d_1\ d_1\ d_3\ d_2\ d_3],$$

векторное представление данных будет иметь вид:

d_0:	0	1	1	1	1	0	1	0	0	1	0	0	0	0	0	0
d_1:	0	0	0	0	0	1	0	0	0	0	1	1	1	0	0	0
d_2:	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0
d_3:	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

Пусть имеется некоторая функция $size(d_j)$, которая определяет размер сообщения d_j в битах. Тогда объем памяти, занимаемой исходной последовательностью S определяется выражением:

$$Ls(D) = \sum_{j=0}^{m-1} k_{d_j} \cdot size(d_j), \quad (2)$$

где $size(d_j)$ - число бит для хранения сообщения d_j;

$m = |D|$ - число уникальных сообщений $d_j \in D$ в последовательности S;

k_{d_j} - число сообщений d_j в последовательности S, $k_{d_j} \in \overline{1,n}$, $\sum_j k_{d_j} = n$.

Объем памяти, необходимый для хранения векторного представления последовательности S сообщений d_j определяется выражением:

$$Lv(D) = \sum_{j=0}^{m-1} \left(size(d_j) + n \right) = \sum_{j=0}^{m-1} \left(size(d_j) \right) + m \cdot n \,. \qquad (3)$$

Выполнив сравнение выражений (2) и (3), можно сделать вывод, что для последовательности S, состоящей из уникальных записей ($k_{d_j} = 1$), объем памяти для векторной модели представления последовательности S буде превышать объем памяти для хранения последовательности S в исходном виде:

$$\left\{ Lv(D) > Ls(D), k_{d_j} = 1 \right\}. \qquad (4)$$

Свойство (4) показывает нецелесообразность векторного представления информации для последовательностей, состоящих из уникальных значений.

При условии, что количество бит $size(d_j)$ для хранения сообщений d_j является одинаковым для всех сообщений и равно $size(d)$, можно определить максимальное число $m = |D|$ сообщений $d_j \in D$ в последовательности $S = (s_{n-1}, \ldots, s_i, \ldots, s_1, s_0)$ в зависимости от длины последовательности n и размера сообщений $size(d)$, при котором векторная модель представления последовательности сообщений будет занимать меньший объем, чем исходное сообщение ($Lv(D) < Ls(D)$):

$$m < \frac{n \cdot size(d)}{n + size(d)} \,. \qquad (5)$$

Граничное значение числа m сообщений d_j в выражении (5) можно увеличить, если выполнить сжатие двоичных векторов. В качестве метода сжатия целесообразно использовать биномиальное нумерационное сжатие, построенное на основе биномиальной системы счисления [3].

МОДЕЛЬ ВЕКТОРНОГО ПРЕДСТАВЛЕНИЯ БАЗ ДАННЫХ

Среди множества БД наиболее разработанными, распространенными и востребованными являются реляционные базы данных (РБД) [4]. РБД можно представить в виде множества связанных отношений R (таблиц значений). Сжатие БД будем рассматривать с точки зрения независимого сжатия каждой из таблиц R. Такой подход позволит проводить операции восстановления сжатых данных лишь в тех таблицах, значения которых необходимы для выполнения запроса к БД. С целью уменьшения времени доступа к данным и в связи с ограниченным объемом аппаратно-программных ресурсов систем управления базами данных (СУБД) отношение R будем разбивать на страницы, содержащие ограниченное число записей (строк). В модели векторного представления БД отношение R будем рассматривать как множество страниц P, каждая из которых

содержит n записей. В случае не кратности числа строк в таблице R значению n, последняя страница дополняется записями до числа n путем введения недостающего числа пустых строк. В результате, страницу P можно рассматривать в виде таблицы R, состоящей из n записей.

Существует два похода к размещению данных в РБД – строчные и колоночные структуры [5]. Строчные структуры размещают данные на физическом носителе в виде множества последовательно сохраненных строк таблицы. Колоночные структуры размещают информацию на физическом носителе в виде последовательности записей для каждой колонки по отдельности [5]. В БД, ориентированных на чтение данных с высокой скоростью поиска и извлечения информации, наибольшим быстродействием обладают колоночные структуры. Помимо высокого быстродействия, колоночные структуры являются наиболее перспективными с точки зрения сжатия информации, поскольку сжатие выполняется над периодически повторяющимися однотипными данными в рамках каждой из колонок таблицы [5]. Поэтому, в качестве структуры для размещения данных в модели векторного представления БД, выбрана колоночная структура.

Отношение R представляет собой множество атрибутов (столбцов) S_l. Столбец S_l является последовательностью $S_l = \left(s_{l(n-1)}, \ldots s_{li}, \ldots s_{l1}, s_{l0}\right)$, $l \in \overline{0,(q-1)}$ длиной n, состоящей из повторяющихся значений атрибутов (сообщений) $d_{lj} \in D_l$, множества допустимых значений (домена) D_l для l-й последовательности. Поскольку каждый столбец S_l отношения R в БД представляет собой последовательность повторяющихся значений, то он моделируется с использованием модели векторного представления информации.

Модель векторного представления базы данных с колоночной структурой ставит в соответствие последовательности значений s_{li}, $s_{li} \in D_l$, $l \in \overline{0,(q-1)}$, $i \in \overline{0,(n-1)}$, атрибута S_l в отношении $R\left\{S_0 : D_0, S_1 : D_1, \ldots, S_l : D_l, \ldots, S_{q-1} : D_{q-1}\right\}$ множество значений $d_{lj} \in D_l$, $j \in \overline{0,(m_l-1)}$ домена $D_l = dom(S_l)$ и соответствующее множество двоичных векторов $Y_{lj} = \left(y_{j(n-1)}, \ldots, y_{ji}, \ldots, y_{j1}, y_{j0}\right)$, таких, что

$$y_{ji} = \begin{cases} 1, s_{li} = d_{lj} \\ 0, s_{li} \neq d_{lj} \end{cases}. \qquad (6)$$

Графическое изображение модели векторного представления таблицы БД с колоночной структурой представлено на рисунке 1.

Векторное моделирование РБД состоит из следующих этапов:

Этап 1. Разбиение РБД на множество отношений R, каждое из которых состоит из n записей (строк).

Этап 2. Разбиение отношения $R\left\{S_0 : D_0, S_1 : D_1, \ldots, S_l : D_l, \ldots, S_{q-1} : D_{q-1}\right\}$ на множество атрибутов (столбцов) S_l, содержащих значения из множества $D_l = dom(S_l)$.

Этап 3. Установление соответствия между множеством значений $d_{lj} \in D_l$ для заданного атрибута $S_l = \left(s_{l(n-1)}, \ldots, s_{li}, \ldots, s_{l1}, s_{l0}\right)$ и множеством двоичных векторов $Y_{lj} = \left(y_{j(n-1)}, \ldots, y_{ji}, \ldots, y_{j1}, y_{j0}\right)$ в соответствии с (6).

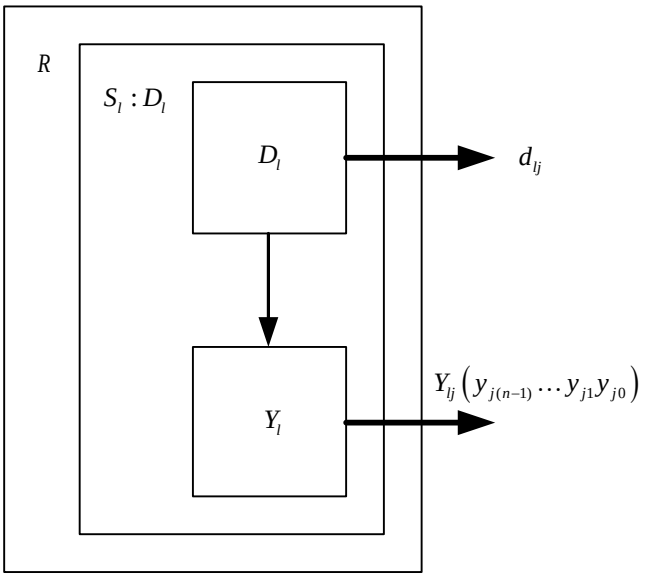

Рисунок 1 – Модель векторного представления базы данных с колоночной структурой

Для хранения отношения R, состоящего из n строк, в исходном виде необходимо выделить объем памяти

$$Ls(R) = \sum_{l=0}^{q-1} Ls(D_l) = \sum_{l=0}^{q-1} \sum_{j=0}^{m_l-1} \left(k_{d_{lj}} \cdot size(d_{lj})\right) \text{ бит.} \qquad (7)$$

Отношение R, представленное в векторном виде, будет занимать объем

$$Lv(R) = \sum_{l=0}^{q-1} Lv(D_j) = \sum_{l=0}^{q-1} \sum_{j=0}^{m_l-1} \left(size(d_{lj}) + n\right) \text{ бит.} \qquad (8)$$

ЗАКЛЮЧЕНИЕ

Одним из основных достоинств модели векторного представления данных является разделение информационной и структурной составляющей этих данных. Под информационной составляющей подразумеваются отдельные сообщения в массиве данных, а под структурной – позиции этих сообщений. Позиции, в которых находятся определенные данные, кодируются с помощью двоичного вектора. Такой подход позволяет уменьшить затраты времени при выполнении наиболее часто используемых операций при работе с РБД [4]:

- поиск данных;
- групповое изменение и удаление данных;
- выборочное извлечение данных.

Двоичные вектора являются одним из видов индексирования данных [4]. Они позволяют получать позиции, на которых размещены

запрашиваемые данные, без выполнения операций просмотра и поиска этих данных. Кроме того, наличие логических операций "И", "ИЛИ", "НЕ" над двоичными векторами в значительной степени упрощает выполнение сложных выборок данных из БД по нескольким параметрам.

Операции замены или изменения одного из атрибутов так же выполняются достаточно быстро. При выполнении такой операции изменяется лишь значение d_{lj} атрибута, а двоичный вектор Y_{lj}, указывающий позиции данного атрибута в отношении R, остается без изменений.

Добавление или удаление атрибута в определенной позиции выполняется путем изменения бита в данной позиции двоичного вектора на единичное или нулевое значение. При этом размер и само сообщение остаются неизменными.

Разделение атрибутов в модели векторного представления БД с колоночной структурой предоставляет возможность выборочного извлечения из БД определенных данных. Особенно это актуально для сложных БД, содержащих значительное число различных атрибутов. Выборочный доступ к данным позволяет в значительной степени уменьшить время поиска и извлечения информации с физического носителя.

Совокупность перечисленных преимуществ модели векторного представления БД с колоночной структурой позволяет уменьшить время доступа к данным в БД, при выполнении указанных операций. Дополнительно стоит отметить тот факт, что представленную модель можно использовать как для кодирования отдельных атрибутов, отношений и БД целиком, так и для кодирования результирующих наборов, формируемых при выполнении запросов к БД.

К недостаткам модели векторного представления данных можно отнести ограниченную скорость произвольного доступа к данным и, в некоторых случаях, избыточность представления данных. Поиск данных по определенной позиции требует перебора и проверки всех запрашиваемых атрибутов, с целью отыскания единичного бита в запрашиваемой позиции. Избыточность данных главным образом вызвана избыточностью представления двоичных векторов Y_{lj}. Для компенсации указанного недостатка необходимо использовать эффективный метод сжатия двоичной информации. В качестве такого метода предлагается использовать метод биномиального нумерационного сжатия информации [1]. Данный метод сжатия обладает достаточно высоким коэффициентом сжатия двоичных сообщений ограниченной длины при незначительных затратах времени на сжатие и восстановление данных.

SUMMARY

THE MODEL OF COLUMN ORIENTED DATABASE VECTOR REPRESENTATION

I.A. Kulik, V.V. Grinenko, S.V. Kostel, O.M. Skordina,
Sumy State University,
2, Rymsky-Korsakov Str., 40007, Sumy, Ukraine

The paper expounds the model of vector representation of column oriented relational database. The formal description and main steps of column oriented relational database vector modeling are contains in the paper. Expressions for finding memory size are considered.

Key words: *vector data representation, relational database, relation, attribute, domain, binomial enumerative compression.*

РЕЗЮМЕ

МОДЕЛЬ ВЕКТОРНОГО ПРЕДСТАВЛЕННЯ БАЗ ДАНИХ ЗІ СТОВПЧИКОВОЮ СТРУКТУРОЮ

І. А. Кулик, В. В. Гриненко, С. В. Костель, О. М. Скордіна,
Сумський державний університет,
вул. Римського-Корсакова, 2, м. Суми, 40007, Україна

В статі розглядається модель векторного представлення реляційних баз даних зі стовпчиковою структурою. Приводиться формалізований опис векторної моделі та основних етапів моделювання реляційних баз даних зі стовпчиковою структурою зберігання інформації. Пропонуються співвідношення для оцінки об'єму пам'яті, котрий займають дані до та після використання векторної моделі.

Ключові слова: *векторне моделювання даних, реляційні бази даних, відношення, атрибут, домен, біноміальне стиснення інформації.*

СПИСОК ЛИТЕРАТУРЫ

1. Кулик И.А. Биномиальное нумерационное сжатие баз данных / И. А. Кулик, С. В. Костель, Е. М. Скордина // Системи обробки інформації. –Харків: Харківський університет повітряних сил ім. Івана Кожедуба, 2013. – Вип. 3 (110), Т. 2.– С. 178.
2. Кулик И. А. Об избыточности адресно-векторного кодирования // Вестник Сумского государственного университета. – 1996. - № 1(5). - С.90-93.
3. Борисенко А. А. Биномиальное кодирование: монография / А. А. Борисенко, И. А. Кулик. – Сумы: СумГУ, 2010. – 206 с.
4. Кириллов В. В. Введение в реляционные базы данных / В. В. Кириллов, Г. Ю. Громов. – СПб.: БХВ-Петербург, 2009. – 464 с.
5. Григорьев Ю. А. Модель обработки запросов в параллельной колоночной системе баз данных / Ю. А. Григорьев, Е. Ю. Ермаков // Информатика и системы управления. - Амур: Амурский государственный университет , 2013. – №1 (31). – С. 3 - 15.

Поступила в редакцию 4 ноября 2013 г.