



**ИТММ · 2010**

**«ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И  
МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ»**

**ЧАСТЬ 2**

Томский государственный университет  
Кемеровский государственный университет  
Кемеровский научный центр СО РАН  
Институт вычислительных технологий СО РАН  
Филиал Кемеровского государственного университета  
в г. Анжеро-Судженске

**ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ  
И МАТЕМАТИЧЕСКОЕ  
МОДЕЛИРОВАНИЕ  
(ИТММ-2010)**

**Материалы IX Всероссийской  
научно-практической конференции  
с международным участием**

**19–20 ноября 2010 г.**

**Часть 2**

Издательство Томского университета

2010

УДК 519  
ББК 22.17  
И74

Редколлегия:

*Р. Т. Якупов*, д-р физ.-мат. наук, профессор (отв. ред.);

*И. Р. Гарайшина*, канд. физ.-мат. наук, доцент;

*А. С. Шкуркин*, канд. техн. наук, доцент

**Информационные технологии и математическое моделирование**  
И74 (ИТММ-2010): Материалы IX Всероссийской научно-практической конференции с международным участием (19–20 ноября 2010 г.). – Томск: Изд-во Том. ун-та, 2010. – Ч. 2. – 196 с.

ISBN 978–5–7511–1949–2

В часть 2 вошли материалы секций «Информационные технологии», «Математические методы и модели в науке и технике» и «Численные методы и комплексы программ».

Для специалистов в области информационных технологий и математического моделирования.

**УДК 519**  
**ББК 22.17**

*Конференция проводится при поддержке Российского фонда  
фундаментальных исследований (проект № 10–07–06095–з)*

ISBN 978–5–7511–1949–2

© Томский государственный университет, 2010  
© Кемеровский государственный университет, 2010  
© Кемеровский научный центр СО РАН, 2010  
© Институт вычислительных технологий СО РАН, 2010  
© Филиал Кемеровского государственного университета в г. Анжеро-Судженске, 2010

### **Назначение базовых и производных понятий**

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. Таким образом, привычные понятия ER-модели обеспечивают более понятные человеку формы восприятия данных.

Особенности языка ERM-модели, ориентированные на удобства для проектировщика, неизбежно влекут за собой возможность синонимии. Существование синонимичных элементов в схеме может привести к ее некорректности в случае задания противоречивых свойств у элементов-синонимов. Поэтому формальная система модели должна обеспечить соответствующие проверки на непротиворечивость схем.

Также формальная система должна предоставить правила автоматической редукции схемы, предусматривающие построение соответствующих базовых элементов (отображений) из явно указанных человеком производных элементов (множеств связей, атрибутов).

Именно базовые концепции ERM-модели фигурируют в правилах проверки схемы на непротиворечивость и правилах трансформации схемы в СУБД-ориентированную модель данных.

#### **Литература**

1. Бабанов А. М. Семантическая модель «Сущность – Связь – Отображение» // Вестник ТГУ. Управление, вычислительная техника и информатика. – 2007. – № 1. – С. 77–91.
2. Бабанов А. М. Формальная система теории семантически значимых отображений // Вестник ТГУ. – 2006. – № 290. – С. 261–263.
3. Бабанов А. М. Развитие формальной системы теории семантически значимых отображений // Вестник ТГУ. – 2006. – № 293. – С. 135–139.
4. Чен П. Модель «сущность – связь» – шаг к единому представлению о данных // СУБД. – 1995. – № 3. – С. 137–158.

### **ПОДХОД К МИГРАЦИИ ДАННЫХ, ХРАНИМЫХ В РЕЛЯЦИОННЫХ СУБД**

***К. С. Важдзева, И. А. Кудрявцев, Е. П. Яковлева***  
*Томский государственный университет*

На сегодняшний день отмечается появление новых и усложнение существующих бизнес-процессов. Это приводит к использованию более мощных по функциональности программных продуктов, что в свою очередь означает использование более сложных моделей данных. При эксплуатации программного обеспечения возможны ситуации, при которых необходим перенос всех данных или какой-то их части с возможным преобразованием данных из одного хранилища данных в другое. Такой перенос называют миграцией данных [3]. Этот процесс характеризуется высокой сложностью, обу-

словленной в первую очередь разнообразием моделей данных (МД) и мотивов выполнения миграции данных. Поэтому наряду с проектами по разработке программного обеспечения широко распространены и проекты по миграции данных.

Миграция данных – сложная задача, выполнение которой сопряжено с рисками. В связи с этим проводятся исследования подходов и разработка средств, позволяющих осуществлять этот процесс [4, 5]. Следует отметить, что отдельная ветвь исследований посвящена причинам неудачи проектов по миграции данных. Примерами таких причин являются отсутствие методологии, недооцененные масштабы проектов по миграции данных, некорректное использование инструментов, недостаточное внимание к обеспечению качества данных, недостаток или отсутствие необходимого опыта.

Предпосылки выполнения миграции данных формируют различные подходы к решению этой задачи, на основании которых развиваются методологии. Так, выполнение процесса миграции данных в значительной степени зависит от исходной и целевой МД. Такая зависимость определяет одно из направлений миграции данных. Еще одно направление этого процесса связано с причинами, инициирующими перенос данных. Стоит заметить, что сложность выполнения проектов, акцентированных на какое-то одно направление миграции данных, высока, а так как зачастую отдельный проект миграции данных относится сразу к нескольким направлениям, то сложность его реализации увеличивается.

В данной статье основное внимание сконцентрировано на процессах миграции, в которых хранилищем данных выступают реляционные СУБД (РСУБД). Выбор такого направления обусловлен широким распространением реляционных СУБД [1].

Можно выделить три основные причины выполнения миграции данных, хранимых в РСУБД:

- изменение схем баз данных [2];
- переход на другую РСУБД;
- перенос данных в другую БД без изменения схемы.

В связи с этим авторами предложен универсальный подход к процессу миграции данных, основанный на покортежном переносе данных. Идея покортежного переноса данных основывается на понятии зависимости кортежей. Независимым кортежем называют кортеж из отношения, не имеющего внешних ключей, либо кортеж из отношения, имеющего внешние ключи, со значениями внешних ключей, равными NULL. В противном случае кортеж называют зависимым. Суть покортежного переноса данных заключается в переносе сначала независимого кортежа, а потом всех зависимых от него кортежей. Стоит заметить, что зависимый кортеж может быть перенесен только в том случае, когда все кортежи, от которых он зависит, уже перенесены.

Этапами такого процесса миграции данных являются:

- получение информации о схеме исходной базы данных;
- получение информации о схеме целевой базы данных;
- определение последовательности переноса данных из исходной БД;
- определение правил переноса данных;
- выбор стратегии обработки ошибок, возникающих при переносе дан-

ных;

- перенос данных в соответствии с выбранной стратегией.

Предложенный процесс миграции данных можно проиллюстрировать диаграммой в нотации BPMN, изображенной на рис. 1.

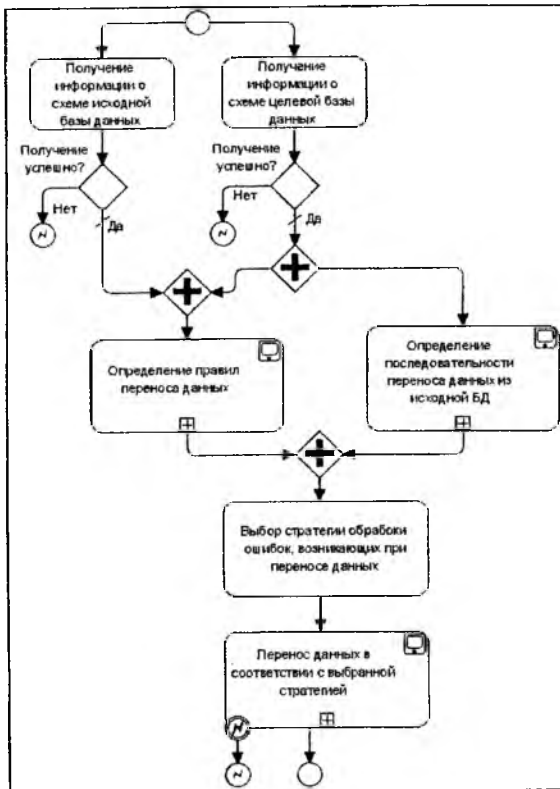


Рис. 1. Диаграмма процесса миграции данных в нотации BPMN

Выполнение первых двух этапов подразумевает представление схем баз данных в оперативной памяти в виде, содержащем необходимые для переноса данных сведения и скрывающем детали работы с конкретными СУБД от остальных шагов процесса миграции данных. Третий этап решает задачу корректного переноса данных из исходной БД без конкретизации места назначения. На следующем этапе задаются правила переноса данных – алгоритмические действия над кортежами, определяющие какие данные, как и куда переносить. На этом же этапе пользователю предоставляется возможность выполнения сравнения схем баз данных и просмотра его результатов. Пятый этап осуществляет настройку выполнения переноса данных выбором одной из возможных стратегий обработки ошибок. Примером стратегии является завершение процесса миграции при возникновении первой ошибки. Заключительный этап выполняет перенос данных из исходной БД в целевую.

Таким образом, предложенный подход описывает прозрачный механизм проверки корректности переносимых данных с точки зрения их целостности. Определяя правила, процесс миграции осуществляет перенос только необходимых данных с возможностью формирования целевых данных по описанным алгоритмам из исходных и новых данных. Также описанный выше подход позволяет абстрагироваться от конкретных РСУБД.

На данной стадии разработки предложенного подхода авторами были достигнуты следующие теоретические и практические результаты: проведен анализ изменений схем баз данных и изменений данных, хранимых в соответствующих БД, разработаны и реализованы представление схем баз данных в описанном выше виде, алгоритм определения последовательности переноса данных из исходной БД, алгоритм сравнения схем баз данных и алгоритм переноса данных в виде библиотек.

В заключение можно сказать, что, по мнению авторов, программное обеспечение, реализующее предложенный подход, будет востребовано в среде разработчиков приложений. Помимо своей основной функции – миграции данных, такое приложение может быть использовано для оценки различий между двумя схемами баз данных.

#### Литература

1. Дейт К. Введение в системы баз данных. – 7-е изд.: Пер. с англ. – М.: Вильямс, 2001. – 1072 с.
2. Эмблер С., Садаладж П. Рефакторинг баз данных: эволюционное проектирование: Пер. англ. – М.: Вильямс, 2007. – 672 с.
3. Chester B. Data migration 101 // AIIIM E-DOC. – 2006. – Vol. 20, Issue 1. – P. 10.
4. Hudicka J. An overview of data migration methodology [Electronic resource] // URL: [http://www.dulcian.com/Articles/Overview\\_Data\\_Migration\\_Methodology.htm](http://www.dulcian.com/Articles/Overview_Data_Migration_Methodology.htm) (reference data 05.09.2010).
5. Scott V. Extraction, transformation and load issues and approaches [Electronic resource] // URL: <http://www.tdan.com/view-articles/4839> (reference data 01.09.2010).