

Полтавцев А.А. Проблемы хранения структурированной информации типа «Перечень материалов» в реляционной базе данных. // Проблемы информатики в образовании, управлении, экономике и технике: Сб. статей XVI Междунар. научно-техн. конф. – Пенза: ПДЗ, 2016. – С. 199-202.

УДК 004.652.4

## ПРОБЛЕМЫ ХРАНЕНИЯ СТРУКТУРИРОВАННОЙ ИНФОРМАЦИИ ТИПА «ПЕРЕЧЕНЬ МАТЕРИАЛОВ» В РЕЛЯЦИОННОЙ БАЗЕ ДАННЫХ

А.А. Полтавцев

## STRUCTURED INFORMATION OF TYPE «BILL OF MATERIALS» STORAGE PROBLEMS IN THE RELATIONAL DATABASE

A.A. Poltavtsev

**Аннотация.** Рассматриваются различные варианты представления структурированной информации, в частности, структур типа «перечень материалов» в базах данных, а именно структуры таблиц и реализация некоторых операций по работе с ними через SQL-запросы к этим таблицам. Речь идет о структуре данных и об операциях обработки. Вопросы написания компонент здесь не рассматриваются и будут освещены в дальнейших работах автора.

**Ключевые слова:** базы данных, реляционная модель данных, реляционная схема, иерархия.

**Abstract.** In article various variants of in databases the structured information representation, in particular structures of type «bill of materials» are considered, namely structures of tables and realization of some operations on job with them through SQL clauses to these tables. It is a question of structure of data and about operations of processing. Questions of a writing a component here are not examined.

**Keywords:** databases, relational data model, relational scheme, hierarchy.

Архитектура реляционных баз данных ориентирована на хранение внутри таблиц БД информации о сущностях информационной системы и связях между ними. Каждая из записей таблицы содержит информацию об одном экземпляре. Организация хранения информации о независимых друг от друга экземплярах сущностей (т.е. так называемых «плоских» данных) не вызывает никаких затруднений. Однако, наряду с «плоскими» данными, при построении даже простых информационных систем приходится хранить в БД и информацию о «вложенных» друг в друга сущностях, например иерархические данные, с иерархией «IsA». Организация хранения такой структурированной информации в реляционных БД не проста [1].

Более того, в реальной жизни иерархия имеет, как правило, многочисленные подусловия, что еще более усложняет проблему. Например, во многих организациях, занимающихся выпуском изделий, используется структура «перечень материалов» – список компонентов (т.е. частей), необходимых для сборки изделия. Задача этой структуры идентифицировать:

- какие компоненты (детали, составные части) используются в различных продуктах (изделиях),

- какие продукты выпускаются и их состав.

В этих случаях говорят о типах компонент и продуктах. Компонента одного и того же типа (например, мотор) может входить в состав множества других компонент или продуктов (иерархия «Has A») [2].

Ситуация еще более усложняется, когда приходится учитывать:

- количественный аспект;
- порядок сборки.

Каждая КОМПОНЕНТА или ПРОДУКТ может состоять из одной и более СОСТАВЛЯЮЩИХ соответствующего количества и порядка сборки. И наоборот, каждая КОМПОНЕНТА или ПРОДУКТ может использоваться в качестве одной и более СОСТАВЛЯЮЩИХ, каждая из которых должна входить в список компонент для другой КОМПОНЕНТЫ или ПРОДУКТА.

Рассмотрим модельную задачу на примере детского конструктора. Сборка из готовых блоков. Пусть имеется следующий перечень материалов (комплектующих):

<b>количество</b>	<b>размер</b>	<b>цвет</b>
32	1×1	желтый
4	1×3	черный
5	1×4	голубой

Соберем из них 5 сборок: буквы E, F, H, I, и L, так, что:

<b>количество</b>	<b>размер</b>	<b>цвет</b>
E: 4	1×1	желтый
1	1×3	черный
2	1×4	голубой
F: 5	1×1	желтый
1	1×3	черный
1	1×4	голубой
H: 12	1×1	желтый
1	1×4	голубой
I: 5	1×1	желтый
2	1×3	черный
L: 6	1×1	желтый
1	1×4	голубой

В принципе процесс сборки можно продолжить и собрать из букв H и I новую сборку – слово HI и т.д. В реальной жизни процесс сборки является не только многокомпонентным, но и многоэтапным.

Хранение в реляционной базе данных даже простых иерархий (организационная структура предприятия, план счетов, статьи сметы) является проблемой и решалось много лет. Предложены различные техники и реляционные схемы, от рекурсивной ссылки Дейта до встроенного рекурсивного типа данных. К сожалению, для описанной проблемы они не могут быть применены напрямую, так как не учитывают ни количественные характеристики, ни порядок сборки компонентов.

Для хранения данных примера можно предложить реляционные отношения с различными схемами. Прежде всего, отношение с хранением каждого компонента

в отдельной строке. Тогда, если ввести таблицу E (таблица компонентов для сборки буквы E), получим, например:

partId	brick	color
1	1×1	yellow
2	1×1	yellow
3	1×1	yellow
4	1×1	yellow
5	1×3	Black
6	1×4	Blue
7	1×4	Blue

Однако для перечня материалов такая схема является неудовлетворительной. Даже вопрос о том, «сколько блоков каждого цвета использованы для сборки E», требует написания нетривиального скрипта. Данный запрос может быть легко выполнен, если добавить в отношение атрибут «количество»:

partId	brick	color	quantity
1	1×1	yellow	4
2	1×3	Black	1
3	1×4	Blue	2

Однако в таком решении потеряна информация о конкретных составляющих сборки, что может оказаться критическим при решении вопросов технического или гарантийного обслуживания продуктов (сборок).

Таки образом, данная проблема подлежит дальнейшему исследованию для поиска адекватного решения.

#### Библиографический список

1. Полтавцева М.А. Хранение сложных структур данных в реляционной базе данных. Тверь: Изд-во ТвГТУ, 2013. 184 с.

2. Halpin T. Information Modeling and Relational Databases: from conceptual analysis to logical design. Morgan-Kaufmann, San Francisco, 2001.

**Полтавцев Анатолий Алексеевич**

Тверской государственный

технический университет,

г. Тверь, Россия

E-mail: aapolt@gmail.com

**Poltavtsev A.A.**

Tver State Technical University,

Tver, Russia