

## Построение схемы реляционной базы данных по нерекурсивной полусхеме

А. Г. Житарюк, Т. В. Семенова

*Харьковский национальный университет им. В.Н. Каразина, Украина*

In the present paper the problem of the information system modeling and the approach to the data modeling using the scheme was considered. The algorithm of relational database structure building by non-recursive scheme was developed.

### 1. Общая постановка задачи и её актуальность

Характерный для современного общества процесс массового внедрения информационных технологий, базирующихся на обработке информации при помощи вычислительной техники, приводит к росту спроса на информационные системы различного назначения. Для повышения эффективности процесса разработки подобных систем используются CASE-средства различного типа [1].

Анализ широко используемых CASE-средств [2-3] показал, что существует разделение между средствами поддержки проектирования и разработки самих систем и средствами разработки хранилищ данных (в частности СУБД), которые будет использовать система. Подобное разделение не является естественным, поскольку именно структурно-логическая модель предметной области, полученная на этапе анализа, является основой при разработке концептуальной модели базы данных. Этот факт игнорируется большинством CASE-средств в силу того, что структурно-логическая модель предметной области не является формализованным понятием. Как следствие этого, CASE-средства не содержат алгоритмов проверки корректности и непротиворечивости структурно-логической модели, которая является отправной точкой для дальнейшего проектирования.

С целью формализации представления структурно-логической модели предметной области авторами в работах [4-5] была предложена теория полусхем.

### 2. Цели работы

Представление концептуальной модели предметной области в терминах теории полусхем позволяет разработать формальные методы построения схемы реляционной базы данных как хранилища данных для проектируемой информационной системы. Однако задача генерации схемы базы данных по произвольной полусхеме является достаточно трудной. Основная проблема состоит в неоднозначном представлении рекурсивно определяемых понятий средствами реляционной модели данных. Вследствие этого, область исследования была сужена до нерекурсивных полусхем [6] и подмножества произвольных полусхем, удовлетворяющих определенным свойствам. Целью

данной работы является описание алгоритмов генерации схемы реляционной базы данных по нерекурсивной полусхеме.

### 3. Краткий обзор теории полусхем

Пусть  $N$  – конечное множество, элементы которого соответствуют именам понятий предметной области,  $R$  – конечное множество, элементы которого соответствуют именам ролей, т.е. ссылкам внутри экземпляра понятия на его структурные части.

Используемые обозначения:

$M(R, N)$  - множество частичных отображений из  $R$  в  $N$ ;

$dom(f)$  - область определения отображений  $f \in M(R, N)$ ;

$\varepsilon$  - отображение с пустой областью определения;

$R^*$  - множества всех слов над  $R$ ;

$R^+$  - множество непустых слов над  $R$ ;

$e$  - пустое слово.

**Полусхемой** предметной области называется тройка  $S = (N, R, D)$ , для которой выполняются следующие условия:

- 1) для любого  $n \in N$ , если  $(n, \varepsilon) \in D$ , то  $\{f \in M(R, N) \mid (n, f) \in D\} = \{\varepsilon\}$ ;
- 2) для любого  $n \in N$  и для любых двух вариантов определения  $f, g \in M(R, N)$  таких, что  $(n, f) \in D$ ,  $(n, g) \in D$  и  $r \in dom(f) \cap dom(g)$ , где  $r \in R$ , выполнено свойство  $f(r) = g(r)$ .

где

$N$  - множество понятий предметной области,

$R$  - множество ролей,

$D \subset N \times M(R, N)$  - отношение, задающее соответствие между понятиями и вариантами определения.

Задается частичное отображение  $\tau: N \times R \rightarrow N$  следующим образом:  $\tau(n, r)$  определено в том и только том случае, если существует  $f \in M(R, N)$ , для которого  $(n, f) \in D$ ,  $r \in dom(f)$  при этом  $\tau(n, r) = f(r)$ .

**Именующей нитью** понятия  $n$  называется элемент  $(n, w)$  из множества  $N \times R^*$ , который удовлетворяет одному из следующих условий:

1.  $w = e$  и  $(n, \varepsilon) \in D$ ;

2. для  $w = r_1 r_2 \dots r_k$  в последовательности  $n_0 = n$ ,  $n_i = \tau(n_{i-1}, r_i)$ , где  $i = 1, \dots, k$ , все члены определены.

Для полусхемы  $S = (N, R, D)$  понятие  $n \in N$  называется **базовым**, если для всякого  $f \in M_+(R, N)$  выполняется  $(n, f) \in D$ .

Функция  $f \in M(R, N)$  называется **сигнатурой понятия**  $n \in N$ , если  $(n, f) \in D$ .

$F_n = \{f : M(R, N) \mid (n, f) \in D\}$  - множество сигнатур понятия  $n$ .

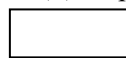
Понятие  $m$  **участвует в определении понятия**  $n$ , если существует  $t \neq e$ -именующая нить, такая, что  $\tau(n, t) = m$ .

Понятие  $m$  **непосредственно участвует в определении понятия**  $n$ , если существует такая сигнатура  $f \in F_n$  и существует  $r \in \text{dom}(f)$  такие, что  $f(r) = m$ .

Понятие  $n$  называется **рекурсивно определяемым понятием**, если существует такая именующая нить  $t \neq e$ , что  $\tau(n, t) = n$ .

Полусхема называется **нерекурсивной**, если она не содержит рекурсивно определяемых понятий.

Для представления полусхем используется следующая графическая нотация.



понятие (сущность) предметной области,



вариант определения понятия

#### 4. Генерация схемы базы данных по нерекурсивной полусхеме

Каждому понятию  $n \in N$  ставится в соответствие отношение в базе данных или атрибут отношения. А агрегирующие связи между понятиями преобразуются в связи между отношениями, реализованные с помощью внешних ключей.

В множестве базовых понятий выделяется подмножество  $\tilde{N}_0$  стандартно определяемых понятий, например String, Integer и др. Такие понятия скорее задают тип другого понятия нежели определяют качественно новые понятия. Поэтому стандартно определяемым понятиям ставится в соответствие атрибут в отношении.

Всем остальным базовым понятиям соответствуют отдельные отношения.

Небазовые понятия в свою очередь разбиваются на два подмножества в зависимости от количества вариантов определения, которые с ними ассоциированы. Такое разбиение обусловлено тем, что правила преобразования

таких понятий в набор отношений реляционной базы данных являются различными.

Ниже сформулированы правила построения схемы РБД.

1. Если понятие  $n$  является базовым, но не стандартно определяемым понятием, т.е.  $n \in N_0 \setminus \tilde{N}_0$ , то создается отношение базы данных

**Rel<sub>n</sub>(id<sub>n</sub> : Integer, atr<sub>n</sub> : String),**

где id<sub>n</sub> – искусственный ключ отношения Rel<sub>n</sub>.

2. Если понятие  $n \in N \setminus N_0$  имеет единственную сигнатуру  $f$ , то создается отношение с именем Rel<sub>n</sub>, в котором количество атрибутов зависит от функции  $f$ . Для всех ролей  $r \in \text{dom}(f)$  выполняется следующее:

- 1) Если  $m = f(r) \in \tilde{N}_0$ , то в отношении Rel<sub>n</sub> добавляется атрибут с именем atr<sub>r</sub> и типом  $m$ .
- 2) Если  $m = f(r) \notin \tilde{N}_0$ , то в отношении Rel<sub>n</sub> добавляется атрибут с именем Ref<sub>m</sub>, который является ссылкой на атрибут id<sub>m</sub> отношения Rel<sub>m</sub>.

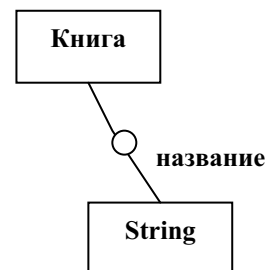
3. Если понятие  $n \in N \setminus N_0$  ассоциировано с несколькими вариантами определения, то создается отношение с двумя атрибутами Rel<sub>n</sub>(id<sub>n</sub>, type<sub>n</sub>), где атрибут с именем type<sub>n</sub> имеет тип *String*, и содержит в качестве значений атрибутов имена вариантов определения понятия  $n$ . Для всех вариантов определения  $f \in M(R, N)$  таких, что  $(n, f) \in D$  выполняется следующее: Создается отношение с именем Rel<sub>f</sub>, в котором принцип формирования атрибутов аналогичен пункту 2.

### Пример 1.

Понятие **Книга** характеризуется строкой, которая выступает в роли названия книги.

Понятие *String* является базовым, стандартно определяемым понятием, т.е.  $String \in \tilde{N}_0$ , потому создается отношение с именем **Rel\_Книга**, которое будет иметь следующий вид:

**Rel\_Книга (id\_Книга : Integer, atr\_название : String)**



### Пример 2.

Понятие **Книга** характеризуется понятием **Жанр**.

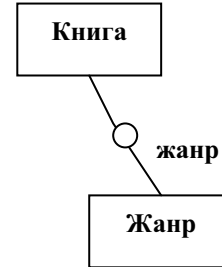
Понятие  $Жанр \in N_0$  является базовым, но не является стандартно определяемым, потому предварительно требуется создать отношение базы данных, соответствующее понятию **Жанр**. Согласно правилу 1, оно будет иметь следующий вид:

**Rel\_Жанр** (**id\_Жанр** : Integer, **atr\_Жанр** : String)

В свою очередь понятию **Книга** ставится в соответствие отношение **Rel\_Книга**, содержащее ссылку на отношение **Rel\_Жанр**. Таким образом, отношение **Rel\_Книга** будет иметь следующий вид:

**Rel\_Книга** (**id\_Книга** : Integer, **ref\_жанр** : Integer),

где **ref\_жанр** – ссылка на атрибут **id\_Жанр** отношения **Rel\_Жанр**.

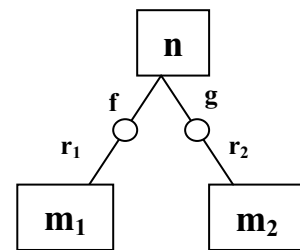


**Пример 3.**

Понятие  $n$  имеет два варианта определения, следовательно, построение схемы базы данных проходит согласно правилу 3.

На первом шаге создается отношение, которое имеет следующий вид:

**Rel\_n**(**id\_n** : Integer, **type\_n** : String), где доменом атрибута **type\_n** является множество имен вариантов определения понятия  $n$ . Отношение **Rel\_n** сразу же может быть заполнено.



id_n	type_n
1	f
2	g

На следующем шаге создается два отношения **Rel\_f** и **Rel\_g** согласно правилу 2.

Описанный набор правил может применяться только в определенной последовательности. Это обусловлено тем, что к моменту построения отношения, соответствующего заданному понятию  $n$ , должны быть построены все отношения, соответствующие понятиям, которые участвуют в определении понятия  $n$ .

Иначе говоря, понятия должны рассматриваться по мере усложнения их структуры.

На первом шаге должны быть построены отношения, соответствующие базовым понятиям, которые не являются стандартно определенными.

На втором шаге должны быть рассмотрены понятия, в определении которых непосредственно участвуют только базовые понятия.

На третьем шаге строятся отношения для таких понятий, в определении которых участвуют базовые понятия, а также понятия, рассмотренные на втором шаге.

Таким образом, на каждом следующем шаге может быть рассмотрено вполне определенное множество понятий:

$$N_i = \{n \in N \setminus N_{i-1} \mid (\forall f \in F_n) \text{im}(f) \subset \bigcup_{k=0}^{i-1} N_k\}.$$

Иначе говоря, множество  $N_i$  будет состоять из тех понятий, в определении которых участвуют только понятия из множества  $\bigcup_{k=0}^{i-1} N_k$ .

### **Алгоритм генерации схемы базы данных по нерекурсивной полусхеме**

Пусть  $S = (N, R, D)$  - нерекурсивная полусхема.

На первом шаге создаются отношения схемы базы данных, соответствующие понятиям из  $N_0 \setminus \tilde{N}_0$ .

Второй шаг работы алгоритма заключается в построении последовательности множеств  $\{N_i\}_{i=1}^k$ .

Далее итеративно построятся отношения схемы базы данных, соответствующие понятиям из множеств  $N_i$ . Сначала строятся отношения, соответствующие понятием из  $N_1$ , далее из  $N_2$  и так далее, пока не будут рассмотрены все  $N_i$ . При построении отношений, соответствующих понятиям из  $N_i$ , отдельно рассматриваются понятия с одним и несколькими вариантами определения, поскольку они требуют различных алгоритмов.

Структурная схема алгоритма приведена на рисунке 1.

**Вход:** полусхема  $S = (N, R, D)$

**Выход:** схема реляционной базы данных

**Используемые функции:**

$SignatureCount(n)$  – функция, вычисляющая количество вариантов определения, ассоциированных с понятием  $n$ ;

$CreateSingleDefinedConcept(n)$  – функция, выполняющая построение отношений, соответствующих понятию  $n$  с одним вариантом определения;

$CreateMultiDefinedConcept(n)$  – функция, выполняющая построение отношений, соответствующих понятию  $n$  с несколькими вариантами определения.

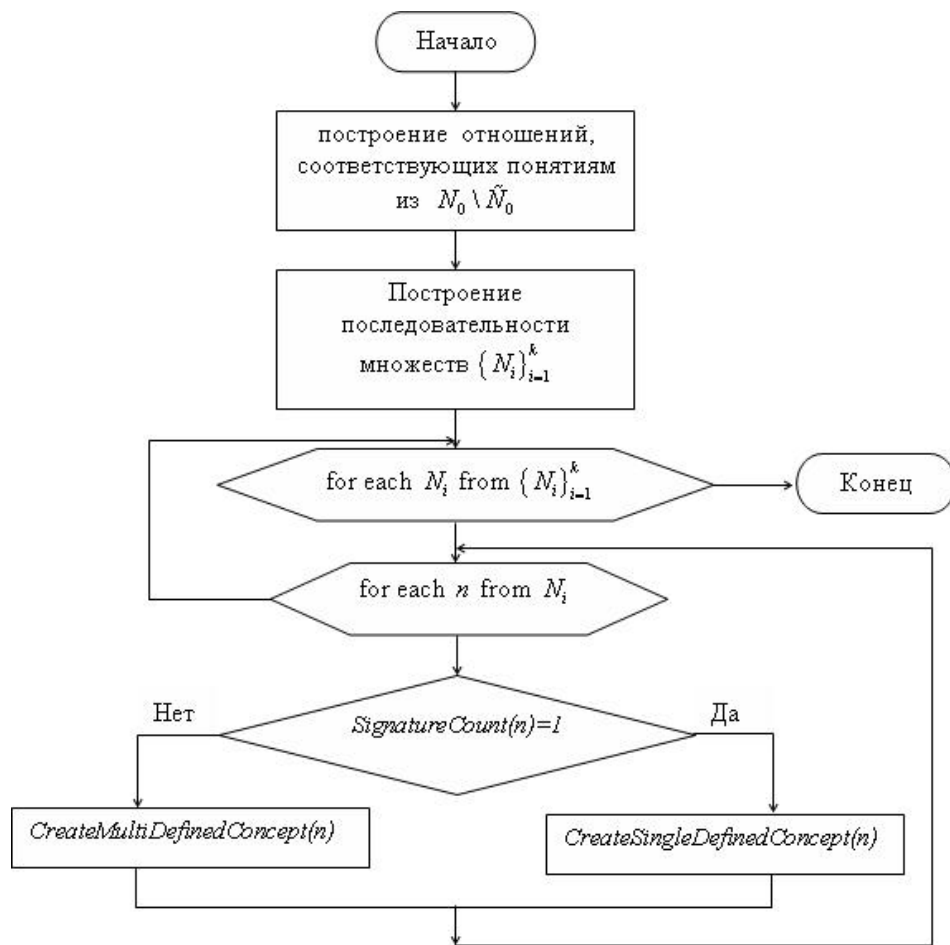


Рис.1. Алгоритм построения схемы БД по нерекурсивной полусхеме

### 5. Нерешенные проблемы и цели работы

Следует отметить, что приведенные алгоритмы позволяют строить схему базы данных только по нерекурсивной полусхеме. Опыт разработки информационных систем [7] позволяет утверждать, что в большинстве случаев структурно-логические модели предметных областей содержат определенное количество рекурсивно определяемых понятий. Искусственное исключение таких понятий из рассмотрения, сделанное в данной работе, является лишь временным решением.

### 6. Выводы по результатам и направления дальнейших исследований

В работе рассмотрен алгоритм формирования схемы реляционной базы данных по структурно-логической модели предметной области информационно-поисковой системы, представленной в терминах теории полусхем.

Рассмотренный алгоритм предполагает обработку лишь нерекурсивных полусхем. Это объясняется тем, что процесс проектирования баз данных включает элемент творчества и не может быть полностью автоматизирован. Особенно если речь идет о представлении в реляционной базе данных рекурсивно определяемых понятий.

В дальнейшем можно расширить полученный результат, рассмотрев класс полусхем, которые содержит стандартные рекурсивные конструкции, например, список.

Но даже при построении схемы реляционной базы данных по произвольной полусхеме приведенный в работе алгоритм может быть полностью использован, как один из этапов преобразования. Так, например, построение схемы базы данных по произвольной полусхеме разбивается на три этапа:

- выделение в полусхеме множества рекурсивно определяемых понятий;
- преобразование нерекурсивных понятий согласно приведенному в работе алгоритму;
- интерактивное преобразование рекурсивных понятий совместно с разработчиком базы данных.

#### ЛИТЕРАТУРА

1. Сомервилл, Ифн. Инженерия программного обеспечения, 6-у издание. : Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 624с.: ил.
2. Боггс У., Боггс М. UML и Rational Rose. - М.: "ЛОРИ", 2000. - 582 с.
3. Кватрани, Терри Rational Rose 2000 и UML. Визуальное моделирование : перевод с : англ. / Терри Кватрани ; Предисл. Грейди Буч ; перевод А.Б. Литвин . - М. : ДМК Пресс, 2001 . - 176 с.
4. Жолткевич Г.Н., Семенова Т.В. К проблеме формализации концептуального моделирования информационных систем // Вісник Харківського національного університету. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління», №605, 2003, с. 33-42.
5. Семенова Т.В. Морфизмы полусхем и их приложения. // Вісник Харківського національного університету. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління», №703, 2005, с. 198-206.
6. Жолткевич Г.Н., Федорченко К.А. Об одном классе концептуальных моделей. // Вестник НТУ «ХПИ». Сборник научных трудов. Тематический выпуск «Системный анализ, управление и информацион технологии». – Харьков: НТУ «ХПИ». – 2006. - №19. – с. 51-56.
7. Советов Б.Я., Яковлев С.А. Моделирование систем. – М.: Высш. шк., 1985. – 271 с., ил.