

Дрождин В.В., Зинченко Р.Е., Герасимова Е.В., Кузнецов Р.Н., Севостьянов Р.Ю. Модель системного изоморфизма концептуальной модели предметной области и схемы базы данных. // Проблемы информатики в образовании, управлении, экономике и технике: Сб. статей IX Междунар. научно-техн. конф. – Пенза: ПДЗ, 2009. – С. 44-49.

## **МОДЕЛЬ СИСТЕМНОГО ИЗОМОРФИЗМА КОНЦЕПТУАЛЬНОЙ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ И СХЕМЫ БАЗЫ ДАННЫХ**

**В.В. Дрождин, Р.Е. Зинченко, Е.В. Герасимова, Р.Н. Кузнецов,  
Р.Ю. Севостьянов**

**Пензенский государственный педагогический университет  
им. В.Г. Белинского,  
г. Пенза, Россия**

Соответствие концептуальной модели предметной области и схемы базы данных в условиях их независимых изменений целесообразно формировать в виде системного изоморфизма, поддерживающего адекватное пользовательское представление предметной области и обеспечивающего эффективное функционирование информационной системы в процессе существования.

### **Drozhdin V.V., Zinchenko R.E., Gerasimova E.V., Kuznetsov R.N., Sevostyanov R.Y. The model of system isomorphism of conceptual model of knowledge domain and database scheme.**

The correspondence between the conceptual model of knowledge domain and database scheme in the conditions of their independent changes is to be produced in the form of a system isomorphism which supports an adequate user's performance of knowledge domain and provides an effective functioning of information system while existing.

Создание автоматизированных информационных систем (АИС), способных на основе концептуальной модели предметной области (ПО) автоматически формировать базу данных (БД) и поддерживать корректное отображение объектов ПО в БД на основе системы базовых SQL-запросов, является актуальной задачей. Решению этой задачи позволит в полном объеме реализовать независимость пользовательского представления ПО от организации данных в БД [1]. Это является важнейшим условием эффективного использования, организации и обработки данных и способствует приобретению АИС свойств самоорганизующейся системы.

Обеспечение независимости концептуальной модели ПО и схемы БД с возможностью их эволюции должно осуществляться на определенной модели соответствия между ними.

В существующих АИС принято статическое изоморфное соответствие концептуальной модели ПО и схемы БД, требующее полного отображения концептуальной модели ПО в схему БД и наоборот [2]. Его можно рассматривать как формально-математическое взаимно-однозначное отображение множеств или структур друг в друга. Динамическая поддержка изоморфного соответствия является очень трудоемкой, поэтому в существующих АИС используется именно статическое соответствие, устанавливаемое на этапе проектирования системы, а изменение

концептуальной модели ПО и/или схемы БД требуют перепроектирования АИС и приводят к существенным изменениям программного обеспечения системы.

Однако статическое изоморфное соответствие концептуальной модели ПО и схемы БД неприемлемы для АИС, в которых эти подсистемы могут изменяться (эволюционировать) относительно независимо, а система должна самостоятельно поддерживать определенное соответствие между ними.

Поэтому рассмотрим системное соответствие [3] концептуальной модели ПО и схемы БД, базирующейся на отображениях «модель ПО – схема БД» и «схема БД – модель ПО», которые в теории БД называются «внешний – концептуальный» и «концептуальный – внешний». При этом возможны следующие варианты отображений:

- 1) концептуальная модель ПО и схема БД частично отображаются друг в друга;
- 2) схема БД полностью отображается в концептуальную модель ПО, а концептуальная модель ПО частично отображается в схему БД;
- 3) концептуальная модель ПО полностью отображается в схему БД, а схема БД частично отображается в концептуальную модель ПО.

Первый вариант соответствия означает, что пользователь, имея представление о реальном мире, выраженное концептуальной моделью ПО, не сможет получить из АИС информацию о всей части реального мира, что нарушает логическую корректность системы, хотя система содержит некоторую дополнительную информацию, которая не будет «видна» извне.

Второй вариант подобен первому лишь с тем отличием, что БД не содержит информации, «невидимой» извне.

Третий вариант соответствия означает, что пользователь полностью получит из АИС информацию о реальном мире в соответствии со своим представлением, что обеспечивает логическую корректность системы. Однако система содержит некоторую дополнительную информацию, которая не будет «видна» извне. Дополнительная информация в БД может быть следующих типов:

отложенная модификация БД при удалении из концептуальной модели ПО различных понятий и отношений;

информация об объектах внешней среды, взаимодействующих с системой, и отношениях между ними, воспринимаемая и накапливаемая системой самостоятельно;

закономерности (знания) о поведении внешней среды и функционировании системы, вырабатываемые системой самостоятельно и используемые для повышения корректности, надежности и эффективности ее функционирования.

Поэтому в АИС с эволюционными изменениями концептуальной модели ПО и/или схемы БД целесообразно использовать динамическое системное соответствие третьего типа, обеспечивающее необходимое подобие и допустимое разнообразие концептуальной модели ПО и схемы БД.

Концептуальная модель ПО содержит следующие компоненты: понятие, представляющее пользователей АИС, отношения между пользователями, понятия, представляющие типы объектов ПО, отношения между понятиями [4].

Понятие, представляющее пользователей АИС, определяется в виде

$$u = \langle n_u, \rho, t_u, o_u, \delta \rangle,$$

где  $n_u$  – имя пользователя;  $\rho$  – пароль пользователя для входа в систему;  $t_u$  – тип пользователя: человек или внешняя система;  $o_u$  – идентификатор объекта, выступающего в роли пользователя СИС;  $\delta \in \Delta$  – полномочия пользователя.

Параметры  $n$  и  $\rho$  являются уникальными для каждого пользователя СИС.

Отношения иерархии между пользователями будем определять в виде

$$R_u = \{r_u \mid r_u = \langle u_1, u_2, n_r, t_{ru} \rangle\},$$

где  $R_u$  – совокупность отношений иерархии между всеми пользователями;  $r_u$  – отношение иерархии между двумя пользователями;  $u_1$  – вышестоящий пользователь;  $u_2$  – нижестоящий пользователь;  $n_r$  – имя отношения;  $t_{ru}$  – тип иерархии: основное подчинение – пользователь  $u_1$  осуществляет наиболее полное руководство и общий контроль пользователя  $u_2$ , дополнительное подчинение – пользователь  $u_1$  осуществляет руководство и контроль пользователя  $u_2$  в какой-то сфере.

Основное подчинение некоторого пользователя возможно только одному вышестоящему пользователю, а дополнительных подчинений может быть несколько.

Понятия, представляющие в модели ПО различные типы объектов, определяются в виде

$$V = \{v \mid v = \langle n_v, v_s, v_c \rangle\},$$

где  $V$  – множество всех понятий модели ПО;  $v$  – понятие, представляющее объекты ПО определенного типа;  $n_v$  – имя понятия;  $v_s$  – состав понятия;  $v_c$  – содержание понятия.

Состав понятия отражает совокупность понятий более низкого уровня, задающих структуру и свойства объектов, соответствующих данному понятию.

Содержание понятия является предикатом, определяющим объекты, относящиеся к данному понятию, и выделяющим эти объекты среди всех других объектов ПО.

Между понятиями определяются различные отношения в виде

$$R_v = \{r_v \mid r_v = \langle v_1, v_2, n_r, t_{rv} \rangle\},$$

где  $R_v$  – совокупность различных отношений между всеми понятиями ПО;  $r_v$  – отношение типа  $t_{rv}$  между понятиями  $v_1$  и  $v_2$ ;  $n_r$  – имя отношения;  $t_{rv}$  – тип отношения: агрегация (часть-целое) – понятие  $v_2$  является компонентом (частью) понятия-агрегата  $v_1$ , классификация – понятие  $v_2$  является подклассом класса  $v_1$ , обобщение (род-вид) – каждое видовое понятие  $v_2$  является категорией родового понятия  $v_1$ , абстрагирование – понятие  $v_2$  является конкретизацией понятия-образа  $v_1$ .

Семантические отношения между понятиями и их типы пользователи могут задавать самостоятельно, поэтому модель ПО будет представляться в форме семантической сети.

Модель ПО определяется в виде

$$M = \langle M_u \rangle = \langle V, R_v \rangle,$$

где  $M$  – модель ПО, являющаяся композицией  $M_u$ ;  $M_u = \langle m_u \rangle = \langle V_u, R_{vu} \rangle$  – модель ПО пользователя;  $m_u' = \langle V_u', R_{vu}' \rangle$  – подмодель (секция) модели ПО пользователя;

$V_u$  – множество понятий модели ПО пользователя;  $R_{vu}$  – множество отношений между понятиями модели ПО пользователя;  $V'_u$  – множество понятий подмодели (секции) модели ПО пользователя;  $R'_{vu}$  – множество отношений между понятиями подмодели (секции) модели ПО пользователя.

Схема базы данных формируется в рамках реляционной модели данных. Это не снижает общности модели динамического отображения концептуальной модели ПО в схему БД, так как внешне информация объектной БД может представляться двумерными таблицами.

Для построения модели системного изоморфизма концептуальной модели ПО и схемы БД введем следующие обозначения:

$M$  – модель ПО;

$R$  – схема БД;

$R_M$  – часть схемы БД, соответствующая  $M$ ;

$R_e$  – часть схемы БД, представляющая дополнительную информацию, недоступную пользователям.

Схема БД является композицией  $R_M$  и  $R_e$ :

$$R = R_M \circ R_e.$$

Системный изоморфизм  $M$  и  $R$  обозначим как

$$M \Leftrightarrow R.$$

При этом подобие  $M$  и  $R$  базируется на изоморфизме  $M \equiv R_M$ , где  $\equiv$  – означает взаимно-однозначное отображение  $M$  и  $R_M$  друг в друга. Степень подобия  $M$  и  $R$  определяется величиной отношения  $R_M/R$ , а степень разнообразия равна  $R_e/R = 1 - R_M/R$ . В случае  $R_M/R \rightarrow 1$  степень подобия  $M$  и  $R$  высокая, так как  $R_e \rightarrow \emptyset$ , а при  $R_M/R \rightarrow 0$  степень подобия  $M$  и  $R$  низкая, так как  $R \approx R_e$  и система обрабатывает преимущественно свою собственную информацию, т.е. работает сама на себя, а не на внешнюю среду. Таким образом, системный изоморфизм  $M$  и  $R$  заключается в разумном сочетании их степени подобия и разнообразия, а точнее, в определении объема информации  $R_e$ , порождаемой самой системой, и ее изменении в процессе существования АИС.

Механизм реализации модели системного изоморфизма  $M$  и  $R$  заключается в следующем. Для каждого понятия концептуальной модели ПО создается базовый SQL-запрос, позволяющий формировать реальный объем понятия (множество объектов, соответствующих содержанию понятия). Для обеспечения независимости изменений концептуальной модели ПО пользователями и схемы БД подсистемой оптимизации АИС содержит модуль корректировки базовых SQL-запросов в процессе функционирования системы. Это позволяет отслеживать все изменения концептуальной модели ПО и схемы БД и приводить их в соответствие на основе модели системного изоморфизма. Таким образом, базовые SQL-запросы для каждого понятия  $v_i$  в любой момент функционирования АИС предоставят пользователям всю информацию об объектах понятия, содержащуюся в БД.

Информация, формируемая системой самостоятельно и представленная в  $R_e$ , предназначена для повышения корректности, надежности и эффективности функционирования АИС. Она позволяет отражать особенности ПО, организации

БД и использования данных в процессе функционирования АИС, что способствует более эффективной организации системы.

Предложенная модель системного изоморфизма позволяет поддерживать в АИС модель ПО, адекватную пользовательскому представлению, и обеспечивает повышение эффективности функционирования АИС в процессе ее существования.

#### Библиографический список

1. Мартин Дж. Организация баз данных в вычислительных системах. – М. : Мир, 1980. – 663 с.
2. Дейт К. Введение в системы баз данных. – М. : Вильямс, 2006. – 1328 с.
3. Система, симметрия, гармония. – М. : Мысль, 1988. – 315 с.
4. Дрождин В.В. Построение системы моделирования на основе самоорганизующейся информационной среды // Модели и алгоритмы для имитации физико-химических процессов : материалы Междунар. науч.-техн. конф. – Таганрог : Изд-во НП «ЦРЛ», 2008. – С. 251 – 255.