

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Томский государственный университет  
Горно-Алтайский государственный университет  
Институт оптики атмосферы им. В.Е. Зуева СО РАН

# **НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ИССЛЕДОВАНИИ СЛОЖНЫХ СТРУКТУР**

**МАТЕРИАЛЫ ДЕСЯТОЙ РОССИЙСКОЙ КОНФЕРЕНЦИИ  
С МЕЖДУНАРОДНЫМ УЧАСТИЕМ**

Томск  
Издательский Дом Томского государственного университета  
2014

Работы велись по следующим направлениям:

1. Обеспечение реализации всех ограничений целостности на специализации, включая ограничения непересечения и участия, а не только проверку полного функционального отображения к родительскому классу. Если специализация непересекающаяся, то каждый экземпляр суперкласса может быть членом только одного из подклассов. В случае пересекающейся специализации – нескольких подклассов. Специализация с полным участием предполагает, что каждый экземпляр суперкласса является членом хотя бы одного из подклассов. В специализации с частичным участием экземпляр суперкласса может не являться членом ни одного из подклассов.

2. Реализация возможности определения оснований выделения подклассов специализации – отображений, определяющих принадлежит ли подклассу объект суперкласса.

3. Расширение спектра методов реализации специализаций в реляционной схеме (преобразователь схем Oracle Designer обеспечивает реализацию не всех вариантов структур).

### Литература

1. Колетски П., Дорси П. Oracle Designer. Настольная книга пользователя : пер. с англ. М. : Лори, 1999. 592 с.
2. Barker R. Case\*Method: Entity Relationship Modelling. Wokingham, England : Addison-Wesley, 1990. 240 p.
3. Бабанов А.М. Семантическая модель «Сущность – Связь – Отображение» // Вестник Томск. гос. ун-та. Сер. Управление, вычислительная техника и информатика. 2007. № 1. С. 77–91.

## ОСОБЕННОСТИ СЕМАНТИЧЕСКОГО МОДЕЛИРОВАНИЯ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ ERM-МОДЕЛИ

*А.М. Бабанов, Т.И. Бабанова*

Национальный исследовательский Томский государственный университет, Томск, Россия  
babanov2000@mail2000.ru

Общепризнанными и наиболее широко распространенными подходами семантического моделирования являются ER-модель («Entity – Relationship») Чена [1], UML («Unified Modeling Language») [2] и получившая развитие в последнее время OR-модель («Object – Role») Халпина [3].

Созданную и развиваемую на факультете информатики ТГУ ERM-модель («Entity – Relationship – Mapping») или модель «Сущность – Связь – Отображение») [4] от своих предшественниц отличают следующие характерные черты.

1. Основополагающие семантические концепции аналогичны базовым концепциям естественных языков – объект, отображение (в логике аналог – предметная функция).

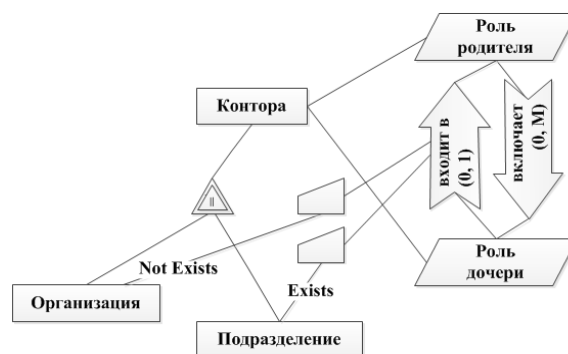
2. Выразительная мощность языка описания структуры данных и бизнес-правил (ограничений целостности) предметной области (ПрО) сравнима с естественным языком.

3. Опора в модели делается на обобщенные математические концепции – множество (или класс) и отображение, а не на специально изобретенные конструкции – роли, сущности, связи, ключи и т.д.

4. В основе модели лежит математизированная теория семантически значимых отображений (ТСЗО), доведенная до уровня формальной системы, фундамент которой составляет исчисление предикатов первого порядка. Операции и ограничения целостности модели имеют определения в виде выражений этой формальной системы.

5. Структурные понятия модели делятся на базовые и производные, изоморфные друг другу; в связи с этим имеется возможность использования в схеме синонимичных форм данных, облегчающая задачу проектировщика схемы.

Подобный подход позволяет человеку при формализации ПрО оперировать в основном знакомыми понятиями (сущность, связь), прибегая к использованию новых форм (отображение) лишь в случае недостаточной выразительности первых. В некоторых случаях понятия «класс» и «отображение»



можно не использовать вовсе. Привычные понятия ER-модели обеспечивают более понятные человеку формы восприятия данных.

6. Язык манипулирования данными может использоваться для определения бизнес-правил. С помощью операций над отображениями определяются так называемые получаемые (или производные) отображения, экстенционалы которых в отличие от хранимых отображений вычисляются системой. Их характеристики являются дополнительными средствами выражения семантики ПрО.

7. Схема данных может содержать избыточные описания ПрО за счет:

– синонимии (использования в схеме разных форм представления для одних и тех же явлений ПрО и законов их взаимодействия);

– следствий и эквивалентностей отображений.

8. Имеется возможность определять в схеме хранимые и получаемые данные, последние непосредственно не хранятся на диске, а вычисляются с помощью тех или иных правил, определенных в схеме.

С использованием ERM-модели семантическая методика проектирования схем БД может быть доведена до идеала – исчерпывающая требования к БД формализация описаний ПрО в рамках семантической модели и трансляция этих описаний в полную и эффективную СУБД-ориентированную схему данных, не требующую дополнительных расширений.

#### Литература

1. *Chen P.P-S.* The Entity-Relationship Model – Toward a Unified View of Data // ACM Transactions on Database Systems. Vol. 1, No. 1. March 1976. pp. 9–36.
2. *Bennett S., McRobb S. and Farmer R.* Object-Oriented Systems Analysis and Design, Fourth Edition. McGraw-Hill Higher Education. 2010, 688 p.
3. *Halpin, T. and Morgan, T.* Information Modeling and Relational Databases, Second Edition. Morgan Kaufman, 2008, 943 p.
4. *Бабанов А.М.* Семантическая модель «Сущность – Связь – Отображение» // Вестник ТГУ. УВТиИ. 2007. № 1. С. 77–91.

## К РАЗРАБОТКЕ ПРОГРАММЫ КОДОГЕНЕРАЦИИ «ИСКУССТВЕННЫЙ ПРОГРАММИСТ»

*С.В. Батрацкий, С.А. Прокопенко*

Национальный исследовательский Томский государственный университет, Томск, Россия  
twilight7775@gmail.com, s.prokopenko@sibmail.com

В данном докладе предлагается подход к разработке программы автоматической кодогенерации по блок-схеме алгоритма на основе булевых матриц. На первом шаге мы рассматриваем только такие блок-схемы алгоритмов, которые удовлетворяют определенным условиям. Как обычно, овалы «Начало» и «Конец» соответствуют началу и концу программы. Соответственно блок «Начало» не имеет входящих дуг и имеет одну исходящую дугу; блок «Конец» не имеет исходящих дуг. Прямоугольники соответствуют выполнению некоторой (одной) вычислительной операции, и так же, как и блок ввода-вывода данных (параллелограмм), имеют не менее одной входящей дуги и только одну исходящую дугу. Ромбовидный логический блок описывает условия, требуемые для выполнения последующих операций, и имеет не менее одной входящей дуги и только две исходящие дуги. Таким образом, в овалы начала и конца можно занести только значения «начало» и «конец» соответственно, в блоке вычислений могут быть только действия, например,  $i = i + j$ ; в ромбовидном блоке – условие, например,  $i = j$ ; в блоке ввода-вывода данных – переменная, значение которой необходимо ввести с клавиатуры или вывести на экран компьютера.

Для разработки программы автоматической генерации программы по блок-схеме алгоритма последнюю можно рассматривать как ориентированный граф [1, 2], вершинам которого поставлены в соответствие блоки блок-схемы, а ребрам – дуги, ассоциированные с блоками блок-схемы. Соответственно компьютерное представление блок-схемы алгоритма содержит две матрицы: матрицу смежности вершин и матрицу инцидентности графа, и одномерный массив, в котором хранится соответствие между вершинами графа и блоками блок-схемы с выполняемыми в них действиями. Номер  $i$  элемента массива соответствует  $i$ -й вершине графа, то есть определенному блоку блок-схемы. В  $i$ -м элементе массива