

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

INFORMATICS AND PROGRAMMING

Научная статья

УДК 004.652.8

doi: 10.17223/19988605/60/10

**Методика структуризации данных в семантических моделях
типа «Сущность–Связь»****Алексей Михайлович Бабанов***Томский государственный университет, Томск, Россия, babanov2000@mail.ru*

Аннотация. Строго определен понятийный базис модели «Сущность–Связь» (ER-модели), сформулирована проблема триализма, возникающая при представлении явлений предметной области (ПрО) в этой модели, предложены методика определения подходящих форм данных и правила структуризации для каждой из этих форм.

Ключевые слова: ER-модель; проектирование; схема данных; методика

Для цитирования: Бабанов А.М. Методика структуризации данных в семантических моделях типа «Сущность–Связь» // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2022. № 60. С. 93–101. doi: 10.17223/19988605/60/10

Original article

doi: 10.17223/19988605/60/10

Technique of data structuring in semantic models of the “Entity-Relationship” type**Alexey M. Babanov***Tomsk State University, Tomsk, Russian Federation, babanov2000@mail.ru*

Abstract. Database designing of complex corporate systems is not complete without the use of a semantic technique, and, therefore, one or another semantic model. Models of the “Entity-Relationship” type are the most widespread in this class. Most publications on the ER model assume that the designer will, on a whim, determine the appropriate forms of data representation. The problem is aggravated by the fact that one (and only one) of the three possible forms should be chosen for each phenomenon of the subject area.

If we take as a basis the terms and definitions of the author of the model P. Chen and look at them from the point of view of logic, we get the following consistent conceptual basis.

An entity is an abstract representation of a single application domain (AD) object. This abstraction is built on the basis of a certain notion about things, which is essential from the point of view of the AD problems. Each notion defines its own entity set, the intension of which is the content of the concept, and the extension is its scope.

A relationship is a single instance of a relation (in a logical sense) between entities. For relationship sets, not the notion about things is used (as with entities), but the notion about n things. A relationship set can be viewed as a mathematical relation defined on n entity sets in the general case, and a relationship as a tuple of this relation with n entities as elements.

Each entity in a relationship plays a specific role, a function that is common to all entities of that role.

Information about an object or a relationship between objects is obtained by observing or measuring the characteristics that are essential for a particular notion, and is expressed by a set of “attribute-value” pairs. Abstract values are classified purely syntactically into different value sets. Thus, a value set in this model is what is sometimes called a domain in other models – the area of admissible attribute values.

An attribute in the ER model is formally defined by Chen as a mapping that associates entities or relationships of some set with single values or tuples of values.

When choosing a data form for a single phenomenon, one should be aware of the possible triadism (the presence of three forms of representation in ER model) of the phenomena of the modeled world. The three forms of data are the three possible kinds of signs in this model. It is necessary to generate such data structures that would optimally take into account two mutually contradictory criteria – the informativeness of data required by business processes and the efficiency of their storage and manipulation. In the vast majority of data models and DBMS, when designing a database schema, it is necessary to make a choice in favor of only one of the possible forms for each phenomenon. The information needs of the AD business processes are decisive.

The procedures for designing entity sets and relationships are similar and consist of the steps of defining the corresponding notion, a common noun as its name, and extracting meaningful attribute mappings.

Since each relationship set corresponds to the notion about n objects, special attention must be paid to the construction of the necessary and sufficient system of entities that make up one relationship of each considered relationship sets. On the one hand, it must include all objects that determine the sense of the notion about n objects, on the other hand, it should not contain objects that can be excluded from the content of the concept without loss of information.

The second difficulty in designing relationship sets is to distinguish between the entities involved in a relationship and the values of the relationship's characteristics.

One of the important skills that a data schema designer should have is the ability to choose short but precise names for structural elements. Namely, by reading them in a diagram, any person should be able to recreate the semantic picture of the modeled world-concepts of interest about objects and n objects. In the ER model, schema elements such as entity sets, relationship sets, roles, and attributes can have semantically meaningful names. Value sets combine structurally uniform elements that do not have semantics; they have standard abstract names-strings, integers, etc.

The article did not focus on certain notations and tools of the ER model. The given definitions and methods can be successfully applied in each of them.

Keywords: ER model; design; data schema; methodology

For citation: Babanov, A.M. (2022). Technique of data structuring in semantic models of the “Entity-Relationship” type. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naja tehnika i informatika – Tomsk State University Journal of Control and Computer Science*. 60. pp. 93–101. doi: 10.17223/19988605/60/10

Проектирование баз данных сложных корпоративных систем не обходится без использования семантической методики, а значит, той или иной семантической модели. Наибольшее распространение в этом классе получили модели типа «Сущность–Связь». В большинстве публикаций по ER-модели предполагается, что проектировщик по наитию определит подходящие формы представления данных. Проблема усугубляется тем, что для каждого явления предметной области следует выбрать одну (и только одну) из трех возможных форм.

Проектирование семантических схем данных до сих пор остается скорее искусством, нежели наукой, поскольку, как правило, не регламентируется строгими определениями и методиками. Характерным является следующий взгляд на этот предмет: «...звучит просто, но в сложной системе может вызвать затруднения. Это то, что будет совершенствоваться только с практикой» [1].

Современные методики проектирования ER-схем в лучшем случае предлагают отталкиваться от грамматики естественных языков [1, 2]: «Стоит запомнить, что сущность – существительное, которое находится в прямоугольнике, а связь – глагол, который находится в ромбе» [2]. Встречаются и более «глубокие» идеи. «Компоненты ER можно приравнять к частям речи. Общее существительное – тип сущностей. Собственное существительное – сущность. Глагол – тип отношений. Прилагательное – атрибут типа сущностей. Наречие – атрибут типа отношений» [1]. Подобные методики не выдерживают критики даже в случае самых простых предметных областей (ПрО).

В статье строго определяются структурные понятия ER-модели, которые впоследствии используются для выбора формы представления явлений ПрО и построения семантической схемы данных. Предлагаются методики решения обеих этих задач.

1. Структурные понятия ER-модели

По поводу понятийного базиса ER-модели отсутствует единство взглядов. Одни авторы используют термин «сущность» и для обозначения знаков (единичных явлений), и для обозначения

типов (множеств знаков) [2, 3]. В таком случае рано или поздно возникает противоречие в понятийном базисе модели. Другие авторы пользуются двумя терминами: один – для знаков, другой – для типов [4–6]. В таком случае традиционно используют пары «экземпляр сущности – сущность» и «сущность – тип (множество, класс, набор) сущностей». Но и в этом случае не избежать противоречия, если отождествлять понятия «объект предметной области» и «сущность» [3, 4]. Если взять за основу термины и определения автора модели П. Чена [7] и посмотреть на них с точки зрения логики, то получится следующий непротиворечивый понятийный базис.

Сущность (англ. entity) – абстрактное представление единичного объекта ПрО. Эта абстракция строится на основании некоторого понятия о предметах, существенного с точки зрения задач ПрО. Любое понятие характеризуется содержанием (условием, истинность которого говорит о том, что объект подпадает под это понятие) и объемом (классом объектов, удовлетворяющих условию содержания). Содержание понятия предполагает наличие определенных характеристик, общих для всех объектов объема, возможно, различающихся значениями. Очевидно, что с точки зрения различных понятий можно построить несколько абстракций-сущностей одного и того же объекта. Например, один и тот же человек может быть и пациентом, и студентом одновременно. Таким образом термины «объект» и «сущность» не только не тождественны, между обозначаемыми ими элементами существует отношение типа «один-ко-многим». Каждое понятие определяет свое множество сущностей (англ. entity set), интенционалом которого является содержание понятия, а экстенционалом – его объем.

Связь (англ. relationship) – это единичный экземпляр отношения (в логическом смысле) между сущностями. Абсолютно все, что говорилось о сущности и множестве сущностей, можно сказать о связи и множестве связей. Разница заключается лишь в том, что для последних используется не понятие о предметах (как у сущностей), а понятие об n -ках предметов. Множество связей (англ. relationship set) можно рассматривать как математическое отношение, определенное в общем случае на n множествах сущностей, а связь – как кортеж этого отношения с n элементами-сущностями.

Каждая сущность в связи играет определенную роль (англ. role) – функцию, общую для всех сущностей этой роли. Таким образом, даже если в понятии об n -ках предметов неоднократно участвует одно и то же множество сущностей, в каждой связи сущности этого типа будут различаться своими ролями. В силу того что множество связей рассматривается как математическое отношение, порядок упоминания сущностей в связи является значимым, и поэтому для представления связей используются кортежи в смысле математики (упорядоченные последовательности). Однако указание ролей сущностей в связях делает соблюдение порядка сущностей необязательным, и поэтому в таком случае для представления связей используются уже множества пар «роль–сущность».

Информацию об объекте или взаимоотношении между объектами получают путем наблюдения или измерения характеристик, существенных для того или иного понятия, и выражают множеством пар «атрибут–значение». Абстрактные значения (англ. value) классифицируются чисто синтаксически в различные множества значений (англ. value set). Таким образом, множество значений в этой модели представляет собой то, что в других моделях иногда называют доменом, – область допустимых значений атрибутов.

Атрибут (англ. attribute) в ER-модели формально определяется Ченом как отображение, ставящее в соответствие сущностям или связям некоторого множества одиночные значения или кортежи значений [7]. Таким образом, областью определения атрибутного отображения может быть множество сущностей или множество связей, а областью значений – множество значений или Декартово произведение множеств значений. Причем это отображение не обязано быть функциональным, образ у одного прообраза может быть несколько. В таком случае говорят о многозначных атрибутах. Обратите внимание, что в этой модели в форме атрибутов представляются как свойства, так и характеристики сущностей и связей. У первых просто фиксирована область значений – множество истинных значений (Истина и Ложь), в то время как у вторых она произвольна.

Мы познакомились со структурными понятиями ER-модели. Заметим только, что множества сущностей, множества связей, роли, множества значений и атрибуты являются типами и составляют схему базы данных (БД) в этой модели, а сущности, связи и значения – суть знаки, и представляют

данные о явлениях ПрО (именно с этими тремя формами данных и связана проблема триализма, обсуждаемая далее).

Как видим, предлагаемые в ER-модели структурные понятия очень близки по смыслу общечеловеческим понятиям: сущность – предмет, связь – отношение, атрибут – свойство или характеристика. Надо только отдавать себе отчет, что происходит переход от интуитивно воспринимаемых человеком понятий к формальным понятиям со своими определениями и свойствами.

2. Проблема триализма данных

Явления ПрО представляются в памяти компьютера, и в частности в БД, в виде некоторым образом структурированных данных. Спроектированная структура данных должна быть, с одной стороны, эффективной (минимизировать расходы памяти и время отклика), с другой – удобной и понятной в использовании людьми. Но главное, структура данных должна отражать потребности бизнес-процессов ПрО, для информационного обеспечения которых создается хранилище. Заинтересованность людей, реализующих эти процессы, – главный критерий релевантности принимаемых при проектировании решений.

В частности, бизнес-процессы определяют представляющие интерес явления ПрО, данные о которых должны быть представлены в БД. Выбирая форму данных для единичного явления в случае ER-модели, следует помнить о возможном триализме (наличии трех форм представления) явлений моделируемого мира [4, 8]. Три формы данных – это три возможных вида знаков в этой модели. «Использование основных понятий “сущность”, “атрибут”, “связь” представляется весьма естественным, хотя при анализе предметной области правильный выбор понятий (что считать сущностями, что связями и что атрибутами сущностей) может вызвать затруднения» [8].

Решение этой задачи должно порождать структуры, оптимально учитывающие два взаимно противоречивых критерия – требуемую бизнес-процессами информативность данных и эффективность их хранения и манипулирования ими. Так, понятие «брак между людьми» можно мыслить как атрибут, множество связей или множество сущностей и, соответственно, представлять элементы этого понятия как значение, связь или сущность. В подавляющем большинстве моделей данных и СУБД, проектируя схему БД, необходимо сделать выбор в пользу только одной из возможных форм для каждого явления. При этом можно использовать следующие соображения (рассмотрим их на конкретном примере понятия «брак между людьми»). Решающими являются информационные потребности бизнес-процессов ПрО.

Если вам достаточно информации о том, состоит человек в браке или нет, можно представлять эти явления как значения атрибута «*Замужем / Женат?*» множества сущностей *ЧЕЛОВЕК*. Если вас к тому же интересует, с кем конкретно заключен брак, необходимо трактовать *БРАК* как множество связей между сущностями множества *ЧЕЛОВЕК*. Даже если вы захотите описать это явление с помощью тех или иных характеристик, вам не придется менять форму множества связей (в ER-модели Чена множества связей могут иметь атрибуты). А вот если вам понадобится представлять связи браков с другими явлениями, вам не обойтись без множества сущностей *БРАК*.

Как вы понимаете, мы перечислили варианты представления в порядке возрастания функциональности и информативности форм. Но это не означает, что все явления нужно трактовать как сущности. Бинаризация представлений (сведение структур данных к понятиям бинарной модели – множествам знаков и исключительно бинарным отношениям между ними), на первый взгляд, упрощает проектирование, но она скрадывает важную для проектирования эффективной схемы БД семантику ПрО, невосполнимую на последующих этапах. Поэтому альтернативы следует рассматривать именно в таком порядке (атрибут, множество связей, множество сущностей) и останавливать процесс выбора при достижении требуемой информативности.

Первые два варианта реализации явлений типа «брак» представлены в левой части рис. 1 (вверху – текстовые определения, внизу – ER-диаграммы) и, по-видимому, в пояснениях не нуждаются. В третьем случае (правая часть рисунка) предполагается, что помимо сведений о состоянии людей

в браке интерес представляет информация о том, кто рожден в момент зарегистрированного брака между родителями. Налицо необходимость в констатации наличия связей типа *РОЖДЕНИЕ В БРАКЕ* между множествами сущностей *ЧЕЛОВЕК* и *БРАК*.

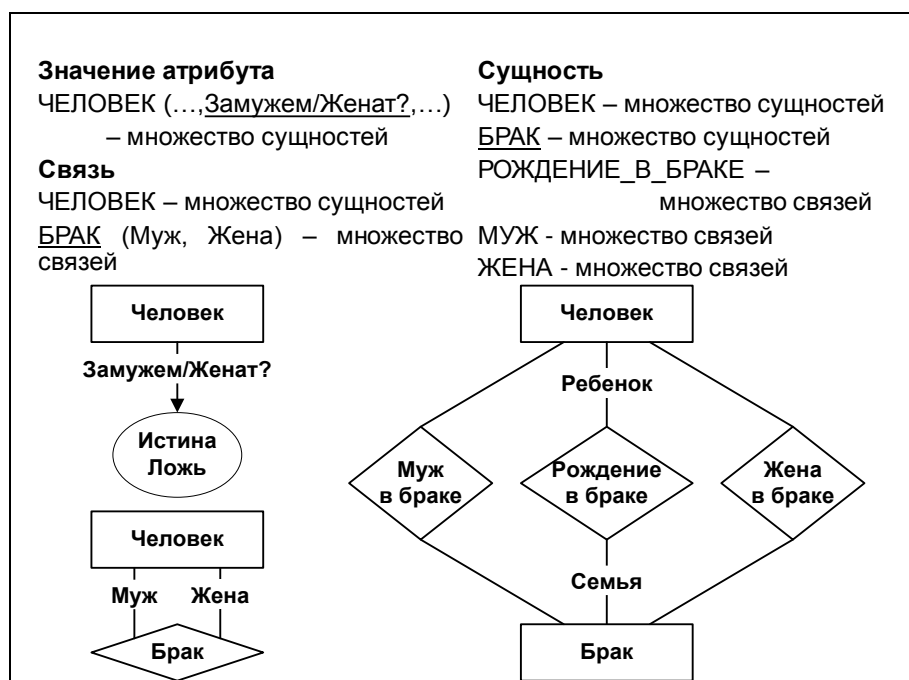


Рис. 1. Проблема триализма в ER-модели
 Fig. 1. The problem of trialism in the ER model

Обратите внимание на то, что по правилам структуризации ER-модели в связи могут вступать только сущности (а не связи или значения), и поэтому явления типа «брак» в последнем случае мы вынуждены представлять в виде множества сущностей. При этом роли *Муж* и *Жена* (второй случай) трансформируются в множества связей *МУЖ В БРАКЕ* и *ЖЕНА В БРАКЕ* (третий случай).

Именно этих соображений следует придерживаться при выборе подходящей структуры. И части речи в текстовых описаниях ПрО не являются определяющими в этом процессе. Получается, что в нашем примере существительное БРАК играло роль и прилагательного (атрибут множества сущностей), и глагола (множество связей), и существительного (множество сущностей).

В подавляющем большинстве опубликованных методик проектирования ER-схем данных [3, 4, 9–12] приводится три основных шага этого процесса:

- определение множеств сущностей;
- определение множеств связей;
- определение ограничений целостности.

Эта статья посвящена проектированию структур данных, поэтому далее будем говорить о первых двух этапах.

3. Проектирование множеств сущностей

Поскольку невозможно определить понятия для *n*-ок предметов, не задав первоначально понятия об этих предметах, сначала создаем структуры данных для последних – множества сущностей. После принятия решения о том, что некоторые однотипные явления будут представляться в форме множества сущностей, следует применить следующую процедуру для его проектирования.

1. Определить соответствующее понятие о предметах. Надо мысленно представить его содержание (правило принадлежности явлений ПрО объему этого понятия). Оно должно быть истинным для всех объектов, подпадающих под это понятие, и ложным в противном случае. Если есть возмож-

ность дать строгое логическое определение, следует это сделать. Но в подавляющем большинстве случаев формально определить понятие не удастся. В таком случае можно дать развернутое словесное описание.

2. Подобрать множеству сущностей существительное в качестве имени. Пространное вербальное определение, хоть оно и максимально точно отражает смысл понятия, неудобно использовать в качестве идентификатора соответствующей структуры. В идеале им должно быть общее существительное в единственном числе, как можно более близкое по значению. В глоссарии этому имени ставят в соответствие словесное описание. Таким образом, имя множества сущностей отражает смысл каждой его сущности.

3. Выделить для множества сущностей значащие (представляющие интерес с точки зрения бизнес-процессов) атрибутивные отображения. Определить для каждого:

- семантику (смысл предметной функции, которая ставит в соответствие каждой сущности некоторое(-ые) значение(-я));
- имя (общее существительное, отражающее смысл этой функции);
- область значений (множество значений, которые могут являться образами сущностей при этом отображении).

4. Проектирование множеств связей

После определения множеств сущностей можно приступить к созданию их агрегатов – множеств связей. Процедура проектирования множества связей выглядит следующим образом.

1. Определить соответствующее понятие об n -ках предметов.
2. Подобрать ему общее существительное в качестве имени.
3. Выделить для него значащие атрибутивные отображения. Определить для каждого:
 - семантику,
 - имя,
 - область значений.

Реализация этих задач во многом схожа с реализацией аналогичных задач предыдущего шага (проектирование множеств сущностей). Поэтому мы не стали повторяться. Однако с выполнением этого этапа связаны определенные сложности.

Поскольку каждому множеству связей соответствует понятие об n -ках предметов, особое внимание необходимо уделить построению необходимой и достаточной системы сущностей, составляющих одну связь каждого рассматриваемого множества связей. С одной стороны, в нее должны войти все объекты, определяющие смысл понятия об n -ках предметов, с другой стороны, в ней не должно быть объектов, которые можно исключить из содержания понятия без потери информации.

Следует руководствоваться следующими соображениями. В момент создания связи должны быть определены сущности для всех ролей без исключения. Не должно быть связей, в которых хотя бы одна сущность появляется позже, а значит, до поры до времени ссылка на нее в связи пуста. И в дальнейшем, пока существует связь, все сущности в ней не изменяются. Лучшее представление для связи – это ребро в экстенсionale БД. Связь степени n – это ребро с n концами, каждому из концов которого соответствует своя роль, и инцидентна одна вершина, представляющая сущность. В таком случае становятся понятными приведенные выше правила.

Проиллюстрировать сказанное можно попыткой применить форму множества связей для третьего случая нашего предыдущего примера (см. рис. 1). Кому-то может прийти в голову идея реализовать в этом случае тернарное множество связей *Брак* с ролями *Муж*, *Жена*, *Ребенок*. Как правило, люди вступают в брак, еще не имея детей. Необходимо создавать связь для фиксации явления «вступление в брак», но не хватает сущности в роли *Ребенок*. Теперь представьте себе, что появился ребенок, заполнилась роль *Ребенок*. Казалось бы, все хорошо. А что делать, когда у этой пары родился второй ребенок, – добавлять роль или создавать новую связь? Оба решения не оптимальны. Все эти практические сложности связаны с неудачно спроектированной структурой множества связей.

Вторая сложность проектирования множеств связей касается того, что следует отличать сущности, участвующие в связи, от значений характеристик связи. Для примера рассмотрим множество связей *ЭЛЕМЕНТ РАСПИСАНИЯ ЗАНЯТИЙ* (рис. 2). На первый взгляд система множеств сущностей (*ПРЕПОДАВАТЕЛЬ*, *ГРУППА*, *ПРЕДМЕТ*, *АУДИТОРИЯ*, *ДЕНЬ НЕДЕЛИ* и *ВРЕМЯ*), объекты которых образуют связи этого типа, безупречна. Действительно, клеточка расписания занятий связывает именно эти объекты.

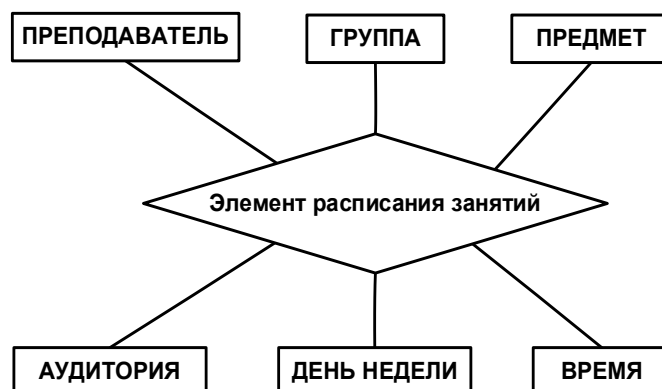


Рис. 2. Неправильно спроектированное множество связей
Fig. 2. Incorrectly designed relationship set

Однако объекты таких типов, как *АУДИТОРИЯ*, *ДЕНЬ НЕДЕЛИ* и *ВРЕМЯ*, скорее всего, не дотягивают до звания сущностей. Кроме собственно знаков этих объектов («104-я аудитория 2-го корпуса», «Понедельник» и «8:45») в БД мы не будем фиксировать о них больше ничего. У них нет характеристик, отношений с другими объектами, представляющих для нас интерес. Поэтому такие объекты, по сути, являются значениями соответствующих атрибутов множества связей – *Аудитория*, *День Недели* и *Время*. А значит, множество связей *ЭЛЕМЕНТ РАСПИСАНИЯ ЗАНЯТИЙ* на самом деле определено на множествах сущностей *ПРЕПОДАВАТЕЛЬ*, *ГРУППА*, *ПРЕДМЕТ* и имеет степень, равную трем (рис. 3).

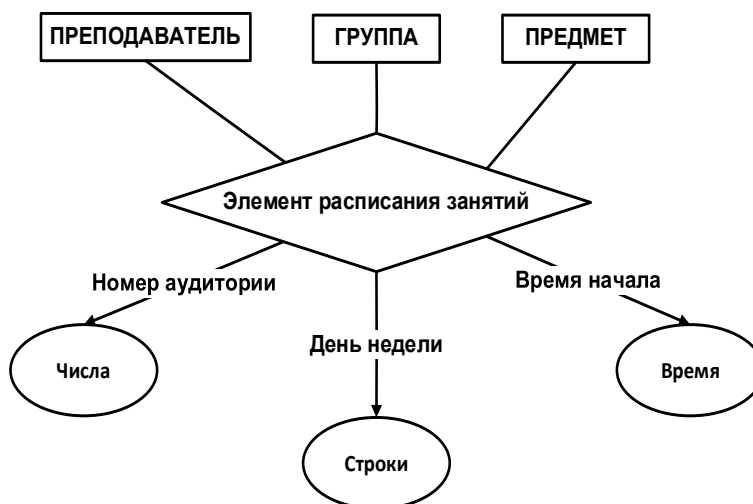


Рис. 3. Правильно спроектированное множество связей
Fig. 3. Correctly designed relationship set

Многие методики посоветовали бы в этом случае использовать для явления *ЭЛЕМЕНТ РАСПИСАНИЯ ЗАНЯТИЙ* такую форму, как множество сущностей. Тем самым мы опять осуществляем уже упомянутую бинаризацию структур и еще на семантическом уровне огрубляем картину мира, фиксированную в схеме данных. Но детальное рассмотрение этого вопроса относится, скорее, уже к ограничениям целостности данных и не является предметом этой статьи, посвященной структуризации.

5. Принципы именования элементов схемы

Одним из важных навыков, которыми должен обладать проектировщик схем данных, является умение подбирать краткие, но точные имена структурным элементам. Именно читая их в схеме, любой человек должен быть в состоянии воссоздать семантическую картину моделируемого мира – представляющие интерес понятия о предметах и *n*-ках предметов. В ER-модели семантически значимыми именами могут обладать такие элементы схемы, как множества сущностей, множества связей, роли и атрибуты. Множества значений объединяют структурно единообразные элементы, не имеющие семантики, они имеют стандартные абстрактные имена – строки, целые числа и т.д.

При проектировании ER-схемы следует придерживаться следующих принципов определения имен структурных объектов.

Классы знаков определяются соответствующими понятиями об индивидах (для множеств сущностей), понятиями об *n*-ках индивидов (для множеств связей) или понятиями о предметно-функциональных характеристиках (для атрибутивных отображений). Каждому такому понятию желательно подобрать краткое общее имя существительное, традиционно используемое для передачи семантики объектов класса (при необходимости можно использовать определения и дополнения). Если не удастся подобрать такого имени, которое бы точно и однозначно раскрывало содержание понятия, следует выбрать его из наиболее подходящих по смыслу и уточнить требуемую семантику в глоссарии в виде формального или развернутого словесного определения.

Как правило, с именами множеств сущностей, атрибутов и ролей проблем не бывает. В естественных языках для них уже придуманы и широко используются подходящие термины. Хуже обстоят дела с множествами связей. Очень немногие из них имеют традиционно используемые имена, определяющие семантику связи без предпочтения какой-то одной роли (хорошими примерами таких имен являются «Брак», «Рождение»). Действительно, эти слова отражают смысл именно ненаправленной связи, в отличие от неудачных имен «Муж» и «Родитель», которые больше подходят ролям этих связей. А теперь представьте себе, что вам необходимо определить в схеме бинарную связь между студентом и группой. Варианты имен *Студент группы* и *Студенческая группа* выделяют одну из ролей, вторую делая второстепенной. В таком случае вместо использования имени одной из ролей для множества связей можно применить комбинацию имен ролей, которые играют сущности в этих связях (например, «Группа–Студент»). С точки зрения точности передачи семантики ПрО и логической строгости этот вариант имени предпочтительней имени, в котором необоснованно выделена и использована только одна роль.

Заключение

В настоящей статье выполнено исследование путей перехода от интуитивного к строго научному подходу в области концептуального проектирования данных.

Предлагаются строгий понятийный базис модели, методика определения подходящих форм данных и правила структуризации данных в семантических моделях типа «Сущность–Связь». Также предложены правила создания компонентов схемы (множеств сущностей и множеств связей), которые определяют основные шаги структуризации.

На примерах показано, как на основе решения проблемы триализма можно выбрать для каждого явления ПрО подходящую структурную форму.

Список источников

1. Фролов И. ER-диаграмма – это... Описание, виды, правила построения. 2018. URL: <https://fb.ru/article/436948/er-diagramma---eto-opisanie-vidyi-pravila-postroeniya> (дата обращения: 12.01.2022).
2. Ефименко В. Учимся проектированию Entity Relationship – диаграмм. 2019. URL: <https://habr.com/ru/post/440556/> (дата обращения: 12.01.2022).
3. Watt A., Eng N. Database Design. 2nd ed. BCcampus, 2014. Viii, 136 p.

4. Швецов В. Базы данных. Первая стадия концептуального проектирования базы данных (концептуальное моделирование). URL: <https://intuit.ru/studies/courses/508/364/lecture/8647> (дата обращения: 12.01.2022).
5. Кара-Ушанов В.Ю. Модель «Сущность–Связь». Екатеринбург : Урал. федер. ун-т, 2017. 64 с.
6. Создание ER-Диаграмм // Информационные технологии. URL: <http://inf-teh-lotos.ru/sozdanie-er-diagramm> (дата обращения: 12.01.2022).
7. Chen P. The entity-relationship model – towards a unified view of data // *ACM Transactions on Database Systems*. 1976. V. 1 (1). P. 9–36.
8. Хрусталёв Е.Ю., Баранова Н.М. Интеллектуальные семантические модели для повышения качества образовательных и научно-исследовательских процессов // *Экономический анализ: теория и практика*. 2013. № 35 (338). С. 2–10.
9. Хусаинова Г.Я. Методика построения ER-диаграммы для базы данных // *NovaInfo*. 2017. № 76-1. С. 322–327.
10. Nguyen Kim Anh. Data Modeling Using Entity-Relationship Model. 2009. URL: https://cnx.org/contents/aM2VU_eRT@1/Data-Modeling-Using-Entity-Relationship-Model (дата обращения: 12.01.2022).
11. Tsichritzis D., Lochovsky F. Data Models. Prentice-Hall, Inc., 1982. 400 p.
12. Бабанов А.М. Семантическая методика проектирования БД и ее перспективы, открывающиеся с применением ERM-модели данных // *Вестник Томского государственного университета. Управление, вычислительная техника и информатика*. 2011. № 3 (16). С. 58–66.

References

1. Frolov, I. (2018) *ER-diagramma – eto... Opisanie, vidy, pravila postroeniya* [ER diagram is... Description, types, construction rules]. [Online] Available from: <https://fb.ru/article/436948/er-diagramma---eto-description-vidyi-pravila-postroeniya> (Accessed: 12th January 2022).
2. Efimenko, V. (2019) *Uchimsya proektirovaniyu Entity Relationship – diagramm* [Learning to design Entity Relationship diagrams]. [Online] Available from: <https://habr.com/ru/post/440556/> (Accessed: 12th January 2022).
3. Watt, A. & Eng, N. (2014) *Database Design*. 2nd ed. BCcampus.
4. Shvetsov, V. (n.d.) *Bazy dannykh. Pervaya stadiya kontseptual'nogo proektirovaniya bazy dannykh (kontseptual'noe modelirovanie)* [Databases. The first stage of the conceptual design of the database (conceptual modeling)]. [Online] Available from: <https://intuit.ru/studies/courses/508/364/lecture/8647> (Accessed: 12th January 2022).
5. Kara-Ushanov, V.Yu. (2017) *Model' "Sushchnost'–Svyaz"* [Model "Entity - Relationship"]. Yekaterinburg: Ural Federal University.
6. Inf-teh-lotos.ru. (n.d.) *Sozdanie ER-Diagramm. Informatsionnye tekhnologii* [Information technology. Creation of ER Diagrams]. [Online] Available form: <http://inf-teh-lotos.ru/sozdanie-er-diagramm> (Accessed: 12th January 2022).
7. Chen, P. (1976) The entity-relationship model – towards a unified view of data. *ACM Transactions on Database Systems*. 1(1). pp. 9–36. DOI: 10.1145/320434.320440
8. Khrustalev, E.Yu. & Baranova, N.M. (2013) Intelligent semantic models for improving the quality of educational and research processes. *Ekonomicheskii analiz: teoriya i praktika – Economic Analysis: Theory and Practice*. 35(338). pp. 2–10.
9. Khusainova, G.Ya. (2017) Metodika postroeniya ER-diagrammy dlya bazy dannykh [Technique for constructing an ER diagram for a database]. *NovaInfo*. 76-1. pp. 322–327.
10. Nguyen Kim Anh. (2009) *Data Modeling Using Entity-Relationship Model*. [Online] Available from: https://cnx.org/contents/aM2VU_eRT@1/Data-Modeling-Using-Entity-Relationship-Model (Accessed: 12th January 2022).
11. Tsichritzis, D. & Lochovsky, F. (1982) *Data Models*. Prentice Hall Inc.
12. Babanov, A.M. (2011) Semantic method of database designing and its prospects opening with application of the erm data model. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naya tekhnika i informatika – Tomsk State University Journal of Control and Computer Science*. 16(3). pp. 58–66.

Информация об авторе:

Бабанов Алексей Михайлович – доцент, кандидат технических наук, доцент кафедры программной инженерии Института прикладной математики и компьютерных наук Национального исследовательского Томского государственного университета (Томск, Россия). E-mail: babanov2000@mail.ru

Автор заявляет об отсутствии конфликта интересов.

Information about the author:

Babanov Alexey M. (Candidate of Technical Sciences, Associate Professor, National Research Tomsk State University, Tomsk, Russian Federation). E-mail: babanov2000@mail.ru

The author declares no conflicts of interests.

Поступила в редакцию 15.01.2022; принята к публикации 30.08.2022

Received 15.01.2022; accepted for publication 30.08.2022