

УДК 658.5:629.78

Doi: 10.31772/2712-8970-2021-22-4-589-599

**Для цитирования:** Картамышев А. С., Черныш Б. А., Мурыгин А. В. Формирование многомерных данных в информационной финансово-экономической системе на предприятии госкорпорации «Роскосмос» // Сибирский аэрокосмический журнал. 2021. Т. 22, № 4. С. 589–599. Doi: 10.31772/2712-8970-2021-22-4-589-599.

**For citation:** Kartamyshev A. S., Chernysh B. A., Murygin A. V. Method for forming multi-dimensional data in the information financial and economic system at the enterprise of state space corporation “Roscosmos”. *Siberian Aerospace Journal*. 2021, Vol. 22, No. 4, P. 589–599. Doi: 10.31772/2712-8970-2021-22-4-589-599.

## **Формирование многомерных данных в информационной финансово-экономической системе на предприятии госкорпорации «Роскосмос»**

А. С. Картамышев<sup>1\*</sup>, Б. А. Черныш<sup>2</sup>, А. В. Мурыгин<sup>2</sup>

<sup>1</sup>АО «Информационные спутниковые системы» имени академика М. Ф. Решетнева)  
Российская Федерация, 662972, г. Железногорск Красноярского края, ул. Ленина, 52

<sup>2</sup>Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева  
Российская Федерация, 660037, г. Красноярск, просп. им. газ. «Красноярский рабочий», 31

\*E-mail: kartam@iss-reshetnev.ru

*Построение детализированного учета, позволяющего генерировать сложную, аналитическую отчетность, является непреложным требованием современной финансовой системы. Для предприятий госкорпорации «Роскосмос» с позаказным и мелкосерийным производством, работающих в условиях 275 ФЗ, отличающихся большим количеством НИОКР и высокой степенью неопределенности в процессе создания продукции, эта задача является системообразующей. Отчетность должна содержать согласованные непротиворечивые данные на любом участке управленческого и бухгалтерского учета на любой момент времени. Наряду с этим, система должна обеспечивать гибкость, надежность и быстрдействие, присущие транзакционным базам данных. Для построения информационной поддержки, удовлетворяющей указанным условиям, требуется либо разделять OLTP и OLAP схемы данных, либо применять специализированные решения, основанные на использовании структур и методик, оптимизированных для выполнения OLAP операций в традиционных РСУБД. В данной статье рассматривается подход к формированию многомерных данных в автоматизированной системе управления финансово-экономическими задачами как эффективная альтернатива сложным и дорогостоящим BI-решениям. В отличие от многих коммерческих решений, описываемая система не хранит избыточные данные (например, регистры оперативного учета в платформе «1С: Предприятие»), необходимые для построения аналитического учета. Лежащие в ее основе структуры данных и методы их обработки позволяют осуществлять все виды учета и иметь мощные инструменты построения аналитической отчетности. В статье предлагаются алгоритмы работы системы на примере построения простых OLAP кубов, применяемых в реальных задачах автоматизации финансово-экономической деятельности в АО «ИСС» для одной из подсистем «Покупки». Проведена формализация этих задач, рассмотрен математический аппарат построения многомерных моделей данных на основе информации из фиксированного набора нормализованных таблиц реляционной БД. Представлены примеры SQL запросов и выходных данных. Обобщены преимущества применения системы в оперативном, управленческом и бухгалтерском учете на предприятии, повышающие ее эксплуатационную эффективность.*

*Ключевые слова:* база данных, OLTP, OLAP, аналитическая обработка, многомерное представление, реляционная схема, нормализация, нормальная форма, система автоматизации.

## Method for forming multi-dimensional data in the information financial and economic system at the enterprise of state space corporation “Roscosmos”

A. S. Kartamyshev<sup>1\*</sup>, B. A. Chernysh<sup>2</sup>, A. V. Murygin<sup>2</sup>

<sup>1</sup>JSC Academician M. F. Reshetnev Information Satellite Systems  
52, Lenin St., Zheleznogorsk, Krasnoyarsk region, 662972, Russian Federation

<sup>2</sup>Reshetnev Siberian State University of Science and Technology  
31, Krasnoyarskii rabochii prospekt, Krasnoyarsk, 660037, Russian Federation

\*E-mail: kartam@iss-reshetnev.ru

*The construction of detailed accounting that allows to generate complex analytical reporting is an indispensable requirement of a modern financial system. For enterprises of the State Space Corporation “Roskosmos” with custom-made and small-scale production, operating in the conditions of Federal Law No. 275, characterized by a large amount of R&D and a high degree of uncertainty in the process of creating products, this task is a system-forming one. The reporting should contain consistent data in any area of management and accounting at any given time. Along with this, system must provide the flexibility, reliability and performance inherent in transactional databases. To build information support that satisfies the specified conditions, it is required either to separate OLTP and OLAP data schemas, or to apply specialized solutions based on the use of structures and techniques optimized for performing OLAP operations in traditional RDBMSs. This article discusses an approach to the formation of multidimensional data in an automated management system for economic tasks, as an effective alternative to complex and expensive BI solutions. Unlike many commercial systems, the ASU FEZ does not store redundant data (for example, operational accounting registers in the IC: Enterprise platform) required to build analytical accounting. The underlying data structures and methods of their processing allow for all types of accounting and have powerful tools for constructing analytical reporting. The article proposes algorithms for the operation of the system using the example of building simple OLAP cubes used in real tasks of automating financial and economic activities in ISS JSC for one of the Purchase subsystems. The formalization of these problems is carried out, the mathematical apparatus for constructing multidimensional data models based on information from a fixed set of normalized tables of a relational database is considered. Examples of SQL queries and outputs are provided. The advantages of using the system in operational, management and accounting at an enterprise, which increase its operational efficiency, are summarized.*

*Keywords: database, OLTP, OLAP, analytical processing, multidimensional representation, relational schema, normalization, normal form, automation system.*

### Введение

Одним из основных требований, предъявляемых к финансово-экономическим системам, является возможность оперативного получения актуальной аналитической информации, необходимой для принятия оперативных управленческих решений и генерации всеохватывающей детализированной отчетности. Особенно актуальной эта задача является для предприятий госкорпорации «Роскосмос», работающих в условиях гособоронзаказа и обязанных вести в своей хозяйственной деятельности раздельный учет затрат по различным проектам. Информационная поддержка бухгалтерского и управленческого учета построена на правилах хранения и обработки данных документов-первоисточников, регистрируемых в системах учета, их связях и процессах формирования новых информационных объектов учета. Данные на разных участках учета есть не что иное, как производные от первичных документов или множества первичной информации. В процессе передела и потребления данных на определенных участках учета ин-

формационные связи между различными производными от одного основания могут теряться. Информационные объекты начинают «жить своей жизнью» и порождать новые производные. При информационных разрывах теряется возможность дифференцирования текущей производной до первоисточника данных (основания) и анализа связанных с ним данных в рамках технологических и экономических процессов. Математическое решение подобных задач, приводящее к логически выстроенной цепочке операций и возможности автоматизации учета, является предметом исследований многих ученых и практиков, работающих над созданием систем информационной поддержки финансово-экономических задач в управлении предприятиями.

### **Анализ исследований**

Ведущее место в решении данного класса задач занимают OLAP-технологии [1] (OLAP – online analytical processing, оперативная аналитическая обработка). Системы, реализующие OLAP, должны отвечать так называемым принципам FASMI [2]: *Fast* (быстрый отклик системы), *Analysis* (полнофункциональный анализ), *Shared* (многопользовательский доступ к данным), *Multidimensional* (многомерное концептуальное представление данных), *Information* (получение информации в нужном объеме там, где она необходима). Как видно, исходя из данных принципов, требования к системе построения отчетности, которую действительно можно назвать OLAP-системой, весьма жесткие. За секунды отчеты должны строиться по миллионам записей, столь же быстро раскрываться по интересующим пользователя аналитикам, фильтроваться и перегруппировываться [3].

Наиболее распространенным подходом в реализации OLAP-систем является разделение данных между двумя базами: транзакционной (OLTP – online transaction processing) и аналитической (OLAP). При этом OLTP-база оптимизируется под быстрое внесение данных, а OLAP – под быстрое построение сложной отчетности. Фактически OLAP-база обычно строится по специальной архитектуре и содержит предварительно просчитанные агрегатные данные, что и обеспечивает высокую скорость выполнения запросов. Платой за это становится необходимость синхронизировать OLTP и OLAP. Так как подобный процесс, как правило, периодичен, то между появлением данных в оперативной базе и аналитическими данными имеется запаздывание. Данный подход реализован практически во всех современных BI-системах (business intelligence – сбор и анализ бизнес-информации): Microsoft Analysis Services, Oracle OLAP, IBM Informix, Arbo Essbase и др.

Другим подходом в решении задач оперативной аналитической обработки данных является встраивание OLAP-инструментария непосредственно в OLTP-хранилище. Это достигается как использованием специальных элементов данных (например, регистры оперативного учета в платформе «1С: Предприятие» [4]), так и оптимальной (с точки зрения оптимизации выполнения OLAP-запросов) структуризацией схемы данных. Данный подход лег в основу разработанной на предприятии системы управления финансово-экономическими задачами (АСУ ФЭЗ) [5] как эффективная альтернатива сложным и дорогостоящим BI-системам.

### **Описание подхода**

Структуры и алгоритмы, лежащие в основе АСУ ФЭЗ, позволяют эффективно сочетать гибкость и быстродействие транзакционной базы данных (БД), с одной стороны, и возможности построения сложной отчетности за приемлемое время – с другой. Далее в статье приведен обзор технической реализации хранилища данных на примере подсистемы «Покупки», отвечающей за учет товаров, работ и услуг, которые предприятие приобретает для создания собственных объектов продажи.

Ядро модуля «Покупки» составляет 5 таблиц, построенных в соответствии с правилами нормальных форм [6; 7], и представлено на рис. 1:

- POK\_PRIH – документы прихода товарно-материальных ценностей, работ, услуг;
- POK\_OSN – документы-основания для учета НДС (входящие счета-фактуры);
- PO\_SOGL – документы-основания для платежа (акцепт-согласие ответственных сотрудников на оплату товарно-материальных ценностей, работ, услуг);
- POK\_OPL – платежные документы;
- POK\_SW – таблица связей, содержит ссылки в виде внешних ключей на первичные ключи записей из перечисленных выше таблиц (ID\_PR – на таблицу POK\_PRIH, ID\_OSN – на таблицу POK\_OSN, ID\_TS – на таблицу POK\_SOGL, ID\_OPL – на таблицу POK\_OPL).

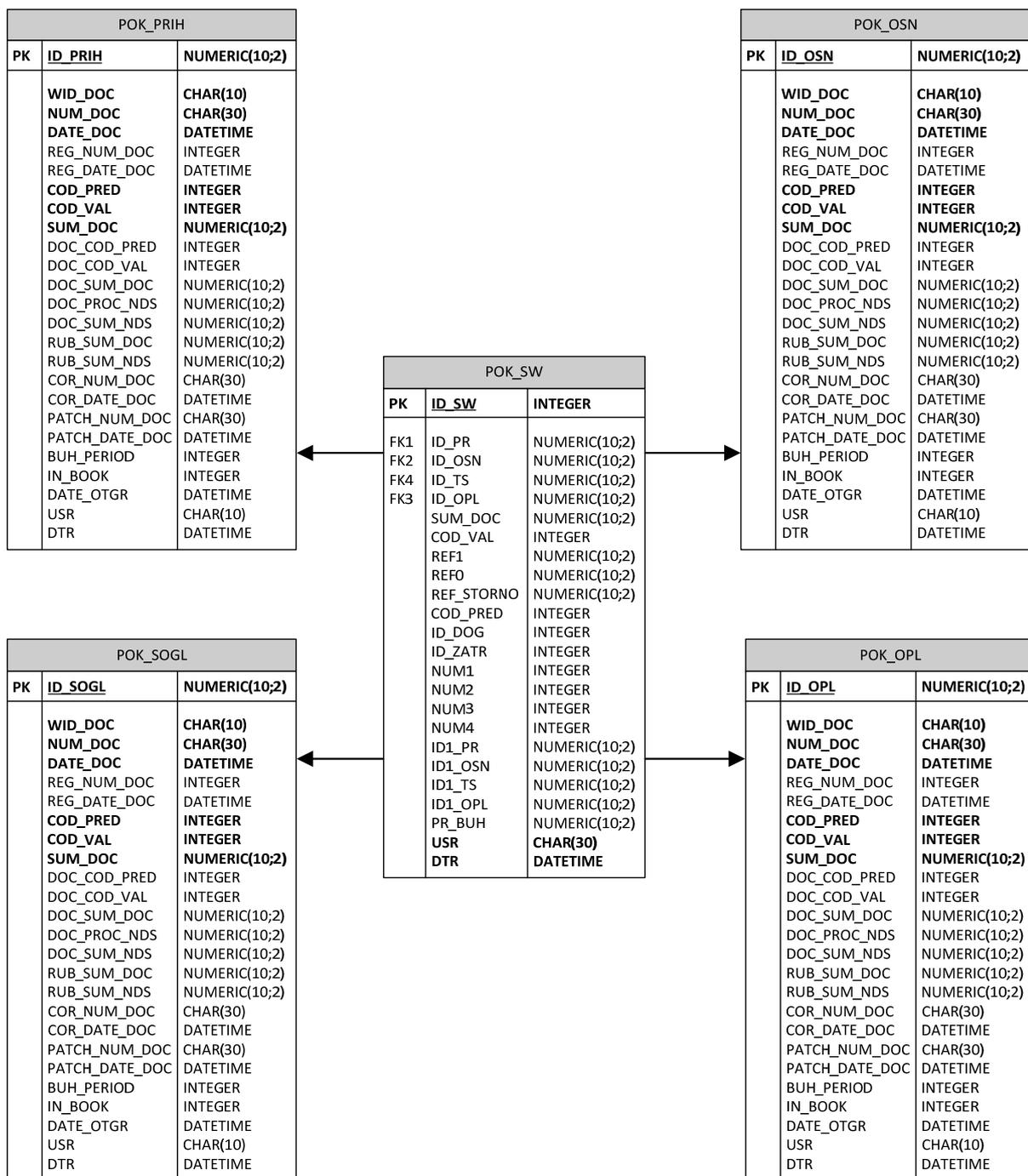


Рис. 1. Схема данных модуля «Покупки»

Fig. 1. Module "Purchase" data schema

Предлагаемый подход к структурированию данных позволяет выполнять широкий спектр задач аналитической обработки без предварительной подготовки данных или использования промежуточных представлений. Для наглядного описания проведем их формализацию [8], затем рассмотрим несколько реальных примеров таких задач, решаемых в рамках подсистемы «Покупки». Итак, пусть задана схема базы данных  $\mathfrak{R} = \{R_1, R_2, \dots, R_k\}$ , полученная в результате нормализации отношений [6; 7]. Отношения  $R_i$  определены на множестве атрибутов  $U = \{A_1, A_2, \dots, A_n\}$ . Пусть  $[R_i]$  – схема отношения, множество атрибутов, на которых определено отношение  $R_i$ . Предположим, что схема  $\mathfrak{R}$  является редуцированной [7], т. е. не существует двух отношений таких, что  $[R_i] \subseteq [R_j]$ , при  $i \neq j$ . Кортеж  $t[X]$  – совокупность значений атрибутов  $A_j \in X \subseteq [R_i]$ , заданных в кортеже  $t \in R_i$ . Неопределенное значение  $NULL$  атрибута  $A_j$  в кортеже  $t$ :  $t[A_j] = NULL$  не равно любому другому значению, в том числе другому неопределенному значению.

Многомерное представление будем задавать в виде совокупности размерностей  $\{D_1, D_2, \dots, D_d\}$ , где  $D_l$  – множество расширенных имен атрибутов:  $R_i A_j, A_\varphi \in [R_i]$ ;  $M$  – множество мер, также заданных в виде расширенных имен атрибутов. Значения  $D_l$  являются значениями координат гиперкуба, значения  $M$  будут располагаться в рабочей области гиперкуба. Для каждой размерности задается ограничение в виде логической формулы  $F_l$ .

**Пример 1.** Необходимо найти данные оплат (любых перечислений денежных средств) предприятию с кодом 12345 по проекту 111 и договорам поставки 222, 223, где были уплачены денежные средства с 01.01.2021 по 28.02.2021 по статьям бюджетного планирования (СБП) 333, 334 и получены товар/работы/услуги в период с 01.01.2021 по 31.01.2021.

С целью упрощения постановки и реализации, в этом и последующих примерах не используемые в выборке атрибуты опускаются. Таким образом, в рамках описанной схемы БД имеем следующее подмножество атрибутов:  $A_1$  – идентификатор связи,  $A_2$  – идентификатор записи о поступлении товара/работ/услуг,  $A_3$  – идентификатор записи платежа,  $A_4$  – код предприятия,  $A_5$  – идентификатор проекта,  $A_6$  – идентификатор договора поставки,  $A_7$  – СБП,  $A_8$  – дата приходной накладной,  $A_9$  – дата платежа,  $A_{10}$  – сумма платежа. Здесь существуют следующие функциональные зависимости:  $DEP = \{A_1 \rightarrow A_2 A_3 A_4 A_5 A_6 A_7, A_2 \rightarrow A_8, A_3 \rightarrow A_9 A_{10}\}$ . На основании данного подмножества получаем упрощенную схему отношений: Связи документов =  $R_1(A_1, A_2, A_3, A_4, A_5, A_6, A_7)$ , Поступления товаров/работ/услуг =  $R_2(A_2, A_8)$ , Платежные документы =  $R_3(A_3, A_9, A_{10})$ , где жирным выделены ключевые атрибуты отношений. Одно из возможных представлений гиперкуба приведено в табл. 1.

Таблица 1

Данные оплат

Код предприятия	12345			
Проект	111			
Договор поставки	222		223	
СБП	333	334	333	334
Дата	Сумма	Сумма	Сумма	Сумма
21.01.2021	75000	250000	357500	65500
26.01.2021	31000	310000	785000	96000
28.01.2021	70000	870080	6500	55000

Атрибуты измерений представлены жирным шрифтом, атрибуты фактов – курсивом, значения атрибутов – обычным шрифтом. Схема гиперкуба в табл. 1 может быть представлена в следующем виде:

$$\{R_3.A_9\} \times \{R_1.A_4\{R_1.A_5\{R_1.A_6\{R_1.A_7(R_3.A_{10})\}\}\}\},$$

где  $D_1 = \{R_3.A_9\}$  и  $D_2 = \{R_1.A_4, R_1.A_5, R_1.A_6, R_1.A_7\}$  – измерения,  $M = \{R_3.A_{10}\}$  – факты. Логическое ограничение:  $F = (R_1.A_2 \neq NULL \wedge R_1.A_3 \neq NULL \wedge R_1.A_4 = 12345 \wedge R_1.A_5 = 111 \wedge (R_1.A_6 = 222 \vee R_1.A_6 = 223) \wedge (R_1.A_7 = 333 \vee R_1.A_7 = 334) \wedge R_3.A_{11} \geq 01.01.2021 \wedge R_3.A_{11} \leq 28.02.2021 \wedge R_2.A_8 \geq 01.01.2020 \wedge R_2.A_8 \leq 31.01.2020)$ .

Соответствующий SQL запрос приведен на рис. 2.

```
SELECT pok.cod_pred AS pok_cod_pred, -- предприятие
       pok.id_zatr AS pok_id_zatr, -- проект
       pok.id_dog AS pok_id_dog, -- договор
       pok.num4 AS pok_num4, -- статья бюджетного плана
       opl.date_doc AS opl_date_doc, -- дата платежного документа
       opl.sum_doc AS opl_sum_doc -- сумма платежного документа
FROM pok_sw pok
LEFT JOIN pok_opl opl ON pok.id_opl = opl.id_opl
LEFT JOIN pok_prih pr ON pok.id_pr = pr.id_prih
WHERE pok.id_opl IS NOT NULL -- были уплачены деньги
AND pok.id_pr IS NOT NULL -- были получены товар/работы/услуги
AND pok.cod_pred = 12345 -- данные по предприятию с кодом 12345
AND pok.id_zatr = 111 -- по проекту с кодом 111
AND pok.id_dog = 222 -- по договору с кодом 222
AND pok.num4 = 333 -- платеж был по СБП с кодом 333
AND opl.date_doc BETWEEN TO_DATE('01.01.2021', 'dd.mm.yyyy') AND
TO_DATE('28.02.2021', 'dd.mm.yyyy') -- оплата в период
AND pr.date_doc BETWEEN TO_DATE('01.01.2020', 'dd.mm.yyyy') AND
TO_DATE('31.01.2020', 'dd.mm.yyyy') -- приход был в период
```

Рис. 2. SQL запрос данных оплат предприятию с указанной аналитикой

Fig. 2. SQL query of payments data to the enterprise with the specified analytics

С использованием описанной схемы БД в запрос аналогичным образом включаются любые другие измерения и факты (данные и аналитика оплат, счетов-фактур, акцептов, доходов и т. д.), позволяющие отследить все движение (документы, операции) по данному либо иному логическому ограничению.

**Пример 2.** Необходимо найти дебиторскую задолженность (сумму долга) предприятию с кодом 12345 за поставленные товары/работы/услуги по проектам 111 и 112.

Задано минимальное подмножество атрибутов:  $A_1$  – идентификатор связи,  $A_2$  – идентификатор записи о поступлении товара/работ/услуг,  $A_3$  – идентификатор записи платежа,  $A_4$  – код предприятия,  $A_5$  – идентификатор проекта,  $A_6$  – идентификатор договора поставки,  $A_7$  – номер документа прихода,  $A_8$  – сумма документа прихода,  $A_9$  – отчетный период. На данном множестве атрибутов существуют следующие функциональные зависимости:  $DEP = \{A_1 \rightarrow A_2A_3A_4A_5A_6, A_2 \rightarrow A_7A_8A_9\}$ . Получаем схему отношений: Связи документов =  $R_1(A_1, A_2, A_3, A_4, A_5, A_6)$ , Поступления товаров/работ/услуг =  $R_2(A_2, A_7, A_8, A_9)$ . Одно из возможных представлений гиперкуба приведено в табл. 2.

Схема гиперкуба в табл. 2 может быть представлена в следующем виде:

$$\{R_2.A_7\} \times \{R_1.A_4\} \{R_1.A_5\} \{R_1.A_6\} \{R_2.A_9(R_2.A_8)\} \},$$

где  $D_1 = \{R_2.A_7\}$  и  $D_2 = \{R_1.A_4, R_1.A_5, R_1.A_6, R_2.A_9\}$  – измерения,  $M = \{R_2.A_8\}$  – факты. Логическое ограничение:  $F = (R_1.A_2 \neq NULL \wedge R_1.A_3 = NULL \wedge R_1.A_4 = 12345)$ .

## Дебиторская задолженность

Код предприятия	12345							
Проект	111				112			
Договор поставки	222	223		224		225		
Отчетный период	333		334		333		334	
№ док.	I/2020	II/2020	III/2020	IV/2020	I/2020	II/2020	III/2020	IV/2020
11	75000	21500	250000	784200	357500	650800	65500	48000
12	31000	79200	310000	58100	785000	130000	96000	51700
13	70000	623300	870080	3700	6500	210000	55000	3580

Пример соответствующего SQL запроса приведен на рис. 3.

```
SELECT pok.cod_pred AS pok_cod_pred, -- предприятие
       pok.id_zatr AS pok_id_zatr, -- проект
       pok.id_dog AS pok_id_dog, -- договор
       prih.num_doc AS prih_num_doc, -- номер документа прихода
       prih.sum_doc AS prih_sum_doc, -- сумма документа прихода
       prih.cod_val AS prih_cod_val, -- код валюты документа прихода
       prih.buh_period AS prih_buh_period -- отчетный период
FROM pok_sw pok
LEFT JOIN pok_prih prih ON pok.id_pr = prih.id_prih
WHERE pok.id_pr IS NOT NULL -- был приход товара (наличие документа
прихода)
AND pok.id_opl IS NULL -- не было оплаты
AND pok.cod_pred = 12345 -- данные по предприятию с кодом 12345
```

Рис. 3. SQL запрос данных о дебиторской задолженности указанному предприятию

Fig. 3. SQL query of data on accounts receivable for the specified company

Результатом будут все строки по исследуемому предприятию, где зарегистрированы первичные приходные документы, по всем проектам, договорам поставки с возможными оплатами в любой валюте. Здесь можно анализировать, за какую поставку чем оплачивалось, какие были согласованы акцепты, авансы или платежи, по каким договорам, для кого, какие счета-фактуры для налогового учета и когда они должны попасть в книги покупок и продаж и т. д. Пример подсчета суммы платежей предприятию представлен на рис. 4.

```
SELECT SUM(DECODE(NVL(pok.id_opl, 0), 0, 0, pok.sum_doc)) AS
sum_opl, pok.cod_val
FROM pok_sw pok
WHERE pok.id_pr IS NOT NULL -- был приход товара (наличие приход-
ного ордера)
AND pok.cod_pred = 12345 -- данные по предприятию с кодом 12345
GROUP BY pok.cod_val
```

Рис. 4. SQL запрос суммы долга указанному предприятию

Fig. 4. SQL query for the amount owed to the specified company

**Пример 3.** Получить информацию о том, какие товары/работы/услуги были оплачены данным документом и на основании какого документа согласования (акцепта) это было сделано.

Задано минимальное подмножество атрибутов:  $A_1$  – идентификатор связи,  $A_2$  – идентификатор записи о поступлении товара/работ/услуг,  $A_3$  – идентификатор записи документа согласования (акцепта),  $A_4$  – идентификатор записи платежа,  $A_5$  – код предприятия,  $A_6$  – идентификатор проекта,  $A_7$  – идентификатор договора поставки,  $A_8$  – вид документа прихода,  $A_9$  – номер документа прихода,  $A_{10}$  – сумма прихода,  $A_{11}$  – номер акцепта,  $A_{12}$  – номер документа оплаты,  $A_{13}$  – дата документа оплаты. Имеются следующие зависимости:  $DEP = \{A_1 \rightarrow A_2A_3A_4A_5A_6A_7, A_2 \rightarrow A_8A_9(A_{10}), A_3 \rightarrow A_{11}, A_4 \rightarrow A_{12}A_{13}\}$  (три функциональных и одна многозначная).

В данном примере предлагается отказаться от необходимости выполнения функциональной зависимости [9–13]

$$D_1D_2...D_d \rightarrow M,$$

которая означает, что любому составному вектору значений размерностей  $D_1D_2...D_d$  соответствует не более одного вектора значений мер  $M$ .

Отказ от данной зависимости позволит использовать содержательные (не ключевые) атрибуты в размерностях и иметь в одной ячейке гиперкуба несколько значений (список) атрибута  $R_iA_j \in M$ . Списки значений используются в анализе данных, когда значения параметров не надо соотносить с объектами.

Получаем следующую схему отношений: Связи документов =  $R_1(A_1, A_2, A_3, A_4, A_5, A_6, A_7)$ , Поступления товаров/работ/услуг =  $R_2(A_2, A_8, A_9, A_{10})$ , Документы согласования платежей =  $R_3(A_3, A_{11})$ , Платежные документы =  $R_4(A_4, A_{12}, A_{13})$ . Одно из возможных представлений гиперкуба приведено в табл. 3.

Таблица 3

Данные платежного документа

Код предприятия	12345							
Проект	111							
Договор поставки	222				223			
Вид документа прихода	Приходная накладная		Приходный ордер		Акт выполненных работ		Приходная накладная	
№ акцепта	Сумма	№ док. прихода	Сумма	№ док. прихода	Сумма	№ док. прихода	Сумма	№ док. прихода
11	75000	21	357500	24	250000	27	65500	30
12	31000	22	785000	25	310000	28	96000	31
13	70000	23	6500	26	870080	29	55000	32

Схема гиперкуба в табл. 3 может быть представлена в следующем виде:

$$\{R_3.A_{11}\} \times \{R_1.A_5\{R_1.A_6\{R_1.A_7\{R_2.A_8(R_2.A_9)\}(R_2.A_{10})\}\}\},$$

где  $D_1 = \{R_3.A_{11}\}$  и  $D_2 = \{R_1.A_5, R_1.A_6, R_1.A_7, R_2.A_8\}$  – измерения,  $M = \{R_3.A_9, R_3.A_{10}\}$  – факты. Логическое ограничение:  $F = (R_4.A_{12} = 987 \wedge R_4.A_{13} = 21.01.2021)$ .

Пример соответствующего SQL запроса приведен на рис. 5.

Аналогичным образом реализован модуль АСУ ФЭЗ «Продажи» и другие вспомогательные модули. В совокупности эти модули составляют ядро системы.

Схема базы данных изначально спроектирована в соответствии с правилами нормальных форм [6; 7] и обладает свойством соединения без потерь информации (СБПИ) в соответствии с теоремой 5.8 [6]:

Пусть  $\sigma$  – декомпозиция отношения  $R$ , образованная схемами отношений в третьей нормальной форме, и пусть также  $X$  – ключ  $R$ . Тогда  $\tau = \sigma \cup \{X\}$  – декомпозиция  $R$ , такая, что все составляющие ее схемы отношений находятся в третьей нормальной форме. Эта декомпозиция сохраняет зависимости и сохраняет свойство соединения без потерь.

```

SELECT pok.cod_pred AS pok_cod_pred, -- предприятие
       pok.id_zatr AS pok_id_zatr, -- проект
       pok.id_dog AS pok_id_dog, -- договор
       prih.wid_doc AS prih_wid_doc, -- вид документа прихода
       prih.num_doc AS prih_num_doc, -- номер документа прихода
       prih.sum_doc AS prih_num_doc, -- сумма прихода
       sogl.num_doc AS osn_num_doc -- номер акцепта
FROM pok_sw pok
LEFT JOIN pok_prih prih ON pok.id_pr = prih.id_prih
LEFT JOIN pok_sogl sogl ON pok.id_osn = sogl.id_osn
LEFT JOIN pok_opl opl ON pok.id_opl = opl.id_opl
WHERE opl.num_doc = 987 -- № документа из таблицы регистрации оплат
      AND opl.date_doc = TO_DATE('21.01.2021', 'dd.mm.yyyy') -- дата
документа из таблицы регистрации оплат

```

Рис. 5. SQL запрос данных платежа

Fig. 5. SQL query for payment data

Полученные в результате кубы данных соответствуют формальным правилам, рассмотренным в работе [14], в соответствии с моделью данных «композиционная таблица», которая, в свою очередь, является обобщением модели «семантическая трансформация» [12], на случай списка значений в одной ячейке. Данный алгоритм формирует иерархии в измерениях гиперкуба, используя функциональные, многозначные зависимости исходной базы данных и иерархии атрибутов, заданные пользователем.

Описанная методика хранения и обработки данных позволяет эффективно выполнять как OLTP, так и OLAP операции без создания дополнительных структур данных и выполнения лишних преобразований. Это, в свою очередь, ведет к существенному снижению накладных расходов как по сложности составления запросов, так и по времени их выполнения по сравнению с классическим OLAP-инструментарием, способствует более оптимальному использованию ресурсов хранилища данных, позволяет избегать необходимости хранить избыточные данные. Применение описанного алгоритма в OLAP-системах позволяет сократить время на формирование схемы новой многомерной модели данных, а также сделать представление модели наиболее удобным для работы пользователя.

### Заключение

Предложенный вариант формирования и хранения данных первичных документов о хозяйственной деятельности предприятия обеспечивает детализированный отдельный учет финансово-хозяйственных операций покупок и продаж, и позволяет эффективно решать проблемы построения бухгалтерского, налогового и управленческого учетов. Применение вышеописанных схем, логики хранения и обработки данных для работы с различными их производными дает возможность дифференцировать элементы цепочки событий (состояний) в учете до первоисточников, увязанных в таблице связей в соответствии с логикой хозяйственных операций, что позволяет анализировать данные по любому интересующему набору аналитики. Таким образом, на любом этапе и участке учета всегда имеются целостные информационные связи. При этом обеспечивается возможность построения сложных OLAP структур данных в сочетании с быстродействием транзакционной СУБД. Эффективность работы АСУ ФЭЗ и лежащих в ее основе алгоритмов подтверждается опытом многолетней промышленной эксплуатации на одном из ведущих предприятий госкорпорации «Роскосмос» АО «ИСС» без привлечения внешних коммерческих решений.

## Библиографические ссылки

1. Вычугжанина Т. В., Долгова И. А. Использование хранилищ данных и OLAP-технологии в современных ERP-системах // Информационные технологии в моделировании и управлении: подходы, методы, решения : сб. тр. конф. Пенза, 2017.
2. Аврунев О. Е., Стасышин В. М. Модели баз данных. Новосибирск : Изд. НГТУ, 2018. 124 с.
3. Компания «Абис Софт». Применение OLAP-технологий для учетных систем на платформе 1С [Электронный ресурс]. URL: <https://www.cfin.ru/itm/olap/1c.shtml> (дата обращения: 11 02 2021).
4. Гончарова Ю. А., Андреева Н. М. Чтение данных периодического регистра сведений «1С:Предприятие 8.1» методами табличной модели информационной базы // Актуальные проблемы авиации и космонавтики. 2012. № 1. С. 393–394.
5. Картамышев А. С., Вилков Ю. В., Инновации в управлении предприятием РКО. Интегрированная информационная система управления предприятием АО «ИСС» // Актуальные вопросы проектирования автоматических космических аппаратов для фундаментальных и прикладных научных исследований. Анапа, 2017.
6. Ульман Дж. Основы систем баз данных. М. : Финансы и статистика, 1983. 334 с.
7. Maier D. The Theory of Relational Databases. Rockville: Computer Science Press, 1983.
8. Зыкин С. В., Мосин С. В., Полуянов А. Н. Технология раздельного формирования многомерных данных // Вестник Донского гос. техн. ун-та. 2016. № 2(85). С. 121–128.
9. Lechtenborger J., Vossen G. Multidimensional normal forms for data warehouse design // Information Systems Frontiers. 2003. No. 28. P. 415–434.
10. Lehner W., Albrecht J., Wedekind H. Normal forms for multidimensional databases // Proceedings of the Tenth International Conference on Scientific and Statistical Database Management. 1998. P. 63–72.
11. Mazon J., Trujillo J., Lechtenborger J. Reconciling requirement-driven data warehouses with data sources via multidimensional normal forms // Data & Knowledge Engineering, 2007. No. 3 (63). P. 725–751.
12. Zykin S.V. Formation of Hypercube Representation of Relational Database. Programming and Computer Software. 2006. No. 6 (32). P. 348–354.
13. Zykin S.V. Dynamic contexts of relational-type database. Informatics and Applications. 2014. No. 1 (8). P. 77–88.
14. Редреев П. Г. Построение иерархий в многомерных моделях данных // Известия Саратовского ун-та, 2009. № 4. С. 7–12.

## References

1. Vychugzhanina T. V., Dolgova I. A. [Using data warehouses and OLAP-technology in modern ERP-systems]. *Informacionnye tekhnologii v modelirovanii i upravlenii: podhody, metody, resheniya*. Penza, 2017 (In Russ.).
2. Avrunev O. E., Stasyshin V. M. *Modeli baz dannykh* [Database models]. Novosibirsk, Novosibirsk state technical university Publ., 2018 (In Russ.).
3. *Kompaniya "Abis Soft"*. *Primenenie OLAP-tekhnologii dlya uchetykh sistem na platforme 1S* [Abis Soft company. Using of OLAP technologies for accounting systems on the 1C platform]. Available at: <https://www.cfin.ru/itm/olap/1c.shtml> (accessed: 11.02 2021).
4. Goncharova Y. A., Andreeva N. M. [Reading data from the 1C: Enterprise 8.1 periodic information register using the methods of the infobase tabular model]. *Aktual'nye problemy aviatsii i kosmonavтики*. 2012, No. 1, P. 393–394 (In Russ.).

5. Kartamyshev A. S., Vilkov Y. V. [Innovation in aerospace enterprise control. Integrated information control system of the enterprise JSC “ISS”]. *Aktual'nye voprosy proektirovaniya avtomaticheskikh kosmicheskikh apparatov dlya fundamental'nyh i prikladnyh nauchnyh issledovaniy*, Anapa, 2017 (In Russ.).
6. Ulman J. *Osnovy sistem baz dannykh* [Principles of Database Systems]. Moscow, Finance and statistics Publ., 1983, 334 p.
7. Maier D. *The Theory of Relational Databases*. Rockville, Computer Science Press, 1983.
8. Zykin S. V., Mosin S. V., Poluyanov A. N. [Technology of separate generation of multidimensional data]. *Vestnik Donskogo gosudarstvennogo tekhnicheskogo universiteta*. 2016, No. 2(85), P. 121–128 (In Russ.).
9. Lechtenborger J., Vossen G. Multidimensional normal forms for data warehouse design. *Information Systems Frontiers*. 2003, No. 28, P. 415–434.
10. Lehner W., Albrecht J., Wedekind H. Normal forms for multidimensional databases. *Proceedings of the Tenth International Conference on Scientific and Statistical Database Management*. 1998, P. 63–72.
11. Mazon J., Trujillo J., Lechtenborger J. Reconciling requirement-driven data warehouses with data sources via multidimensional normal forms. *Data & Knowledge Engineering*. 2007, No. 3 (63), P. 725–751.
12. Zykin S. V. Formation of Hypercube Representation of Relational Database. *Programming and Computer Software*. 2006, No. 6 (32), P. 348–354.
13. Zykin S. V. Dynamic contexts of relational-type database. *Informatics and Applications*. 2014, No. 1 (8), P. 77–88.
14. Redreev P. G. [Construction of Hierarchies in Multidimensional Data Models]. *Izvestiya Saratovskogo universiteta*. 2009, No. 4, P. 7–12 (In Russ.).

© Картамышев А. С., Черныш Б. А., Мурыгин А. В., 2021

---

**Картамышев Александр Сергеевич** – начальник группы автоматизации задач управления в Обществе; АО «Информационные спутниковые системы» имени академика М. Ф. Решетнева». E-mail: kartam@iss-reshetnev.ru.

**Черныш Борис Александрович** – аспирант; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева. E-mail: borisblack@mail.ru.

**Мурыгин Александр Владимирович** – доктор технических наук, профессор, заведующий кафедрой информационно-управляющих систем; Сибирский государственный университет науки и технологий имени академика М. Ф. Решетнева. E-mail: avm514@mail.ru.

**Kartamyshev Alexandr Sergeevich** – Chief of control process automation group in the Company, JSC Academician M. F. Reshetnev “Information Satellite Systems”. E-mail: kartam@iss-reshetnev.ru.

**Chernysh Boris Aleksandrovich** – postgraduate student; Reshetnev Siberian State University of Science and Technology. E-mail: borisblack@mail.ru.

**Murygin Alexandr Vladimirovich** – Dr. Sc., Professor; Reshetnev Siberian State University of Science and Technology. E-mail: avm514@mail.ru.

---