УДК 681.3

Е. Е. Иванов

ТАБЛИЧНО-ФОРМУЛЬНЫЙ ПРОЦЕССОР ДЛЯ ПОДГОТОВКИ ОТЧЕТОВ

Описана структура таблиц и программные средства таблично-формульного процессора для подготовки отчетов. Процессор реализован с помощью СУБД InterBase/FireBird и позволяет создавать и работать со сложной иерархической системой прикладных таблиц, содержащих фактические данные и формулы, производящие некоторую обработку информации.

Одной из задач автоматизации и информатизации управления является проектирование хранилища данных, позволяющего размещать в нем информацию, содержащуюся в прикладных таблицах, подобных табл. 1 [1, 2]. Различные статистические данные имеют вид наборов таких таблиц.

Таблица 1

Прикладная таблица со статистическими данными

Район	Ремонт комбайнов						Ремонт тракторов					
	«Нива»			«Дон»			«ДТ-75»			«Беларусь»		
	2008	2009	%	2008	2009	%	2008	2009	%	2008	2009	%
Гродненский	48	52	108	15	21	140	5	6	120	32	32	100
Лидский	36	32	88	16	14	87	13	17	130	23	29	126
Ивьевский	18	19	105	22	24	109	8	4	50	19	13	68
Итого	102	103	101	53	59	111	26	27	103	74	74	100

Видно, что структура верхнего заголовка такой прикладной таблицы может представлять собой сложную иерархическую структуру с большим числом уровней детализации (а в ряде случаев подобный вид может иметь и левый заголовок таблицы).

Часть ячеек в такой таблице содержит числовые данные из некоторой предметной области (ячейки со светлым фоном для приведенной таблицы). Другие ячейки являются результатом выполнения некоторых математических операций над содержимым ячеек с данными (например, вычисление процентной доли, суммы содержимого некоторого диапазона ячеек и пр.). При этом, если данные размещены в нескольких таблицах, то при вычислениях могут использоваться значения из ячеек разных таблиц.

Информация в горизонтальные ряды ячеек прикладных таблиц такого вида попадает, как правило, из разных и удаленных источников. В табл. 1 это районные управления сельского хозяйства. Данные, которые поступают от этих управлений, помещаются в сводную таблицу, которая формируется на уровне областного управления комитета сельского хозяйства областного исполнительного комитета. В нижней части таблицы представлена сводная информация по фактическим данным.

Примером неудачной попытки решить задачу организации хранения данных подобного рода может являться работа [3], в которой хранилище данных спроектировано в рамках технологии «файл-сервер», а прикладная логика хранимой информации внедрена в логику данных.

В настоящей работе предпринята попытка создать хранилище данных вышеописанного вида и обеспечить возможность его наполнения из удаленных источников, например, с использованием Web-интерфейса.

Информация о заголовках прикладных таблиц хранится в таблице TITLES реляционной базы данных формата InterBase/FireBird. Заголовки (верхний и левый) табл. 1 представлены в ней в виде, показанном в табл. 2. Колонка ID_TITLE является первичным ключом, а колонка ID_PARENT – внешним ключом, ссылающимся на первичный ключ этой же таблицы.

Таблица 2

TITLES с информацией о заголовках прикладной таблицы

ID_TITLE	ID_PARENT	ID_TABLE	TITLE_TYPE	TITLE_LABEL
0	0	0		ROOT_TITLES
1	0	1	U	Ремонт комбайнов
2	1	1	U	«Нива»
3	2	1	U	2008
4	2	1	U	2009
5	2	1	U	%
6	1	1	U	«Дон»
7	6	1	U	2008
8	6	1	U	2009
9	6	1	U	%
10	0	1	U	Ремонт тракторов
11	10	1	U	«Нива»
12	11	1	U	2008
13	11	1	U	2009
14	11	1	U	%
15	10	1	U	«Дон»
16	15	1	U	2008
17	15	1	U	2009
18	15	1	U	%
19	0	1	L	Гродненский район
20	0	1	L	Лидский район
21	0	1	L	Ивьевский район
22	0	1	L	**Итого**

Такой способ хранения в прямоугольных таблицах информации о структурах типа «дерево» является традиционным [4]. В табл. 2 TITLES могут храниться данные о заголовках сотен и тысяч взаимосвязанных прикладных таблиц, множество которых часто называют отчетом (или набором отчетов).

Легко заметить, что каждая ячейка тела прикладной таблицы может быть однозначно идентифицирована парой уникальных значений первичного ключа ID_TITLE (например, ячейка табл. 1 с числом 48 находится на пересечении колонки и строки, «заголовочные» ячейки которых «2008 г.» и «Гродненский район» имеют значения ключей 3 и 19). Таким образом, ячейки с данными из первой строки табл. 1 могут быть сохранены в табл. 3.

Таблица 3

TABLE_BODIES с информацией о содержимом ячеек прикладной таблицы

ID_UPPER	ID_LEFT	CELL_CONTENT
3	19	48
4	19	52
7	19	15
8	19	18
12	19	5
13	19	6
16	19	32
17	19	32

В таблице 3 TABLE_BODIES колонки ID_UPPER и ID_LEFT являются внешними ключами, ссылающимися на первичный ключ ID_TITLE таблицы TITLES. Фактически эта пара внешних ключей является координатами ячейки в теле прикладной таблицы и средством уникальной идентификации ячейки в пределах всей базы данных.

Для того чтобы с данными какой-либо ячейки прикладной таблицы можно было проводить математические операции, она должна получить статус «элемент формулы» и информация о ее координатах должна быть помещена в таблицу FORM_ELEM. Из табл. 4 видно, что все 8 ячеек первой строки прикладной табл. 1, в которых размещены данные (ячейки со светлым фоном), объявлены элементами формул. Идентификаторы этих элементов формул (значение первичного ключа ID_ELEM) лежат в диапазоне от 47 до 56. Здесь также видно, что ячейка с адресом (3,21), в которой расположено число 18, также объявлена элементом формулы.

Таблица 4

FORM_ELEM с информацией об элементах формул прикладной таблицы

ID_ELEM	ID_UPPER	ID_LEFT
47	3	19
49	4	19
51	7	19
52	8	19
53	12	19
54	13	19
55	16	19
56	17	19
58	3	21

Информация о формулах хранится в табл. 5 FORM. Здесь ID_FORM является первичным ключом таблицы (уникальный идентификатор формулы), а колонки ID_UPPER и ID_LEFT задают координаты ячейки, в которую будет помещен результат вычислений. В колонке OPERATION определяется тип формулы, в колонках OPERAND1 и OPERAND2 ее операнды.

Таблица 5

FORM с информацией о формулах прикладной таблицы

ID_FORM	ID_UPPER	ID_LEFT	OPERATION	OPERAND1	OPERAND2
1	4	19	%	47	49
2	9	19	%	51	52
3	14	19	%	53	54
4	18	19	%	55	56
5	3	22	S	47	58

Так, символ «%» говорит о том, что формула с идентификатором ID_FORM=1 является операцией вычисления процентной доли, которую составляет элемент формулы с ID_ELEM=49 от элемента с ID_ELEM=47 (нужно вычислить процентную долю, которую число 52 составляет от 48, то есть какую долю второй операнд составляет от первого). Результат вычисления (в нашем случае это 108 %) необходимо поместить в ячейку с адресом (4,19).

Для формулы с ID_FORM=5 видно, что результат вычислений нужно поместить в ячейку с адресом (3,22), а формула имеет тип «S» (суммирование). Это означает, что необходимо просуммировать содержимое ячеек от ячейки – элемента формулы с ID_ELEM=47 до ячейки – элемента формулы с ID_ELEM=58 (первый столбец данных с числами 48, 36 и 18) и результат поместить в ячейку «ИТОГО», находящуюся ниже ячеек с числами 48, 36 и 18.

Следует отметить, что элементом формулы может быть не только ячейка, в которой содержатся фактические числовые данные, но и ячейка, в которую помещается результат вычисления другой формулы. Координаты ячеек – элементов формул являются уникальными в пределах базы данных и это позволяет работать с формулами, в которых производятся вычисления с использованием содержимого любых ячеек любой при-

кладной таблицы, а результат вычислений помещать в любую ячейку любой прикладной таблицы. В настоящий момент в описываемом таблично-формульном процессоре реализованы следующие типы формул: «S» - суммирование строки или столбца (начало и конец диапазона ячеек задаются ячейками – элементами формул с идентификаторами OPERAND1 и OPERAND2), «%», «+», «*» или «–» – процентная доля, сумма, произведение или разность содержимого ячеек – элементов формул, задаваемых идентификаторами OPERAND1 и OPERAND2, «P» – ссылка на ячейку с идентификатором OPERAND1 (позволяет отображать в ячейке те данные, которые содержатся в другой ячейке – элементе формулы).

Для поддержания в базе данных ссылочной целостности используется программный код, размещенный в триггерах, которые выполняются сервером базы данных при операциях вставки, удаления или модификации записей в таблицах базы [4]. Так, например, при удалении элемента верхнего заголовка прикладной табл. 1 «Ремонт тракторов» в соответствующем триггере будет выполнен SQL-запрос, удаляющий все элементы-потомки, из которых есть ссылки на этот удаляемый элемент. Это приведет, в свою очередь, к каскадному процессу выполнения таких же триггеров при удалении элементов «ДТ-75» и «Беларусь» и в конечном итоге из таблицы TITLES будут удалены все записи, соответствующие элементам заголовка из поддерева «Ремонт тракторов».

При этом также при работе триггера с помощью отдельного SQL-запроса из таблицы TABLE_BODIES будут удалены все записи, внешние ключи ID_UPPER которых ссылаются на удаляемые элементы верхнего заголовка прикладной таблицы. То есть, если мы захотим из прикладной табл. 1 удалить поддерево заголовка «Ремонт тракторов», то также из базы данных будет удалена вся информация о содержимом, размещенном в 12 ячейках тела прикладной таблицы со светлым фоном, которые находятся под элементом «Ремонт тракторов».

Если при операции удаления будут затронуты ячейки прикладной таблицы, которые являются либо элементами формул, либо ячейками, в которые помещается результат вычисления формулы, то удаление соответствующего объекта не производится. В этом случае возбуждается исключение с сообщением, оповещающим о том, что предварительно для выполнения данной операции необходимо удалить расположенные по соответствующим адресам формулы или элементы формул.

При изменении значения поля первичного ключа ID_TITLE в таблице TITLES (это диктуется логикой системы при операции вставки, например, в структуру верхнего заголовка прикладной таблицы ячейки «Ремонт газонокосилок» между ячейками «Ремонт комбайнов» и «Ремонт тракторов») в соответствующем триггере выполняется SQL-запрос, корректирующий значения внешних ключей из других таблиц, ссылающихся на эту запись с изменяемым значением ключа.

Для модификации структуры прикладных таблиц (работа по изменению структуры элементов верхнего и левого заголовков прикладных таблиц), а также работы с содержимым тела прикладных таблиц (данные в ячейках, элементы формул и формулы) в базе данных создано 36 хранимых процедур. Эти процедуры реализуют все возможности для работы с системой как для случая использования Web-интерфейса, так и в случае использования доступа к хранилищу данных через локальную вычислительную сеть. Так, например, работник, обладающий правами администратора в данной системе, может дистанционно посредством любого интернет-браузера создавать прикладные таблицы, менять их структуру, определять ячейки – элементы формул, создавать формулы и работать с фактическими данными, хранящимися в ячейках прикладных таблиц.

Для ограничения доступа к отдельным частям прикладных таблиц в рассматриваемом процессоре используется система фильтров. Так, например, работник Лидского райисполкома, подключившийся к процессору через Web-интерфейс с соответствующими идентификационными данными, получит от сервера данные по прикладной табл. 1 в соответствии с фильтрами «верхний заголовок-11111111111» и «левый заголовок-0100». Это будет означать, что в своем интернет-браузере он увидит эту таблицу так, как если бы она имела полноценный верхний заголовок и только одну строку с элементом левого заголовка «Лидский район». После редактирования данных этой строки и использования кнопки «Сохранить данные» информация передается на сервер, где и сохраняется в соответствующих записях таблицы TABLE_BODIES. Другой работник, например, сотрудник райисполкома в Ивьевском районе, может иметь доступ к данным прикладной таблицы с фильтрами «верхний заголовок-00000011111» и «левый заголовок-0010». В этом случае ему будет видна и доступна для редактирования таблица со строкой «Ивьевский район» и набором колонок, соответствующим элементу верхнего заголовка «Ремонт тракторов». И, наконец, работник, например, областного управления будет иметь доступ к данным системы с фильтрами «верхний заголовок-11111111111» и «левый заголовок-1111». То есть ему будет видна вся прикладная табл. 1.

Таким образом, в описанном таблично-формульном процессоре для подготовки отчетов реализован принцип, в соответствии с которым удаленный клиент любого ранга может работать с системой при наличии у него только стандартного интернет-браузера. Весь программный код, реализующий работоспособность системы, выполнен в виде триггеров и хранимых процедур, находящихся в базе данных сервера.

Литература

1. *Кастеллани, К.* Автоматизация решения задач управления / К. Кастеллани. – М. : Мир, 1982. – 472 с.

2. *Железко, Б. А.* Теория и практика построения информационно-аналитических систем поддержки принятия решений / Б. А. Железко, А. Н. Шрозевич. – Минск : Армита-Маркетинг, Менеджмент, 1999. – 144 с.

3. *Бабарика, Н. Н.* Концепция и реализация многофункционального редактора отчетов / Н. Н. Бабарика, А. В. Никитин // Весн. ГрДУ. – Сер. 2. – 2004. – № 1. – С. 100- 105.

4. *Ковязин, А. Н.* Мир InterBase / А. Н. Ковязин, С. М. Востриков. – М. : КУДИЦ-ОБРАЗ, 2003. – 492 с.

Иванов Евгений Евгеньевич, доцент кафедры теоретической физики физико-технического факультета Гродненского государственного университета имени Янки Купалы, кандидат физико-математических наук, доцент, e.ivanov@grsu.by

УДК 519.1

Я. А. Исаченко

О НЕКОТОРЫХ ГИПЕРГРАНЯХ МНОГОГРАННИКА ЦИКЛИЧЕСКИХ ПЕРЕСТАНОВОК

Рассматривается многогранник циклических перестановок. Вводится понятие наследуемых граней. Исследуются гиперграни многогранника циклических перестановок, наследуемые у перестановочного многогранника.

Введение

Пусть $A=\{a_1,a_2,\ldots,a_n\}$, $B=\{b_1,b_2,\ldots,b_n\}$ – множества действительных чисел, удовлетворяющих неравенствам $a_1<a_2<\ldots<a_n$, $b_1<b_2<\ldots<b_n$. Многочисленный класс задач дискретной математики составляют экстремальные задачи на множествах перестановок:

$$\sum_{i=1}^{n} a_{\pi(i)} b_i \to \min\ (\max),\ \pi \in P_n. \tag{1}$$

Здесь P_n – некоторое множество перестановок из n первых натуральных чисел.

Если удается описать выпуклую оболочку $M(P_n)=conv\{a(\pi)=(a_{\pi(1)},a_{\pi(2)},\ldots,a_{\pi(n)} \mid \pi\in P_n\}$, то задача (1) сводится к задаче линейного программирования [1]. Частичное описание выпуклой оболочки дает возможность построения релаксационной задачи или последовательности релаксационных задач в виде задач линейного программирования. Поэтому вызывают интерес многогранники, задаваемые множествами перестановок. Для множества всех перестановок S_n n первых натуральных чисел N известно [2], что размерность перестановочного многогранника $dimM(S_n) = n-1$, любая точка $a(\pi)$, $\pi\in S_n$ является его вершиной, а задающая его полная неприводимая система имеет вид

$$\sum_{i=1}^{n} x_i = \sum_{i=1}^{n} a_i,$$

$$\sum_{i\in\omega} x_i \ge \sum_{j=1}^{|\omega|} a_j,\ \forall \omega \subset N,\ |\omega| \le n-1. \tag{2}$$

Пусть $\pi\in S_n$. Если $\pi(i)=i$, то говорим, что число i образует в перестановке цикл длины 1. Будем говорить, что множество чисел $\{i_1,i_2,\ldots,i_k\}\subseteq N$, $2\le k\le n$ образуют в перестановке π цикл длины k, если: 1) множества $\{i_1,i_2,\ldots,i_k\}$ и $\{\pi(i_1),\pi(i_2),\ldots,\pi(i_k)\}$ совпадают; 2) $\pi^s(i_1)\ne i_1$ для любого s, $1\le s\le k-1$; 3) $\pi^k(i_1)=i_1$.

Заметим, что свойство 3) будет иметь место для любого числа, входящего в цикл.