

ФОРМАЛИЗАЦИЯ ПРОЦЕССА ОБРАБОТКИ ДАННЫХ В ИНФОРМАЦИОННЫХ БУХГАЛТЕРСКИХ ТЕХНОЛОГИЯХ С ПОМОЩЬЮ АППАРАТА ТЕОРЕТИКО-МНОЖЕСТВЕННЫХ ПРЕДСТАВЛЕНИЙ

В.И. Лойко, Н.П. Орлянская

Кубанский государственный технический университет, г. Краснодар

Адыгейский государственный университет, г. Майкоп

В статье рассмотрены методы формализации программного обеспечения бухгалтерских подсистем с помощью аппарата теоретико-множественных представлений. Это позволяет формализовать алгоритм задачи и оптимизировать процесс обработки данных в информационных бухгалтерских технологиях.

В известных программных продуктах как “1С-бухгалтерия” “БЭСТ” “Парус” “Комфорт” успешно используются возможности реляционной аналитики. Правильнее сказать – эти программные средства совершенно справедливо завоевали пользовательские симпатии и приобрели популярность благодаря реализации инструментария реляционной модели СУБД и применения в ней реляционной алгебры. Описание способов реализации традиционных теоретико-множественных операций в новейших информационных бухгалтерских технологиях говорит об актуальности и совершенстве этого математического аппарата, кроме того, такое описание формализует вопрос проектирования информационных бухгалтерских систем.

Реляционная модель данных характеризуется следующими компонентами [6]:

- информационной конструкцией - отношением с двух-уровневой структурой,
- допустимыми операциями,
- ограничениями - функциональными зависимостями между атрибутами отношения.

Каждому классу объектов P материального мира ставится в соответствие некоторое множество атрибутов, например

A_1, A_2, \dots, A_n .

Отдельный объект класса P описывается строкой величин (a_1, a_2, \dots, a_n) , где a_i - значение атрибута M .

Строка (a_1, a_2, \dots, a_n) , называется кортежем. Всему классу объектов соответствует множество кортежей, называемое отношением.

Обозначим отношение, описывающее класс объектов P , также через R . Выражение

$R(A_1, A_2, \dots, A_n)$ называется схемой отношения R .

Для каждого компонента кортежа должна быть указана ее связь с соответствующим атрибутом. В реляционной модели данных для обеспечения этой связи порядок компонентов кортежа совпадает с порядком следования атрибутов в схеме отношения. Каждое отношение представляет состояние класса объектов в некоторый момент времени. Множество значений отношения можно представить в виде таблицы, в которой соблюдаются следующие соответствия:

- название таблицы и перечень названий граф соответствуют схеме отношения,
- строке таблицы соответствует кортеж отношения,
- все строки таблицы (и соответственно все кортежи) различны,
- порядок строк и столбцов произвольный (в частности, реляционная модель данных не предполагает специальную сортировку).

Реляционная база данных представляет собой множество отношений.

Схема реляционной БД содержит следующие компоненты

$S(\text{rel}) = \langle A, R, \text{Dom}, \text{Rel}, V(s) \rangle$,

где A - множество имен атрибутов,

R - множество имен отношений,

Dom - вхождение атрибутов в домены,

Rel - вхождение атрибутов в отношения,

$V(s)$ - множество ограничений (в том числе функциональных зависимостей).

Опишем процессы обработки отношений в бухгалтерской подсистеме. Отношение реляционной БД может быть описано в терминах теории множеств.

Множество отношений и операций над ними образует реляционную алгебру. Каждая операция реляционной алгебры использует одно или два отношения в качестве операндов и продуцирует в результате некоторое новое отношение.

Обозначим исходные отношения $R1$ и $R2$, а результирующее через T , тогда:

Объединение $T=U(R1,R2)$ двух (совместимых по объединению) отношений $R1$ и $R2$ содержит строки, присутствующие либо в отношении $R1$, либо в $R2$ либо принадлежащие им обоим.

Операция объединения может быть применена в информационных бухгалтерских технологиях для объединения одинаковых справочников, имеющих как различные, так и повторяющиеся записи в двух или нескольких подразделениях, например, клиентов для подготовки счета-фактуры и платежного поручения и др. отчетных формах.

Например, в языке SQL:

```
SELECT КЛИЕНТ FROM S
UNION
SELECT КЛИЕНТ FROM SP;
```

В результате справочники будут объединены, причем повтора реквизитов одного и того же клиента не будет.

Удобно использовать эту операцию для слияния в один Журналы хозяйственных операций с разных компьютеров, причем если нет четкого разделения труда по участкам учета, ситуация повтора хозяйственной операции исключается автоматически.

Пересечение $T=I(R1, R2)$ двух (совместимых по объединению) отношений $R1$ и $R2$ содержит строки, присутствующие в отношениях $R1$ и $R2$ одновременно.

Операция пересечение может быть применена в информационных бухгалтерских технологиях для выяснения суммы задолженности сотрудника по различным счетам. Один и тот же сотрудник может проходить и по счету 70 „Зарплата“, и по счету 71 „Расчеты с подотчетными лицами“, и еще по ряду других счетов. При увольнении важно знать общую задолженность на сотрудника по всем счетам.

Например. Реализация этой операции в языке SQL:

```
SELECT Табельный_N FROM S
WHERE EXISTS
SELECT Табельный FROM SP
WHERE SP.Табельный_N=S.Табельный_N);
```

Разность $T=M(R1, R2)$ между двумя (совместимыми по объединению) отношениями $R1$ и $R2$ содержит те строки из $R1$, которые отсутствуют в $R2$. [30,31,32]

Операция разности применима в анализе бухгалтерской информации для расчленения некоторых данных, допустим необходимо проанализировать работу автотранспорта фирмы по доставке товара со склада, тогда самовывоз необходимо исключить, или для подготовки формы по оплаченным но не отгруженным товарам со склада например, используя язык SQL:

```
SELECT ТОВАР FROM S
WHERE NOT EXISTS
(SELECT ТОВАР FROM SP
WHERE SP.ПОСТАВЩИК=
S. ПОСТАВЩИК);
```

Декартовым произведением $T=D(R1,R2)$ двух отношений $R1$ и $R2$ называется множество всех кортежей t , таких, что t является конкатенацией некоторого кортежа r_1 , принадлежащего $R1$, и какого-либо кортежа r_2 , принадлежащего $R2$.

Декартово произведение применимо в бухгалтерских технологиях для развернутых форм с использованием дополнительных данных, например по износу основных средств.

Пример в языке SQL.

```
SELECT S.*,SP.*  
FROM S, SP;
```

Фундаментализация программного обеспечения бухгалтерских подсистем и их формализация на уровне теоретико-множественных представлений открывает новые горизонты в информационных технологиях, позволяет найти пути оптимизации вычислительного процесса.

Литература

1. Мамиконов А.Г. Проектирование АСУ. – М.: Высшая школа, 1987.
2. Цикритзис Д., Лоховски Ф. Модели данных. – М.: Финансы и статистика, 1985.
3. Озкарахан Э. Машины баз данных и управление базами данных. – М.: МИР, 1989.
4. Бойко В.В., Савинков В.М. Проектирование баз данных информационных систем. – М.: Финансы и статистика, 1989.
5. Автоматизированные информационные технологии в экономике Сборник научных трудов /КГАУ, Краснодар 1998.
6. Мишин А.И. Теория экономических информационных систем. – М.: Финансы и статистика, 1999.
7. Дейт К. Руководство по реляционной СУБД ДВ2. – М.: Финансы и статистика, 1988 г.
8. Ревунков Г.И., Самохвалов Э.Н., Чистов В.В. Базы и банки данных и знаний. – М.: Высшая школа, 1992.

Formalization of the process of handling data with assistance of set-theoretic presentations in information technology for account departments

V.I. Loiko, N.P. Orlyanskaya

The methods of formalization software for account departments are considered, which make optimal the process handling data.