

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

УДК 004.652.8

А.М. Бабанов

СЕМАНТИЧЕСКАЯ МЕТОДИКА ПРОЕКТИРОВАНИЯ БД И ЕЕ ПЕРСПЕКТИВЫ, ОТКРЫВАЮЩИЕСЯ С ПРИМЕНЕНИЕМ ERM-МОДЕЛИ ДАННЫХ

В статье обсуждаются преимущества семантической методики проектирования БД, определяются ее этапы и основные факторы, влияющие на успех проектирования. Исходя из анализа факторов, предлагается идеальная схема применения семантической методики и на конкретном примере показывается, что использование ERM-модели позволяет существенно приблизиться к этому.

Ключевые слова: *семантическая методика, проектирование БД, семантическая модель данных, ERM-модель.*

Как только появились первые СУБД, люди задумались над проблемой проектирования эффективной схемы для каждого случая применения технологии БД. Достаточно стройной и конструктивной является классическая методика декомпозиции отношений, предложенная для реляционной модели данных [1 – 4].

Однако трудно не согласиться с мнением многих авторов [3, 4], утверждающих, что практическое ее использование осложняется такими факторами, как:

- нетрадиционный для большинства людей и весьма нетривиальный способ восприятия и формализации предметной области (ПрО);
- практическая неприменимость для сложных ПрО;
- неоднозначность решения проблемы проектирования, приводящая к прямому перебору многочисленных вариантов схемы в поисках наиболее подходящего.

Непрекращающиеся поиски решения проблемы проектирования БД привели к появлению семантических моделей данных, которые обеспечивают высокоуровневое формальное представление ПрО. Они должны были устранить первые два недостатка классической методики – неестественность представлений для человека и неприменимость для сложных ПрО.

Но, поскольку коммерческие СУБД не собираются поддерживать напрямую семантические модели данных, необходима трансляция схем БД с языка этих моделей на язык СУБД и в первую очередь – реляционных. Примерно так, практически с первых публикаций по ER-модели [5] стала вырисовываться основная схема семантической методики проектирования реляционных БД.

1. Этапы семантической методики

О семантической методике можно говорить в узком и широком смысле. Если рассматривать только те задачи проектирования, которые решает классическая реляционная методика, то последовательность шагов семантической методики выглядит следующим образом:

- проектирование семантической схемы ПрО с использованием той или иной семантической модели данных;
- перевод полученной схемы в реляционную модель с применением подходящего набора правил трансформации и получение множества предварительных отношений;
- проверка полученных отношений на удовлетворение требований нормальных форм и их дальнейшая нормализация методом декомпозиции.

Все семантические методики действуют по одной и той же схеме и отличаются лишь используемыми моделями и правилами трансляции схем.

Если рассматривать задачу проектирования схемы БД для конкретной СУБД максимально широко, начиная с первоначальной постановки задачи и заканчивая полностью готовой схемой, реализованной в этой СУБД, можно говорить о трех дополнительных этапах:

- предшествующее собственно моделированию данных функциональное моделирование бизнес-процессов предприятия, для информационного обеспечения которых создается БД;
- повышение эффективности полученной логической схемы БД за счет внесения управляемой избыточности (денормализация);
- повышение эффективности схемы БД за счет физического проектирования, учитывающего особенности хранения данных на диске и методов доступа к ним в конкретной СУБД.

Ограничение на объем статьи не позволяет осветить должным образом эти вопросы, поэтому в дальнейшем речь пойдет о трех основных этапах методики. Отметим лишь, что передача логической схемы СУБД сопровождается созданием дополнительных элементов БД, таких, как динамические представления, триггеры, процедуры, функции и так далее.

2. Факторы, влияющие на результат проектирования

Успех применения семантической методики проектирования БД помимо знаний и умений человека определяют два фактора:

- мощность выразительных средств используемой семантической модели,
- детальность анализа семантической схемы БД в применяемых правилах трансляции.

Выбор неадекватных инструментов может полностью дискредитировать методику. Так, невыразительная ER-модель нотации IDEF1X вынуждает проектировщика еще на этапе анализа семантики ПрО принимать сугубо технические решения (например, сводить бинарное множество связей типа $M:N$ к дополнительному множеству сущностей и паре множеств связей). Это происходит в силу того, что эта модель отличается от реляционной модели, по сути, взаимно-однозначным переименованием понятий: «множество сущностей» – «отношение», «множество связей» – «ограничение ссылочной целостности», «атрибут множества сущностей» – «атрибут отношения». А это свидетельствует о том, что уровень ее «семантической» такой же, как и у реляционной модели.

Совмещение двух в принципе различных мыслительных процессов – выявления семантики ПрО и определения схемы БД на весьма абстрактном языке СУБД – еще полбеды. Хуже то, что принимаемые в ходе этого технические решения (например, введение дополнительного множества сущностей и бинарных множеств связей вместо множества связей степени больше двух) должны быть по-

хорошему проанализированы заново на этапе окончательного построения реляционной схемы. В противном случае можно получить не самую эффективную схему БД.

Что уж говорить о бизнес-правилах, которые потребуют процедурной (а не декларативной) реализации на языке СУБД. На этапе семантического анализа их в лучшем случае фиксируют в виде текстовых описаний. Естественно, никакие правила трансляции для таких бизнес-правил не предусматриваются. Дай бог, чтобы проектировщик не забыл о них на последнем этапе разработки БД и вручную создал необходимые для их проверки триггеры.

Для иллюстрации влияния детальности правил трансляции рассмотрим по одному правилу из трех разных наборов, разработанных для преобразования схемы из ER-модели Чена в реляционную модель.

1. Для каждого множества сущностей и каждого множества связей создать собственное отношение.

2. Бинарные множества связей типа 1:1 или 1:М представляются дублированием первичного ключа 1-отношения в М-отношение (1-отношение – это отношение, построенное для представления множества сущностей, возле которого на ребре роли указана 1. Для М-отношения – соответственно, М) [1].

3. Каждое бинарное множество связей типа 1:1, у которого класс принадлежности обоих сущностей является обязательным, и множества сущностей, в нем участвующие, заменяются в ER-схеме одним агрегированным множеством сущностей, с которым соединяются все другие множества связей, имевшиеся у двух исходных множеств сущностей. Вновь образованное множество сущностей порождает одно отношение в реляционной схеме [4].

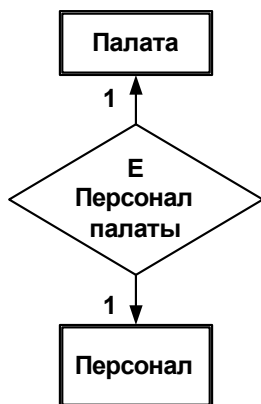


Рис. 1. Пример ER-диаграммы (нотация Чена)

Посмотрим, какие результаты будут получены с помощью этих правил для ER-схемы, приведенной на рис. 1.

Применение первого правила породит три отношения, по одному на каждое множество. По второму правилу в реляционной схеме будет построено два отношения. Этот прогресс (в смысле качества результата) объясняется дополнительным анализом максимальных кардинальных чисел отображений, определяемых множеством связей. И, наконец, третье правило приведет к одному отношению, наилучшим образом подходящему в нашем случае. Таким результатом мы обязаны дополнительному рассмотрению минимальных кардинальных чисел упомянутых отображений. Как видим, детальность анализа семантической схемы повышает эффективность результирующей схемы.

3. Идеальная схема применения семантической методики

Выбор семантической модели и набора правил преобразования схем во многом определяет успех проектирования. Конечно, хорошо, когда семантическая методика реализована в виде программного CASE-средства, но часто лучше провести проектирование на бумаге, используя мощную модель и правила, чем воспользоваться тривиальным автоматизированным помощником.

Идеальное применение семантической методики характеризуется следующими чертами.

1. На этапе первоначальной формализации описаний ПрО в рамках семантической модели схема данных должна вобрать в себя определения всех основных понятий ПрО и всех закономерностей взаимоотношений между ними. Для этого используемая семантическая модель должна быть в состоянии представить любые требования, предъявляемые к данным. Идеальная семантическая схема ПрО должна содержать ее полное формальное описание с тем, чтобы все последующие межмодельные преобразования схемы носили чисто синтаксический характер и не требовали повторного анализа семантики ПрО и работы с текстами на естественном языке.

2. Детальность применяемых в методике правил трансформации схемы должна быть такой, чтобы не оставить без внимания ни один элемент семантической схемы. Это в сочетании с доказанной надежностью и эффективностью правил позволит гарантировать идеальную со всех точек зрения СУБД-ориентированную схему.

Если к этому добавить средства автоматизации процесса проектирования в виде соответствующего CASE-инструмента, то получится идеальная реализация семантической методики. Подобная система, с одной стороны, облегчит построение семантической схемы ПрО, с другой – полностью автоматизирует процесс получения реляционной схемы и ее передачу целевой СУБД.

4. Ручное применение методики на примере ER-модели

Продemonстрируем применение семантической методики человеком без средств автоматизации. В примере используется классическая ER-модель Чена с расширением в виде специализаций [2] и средний по детальности анализа набор правил трансляции схем. В качестве предметной области выбраны родственные отношения между людьми. На рис. 2 приведена ER-диаграмма этой ПрО в нотации Чена (атрибуты и множества значений опущены).

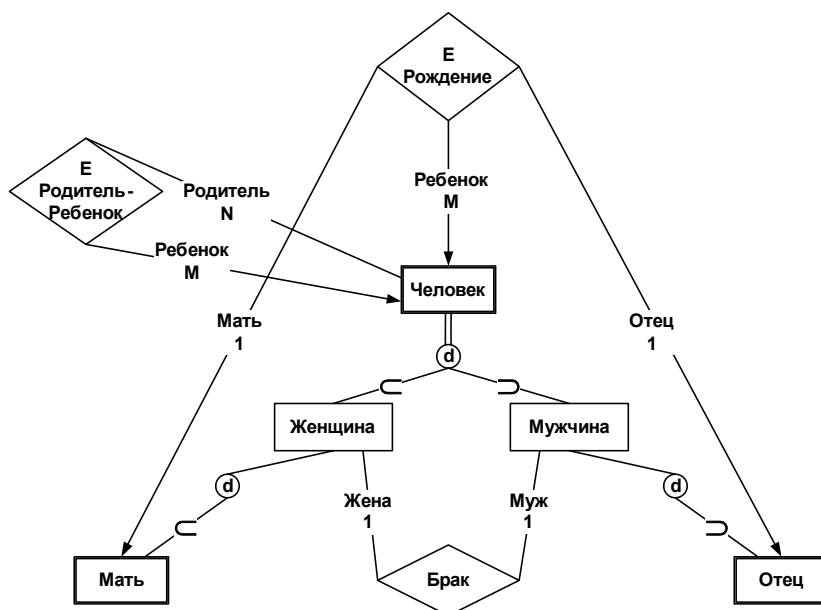


Рис. 2. ER-диаграмма родственных отношений (нотация Чена)

Если к этой ER-диаграмме добавить однозначные атрибуты множества сущностей ЧЕЛОВЕК (ФИО CHARACTER(50), ПОЛ CHARACTER(10)), мы получим исходную ER-схему этой ПрО.

Приведем один из традиционно используемых наборов правил трансформации схем ПрО из ER-модели Чена в реляционную модель.

1. Каждое множество сущностей представляется самостоятельным отношением, однозначные атрибуты множества сущностей становятся атрибутами отношения, ключи множества сущностей являются возможными ключами отношения; при необходимости в качестве первичного ключа отношения используется суррогатный атрибут.

Применяя это правило, мы получим первые пять отношений нашей схемы:

ЧЕЛОВЕК (ЧЕЛОВЕК_ID NUMBER(10, 0), ФИО CHARACTER(50), ПОЛ CHARACTER(10)),

ЖЕНЩИНА (ЖЕНЩИНА_ID NUMBER(10, 0)),

МУЖЧИНА (МУЖЧИНА_ID NUMBER(10, 0)),

МАТЬ (МАТЬ_ID NUMBER(10, 0)),

ОТЕЦ (ОТЕЦ_ID NUMBER(10, 0)).

2. Бинарные множества связей типа 1:1 и 1:М без атрибутов представляются дублированием первичного ключа 1-отношения в М-отношение.

В соответствии с этим правилом добавлен атрибут МУЖЧИНА_ID в отношение ЖЕНЩИНА: ЖЕНЩИНА (ЖЕНЩИНА_ID NUMBER(10, 0), МУЖЧИНА_ID NUMBER(10, 0)). Поскольку роли мужчин и женщин в браках абсолютно равноправны, можно было проделать симметричное изменение отношения МУЖЧИНА.

3. Бинарные множества связей типа М:М без атрибутов представляются самостоятельными отношениями, куда дублируются первичные ключи отношений, построенных для множеств сущностей.

По этому правилу построено отношение РОДИТЕЛЬ-РЕБЕНОК (РОДИТЕЛЬ_ID NUMBER(10, 0), РЕБЕНОК_ID NUMBER(10, 0)).

4. Множества связей с атрибутами представляются самостоятельными отношениями, куда дублируются первичные ключи отношений, построенных для множеств сущностей. Однозначные атрибуты множества связей становятся атрибутами этого отношения.

В нашей ER-схеме не было множеств связей такого рода.

5. Множества связей степени больше двух представляются самостоятельными отношениями, куда дублируются первичные ключи отношений, построенных для множеств сущностей. Однозначные атрибуты множества связей становятся атрибутами этого отношения.

По этому правилу построено отношение РОЖДЕНИЕ (РЕБЕНОК_ID NUMBER(10, 0), МАТЬ_ID NUMBER(10, 0), ОТЕЦ_ID NUMBER(10, 0)).

6. Каждый многозначный атрибут множества сущностей представляется отдельным отношением, куда дублируется первичный ключ отношения, построенного для множества сущностей; второй атрибут этого отношения предназначен собственно для значения.

В нашей ER-схеме атрибутов такого рода у множеств сущностей не было.

7. Каждый многозначный атрибут множества связей представляется отдельным отношением, куда дублируется первичный ключ отношения, построенного для множества связей; второй атрибут этого отношения предназначен собственно для значения.

В нашей ER-схеме атрибутов такого рода у множеств связей не было.

8. Каждая ветвь специализаций трактуется как обычное бинарное множество связей типа 1:1, которое преобразуется согласно второму правилу.

В соответствии с этим правилом первичные ключи отношений-родителей дублированы в дочерние отношения.

В результате применения указанных правил получились следующие отношения реляционной модели:

ЧЕЛОВЕК (ЧЕЛОВЕК_ID NUMBER(10, 0), ФИО CHARACTER(50), ПОЛ CHARACTER(10)),

ЖЕНЩИНА (ЖЕНЩИНА_ID NUMBER(10, 0), ЧЕЛОВЕК_ID NUMBER(10, 0), МУЖЧИНА_ID NUMBER(10, 0)),

МУЖЧИНА (МУЖЧИНА_ID NUMBER(10, 0), ЧЕЛОВЕК_ID NUMBER(10, 0)),

МАТЬ (МАТЬ_ID NUMBER(10, 0), ЖЕНЩИНА_ID NUMBER(10, 0)),

ОТЕЦ (ОТЕЦ_ID NUMBER(10, 0), МУЖЧИНА_ID NUMBER(10, 0)),

РОДИТЕЛЬ-РЕБЕНОК (РОДИТЕЛЬ_ID NUMBER(10, 0), РЕБЕНОК_ID NUMBER(10, 0)),

РОЖДЕНИЕ (РЕБЕНОК_ID NUMBER(10, 0), МАТЬ_ID NUMBER(10, 0), ОТЕЦ_ID NUMBER(10, 0)).

5. Автоматизированное применение методики на примере Oracle Designer

В качестве примера автоматизированного применения методики рассмотрим процесс проектирования реляционной схемы БД для той же ПрО с использованием интегрированной CASE-системы Oracle Designer. Ее ER-диагграммер использует ER-модель в нотации Баркера.

По сравнению с ER-моделью Чена в ней отсутствуют:

- множества связей степени больше двух,
- атрибуты множеств связей,
- многозначные атрибуты,
- атрибутивные отображения в декартово произведение множеств значений.

Давайте проанализируем, какие «технические решения» предстоит предпринимать проектировщику схемы БД, выбравшему в качестве семантической модели ER-модель Баркера.

Если ему необходимо представить множество связей степени n , придется использовать вспомогательное множество сущностей и n бинарных множеств связей.

В случае наличия атрибутов у множества связей универсальным решением является замена этого множества связей множеством сущностей и бинарными множествами связей. Однозначные атрибуты бывшего множества связей станут атрибутами нового множества сущностей.

Многозначные атрибуты множеств сущностей и множеств связей потребуют создания для каждого такого атрибута самостоятельного множества сущностей с атрибутом для значений и бинарного множества связей типа 1:М между множеством – владельцем многозначного атрибута и новым множеством сущностей. Очевидно, что в таком случае множество связей – владельца многозначного атрибута – необходимо превратить во множество сущностей.

В случае наличия атрибутивных отображений в декартово произведение множеств значений их надо представлять или элементарными атрибутами, или атрибутами-агрегатами, или и тем, и другим одновременно.

На рис. 3 приведена ER-схема нашей ПрО в нотации Oracle Designer, полученная из исходной ER-схемы в нотации Чена.

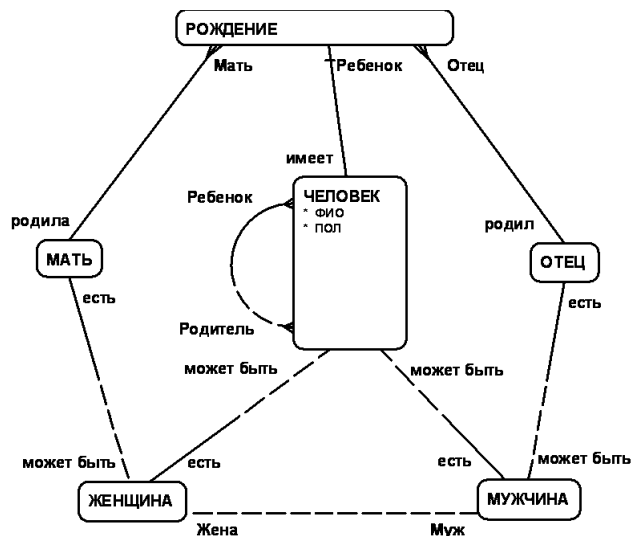


Рис. 3. ER-диаграмма родственных отношений (нотация Oracle Designer)

Далее Oracle Designer позволяет автоматически получить из этой схемы реляционную схему, приведенную на рис. 4.

Как видим, полученный результат ничем не отличается от предыдущего.

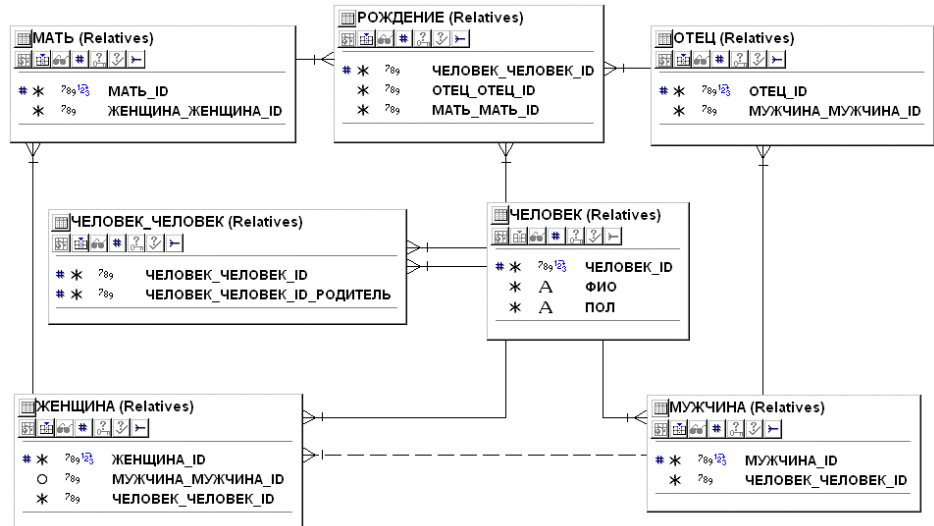


Рис. 4. Диаграмма реляционной схемы родственных отношений (нотация Oracle Designer)

6. ERM-модель

Целью научных исследований автора является совершенствование семантической методики проектирования схем БД. Результатом этих исследований является модель «Сущность – Связь – Отображение» (раннее название – «Объект – Отображение») [6], которая продолжает развитие ER-модели в сторону более детального

описания закономерностей ПрО. При этом сохраняются основные, практически востребованные конструкции, а также вводятся новые, не менее полезные элементы, существенно повышающие выразительные возможности модели. Англоязычное название модели – Entity-Relationship-Mapping Model или ERM-model.

Кратко перечислим нововведения.

1. Явно введено понятие «класс» как обобщение понятий «множество сущностей», «множество связей» и «множество значений». Именно классы образуют области определения и области значений отображений.

2. Специализации и категоризации подняты на уровень класса, что позволяет рассматривать иерархии обобщения не только множеств сущностей, но также и множеств связей, и множеств значений.

3. Введено понятие «отображение». Определение каждого отображения включает помимо прочего указание ролей образов и прообразов, а также классов, объекты которых играют эти роли.

4. Выделены частные случаи отображений – атрибутивные и реляционные отображения. Первые полностью соответствуют аналогам в модели Чена. Вторые определяются множествами связей, и в качестве образов и прообразов в них выступают сущности.

5. Между отображениями вводятся отношения следствия и эквивалентности.

6. Определена алгебра отображений – набор операций, задаваемых на множестве отображений. Каждая операция имеет одно или два отображения на входе и производит одно отображение на выходе.

7. Для специализаций введено понятие основания деления. В качестве такового выступают отображения.

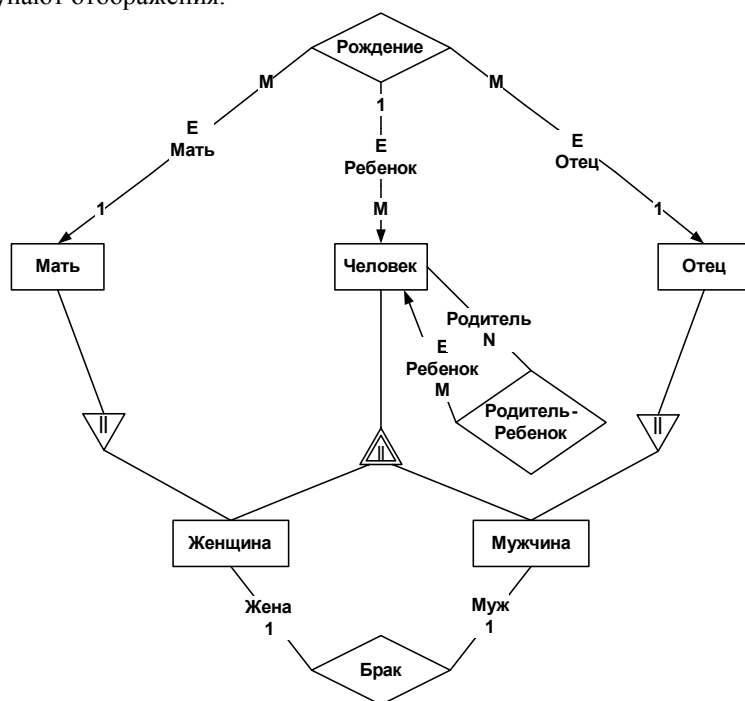


Рис. 5. ERM-диаграмма родственных отношений (расширенная графическая нотация)

Ключевым моментом, повлекшим за собой возможности более глубокого анализа ПрО, является дальнейшая декомпозиция понятия «множество связей» (в ЕР-модели при этом рассматриваются только роли сущностей) и выделение так называемых семантически значимых отображений. Каждое множество связей степени n определяет $2^n - 2$ отображений между множествами сущностей.

На рис. 5 приведена ERM-схема ПрО родственных отношений.

Методика трансляции ERM-схем в реляционную схему для рассматриваемого примера приводит к схеме с одним единственным отношением ЧЕЛОВЕК (ЧЕЛОВЕК_ID, ФИО, ПОЛ, ОТЕЦ_ID, МАТЬ_ID, СУПРУГ_ID) и следующими ограничениями целостности (нотация SQL):

ЧЕЛОВЕК_ID	NUMBER(10,0)	NOT NULL,
ФИО	CHARACTER(50)	NOT NULL,
ПОЛ	CHARACTER(10)	NOT NULL,
ОТЕЦ_ID	NUMBER(10,0)	NOT NULL,
МАТЬ_ID	NUMBER(10,0)	NOT NULL,
СУПРУГ_ID	NUMBER(10,0)	NULL,
PRIMARY KEY (ЧЕЛОВЕК_ID),		
UNIQUE (СУПРУГ_ID),		
FOREIGN KEY (ОТЕЦ_ID) REFERENCES ЧЕЛОВЕК (ЧЕЛОВЕК_ID),		
FOREIGN KEY (МАТЬ_ID) REFERENCES ЧЕЛОВЕК (ЧЕЛОВЕК_ID),		
FOREIGN KEY (СУПРУГ_ID) REFERENCES ЧЕЛОВЕК (ЧЕЛОВЕК_ID).		

Заключение

Применение ERM-модели в сочетании с детально проработанными правилами трансляции схем позволит приблизить семантическую методику проектирования БД к намеченному идеалу. А ее воплощение в CASE-инструменте сведет задачу проектирования к построению ERM-схемы и применению средств автоматического преобразования.

Причем в результирующей схеме будут представлены помимо прочего процедурные реализации (в виде триггеров) ограничений целостности, выходящих за рамки возможностей декларативных средств реляционной модели, а также определения динамических представлений для виртуальных отношений, соответствующих избыточным данным с точки зрения теории проектирования реляционных БД, но представляющих интерес с точки зрения пользователей.

ЛИТЕРАТУРА

1. Дейт К. Введение в системы баз данных. 7-е изд.: пер. с англ. М.: Вильямс, 2001. 1072 с.
2. Коннолли Т., Бегг К., Страчан А. Базы данных: проектирование, реализация и сопровождение. Теория и практика: пер. с англ. М.: Вильямс, 2000. 1120 с.
3. Цикритзис Д., Лоховски Ф. Модели данных: пер. с англ. М.: Финансы и статистика, 1985. 344 с.
4. Джексон Г. Проектирование реляционных баз данных для использования с микроЭВМ: пер. с англ. М.: Мир, 1991. 252 с.
5. Чен П. Модель «сущность – связь» – шаг к единому представлению о данных // СУБД. 1995. № 3. С. 137–158.
6. Бабанов А.М. Семантическая модель «Сущность – Связь – Отображение» // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2007. № 1. С. 77–91.