Лекции по курсу «Модели данных»

Литература:

Основная

|  |  |
| --- | --- |
| 1. | **Григорьев Ю.А., Ревунков Г.И.**  Г 82 Банки данных: Учеб. для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. – 320 с. (Сер. Информатика в техническом университете). |
| 2 |  |
| 3 |  |

К лабораторному практикуму

|  |  |
| --- | --- |
| 1 |  |

Лекция 1. Методические указания к лабораторным работам.

Лабораторные работы по курсу «Модели данных» предназначены для конструктивного освоения практической методики проектирования локального банка данных. Эта методика включает в себя одну из возможных методик инфологического подхода для проектирования модели предметной области – а именно методику применения модели «Сущность – связь», а также методику отображения спроектированной инфологической модели предметной области в структуру базы данных реляционной СУБД “MS ACCESS”, приемы программирования запросов к базе данных и других программных приложений на основе инструментов СУБД “MS ACCESS”.

**Формулирование сущностей**

Вначале необходимо сформулировать сущности, т.е. необходимо указать те типы объектов (т.е. наборы подобных объектов) ПО, о которых в системе должна накапливаться информация. В отдельных случаях это может оказаться затруднительным, так как некоторый фрагмент информации может быть представлен любым из типов конструктивных элементов (сущность, атрибут или связь). Например, тот факт, что конкретная деталь используется при сборке конкретного изделия, может быть выражен либо как связь Входит в состав между сущностями Деталь и Изделие, либо как атрибут Имеет в составе деталь для сущности Изделие, либо как сущность Схема сборочного состава и т.п.

В этих случаях рекомендуется проработать несколько вариантов моделей представления предметной области и отобрать тот, который окажется более гибким с точки зрения представления информации (т.е. позволяет представлять не только всю порцию некоторой информации, но и ее отдельные фрагменты). Такой подход расширяет возможности совместного использования данных в базе различными пользователями и закладывает основы для обеспечения долгосрочной гибкости системы при удовлетворении их информационных потребностей.

**Пример.** Пусть в некотором локальном представлении выполняется описание поставок товаров на склад. Предполагается, что в одной поставке участвует только один поставщик, поставляя только один вид товара. При этом поставщик может участвовать в нескольких поставках. Для описания можно воспользоваться, например, только двумя основными конструкциями (сущность и атрибут). Введем в рассмотрение сущность Поставка и выполним ее описание с помощью следующих атрибутов:

Индекс поставки;

Индекс поставщика;

Адрес поставщика;

Индекс товара;

Название товара;

Количество поставленного товара;

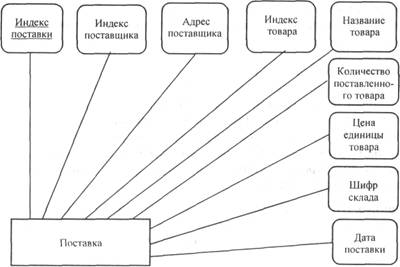
Цена единицы товара;

Шифр склада;

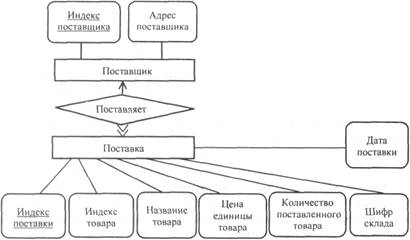
Дата поставки.

Графическая диаграмма локального представления приведена на рис. 3.1. Такая модель обладает определенными недостатками. С ее помощью нельзя представить такую порцию информации, как информация об отдельном поставщике, который не выполняет поставок в настоящее время. Чтобы эту порцию информации можно было представлять, необходимо ввести в модель сущность Поставщик, назначить ей соответствующие атрибуты, связать с сущностью Поставка и удалить избыточные элементы (рис. 3.2.).

При таком представлении всегда можно определить, какой конкретно поставщик выполнил поставку, используя для этого связь Поставляет между сущностями Поставщик и Поставка, т.е. в информационном плане данная модель сохраняет все возможности предыдущей модели. Но при этом она является более универсальной с точки зрения информационного представления, так как представляет еще информацию и об отдельных поставщиках независимо от того, выполняли ли они уже поставку товаров или еще нет.

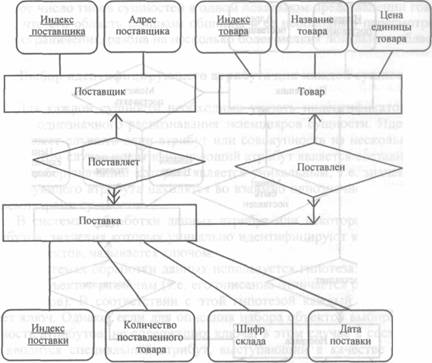


**Рис. 3.1.** Графическая диаграмма локального представления данных



**Рис. 3.2.** Введение в модель сущности ПОСТАВЩИК

Однако полученный вариант не позволяет представлять информацию об отдельных товарах, если они отсутствуют в поставках. Чтобы такие порции информации можно было в модели представлять, необходимо ввести в модель сущность Товар и выполнить аналогичные процедуры построения, как и для сущности Поставщик (рис. 3.3.). Данный вариант заключает в себе возможности предыдущих вариантов и, кроме того, позволяет представлять информацию об отдельном товаре независимо от того, присутствовал он в поставках или нет. Однако в этом варианте нельзя представить такую информацию, как «какие товары может поставлять отдельный поставщик» и «какие поставщики могут поставлять данный товар». Чтобы реализовать в модели возможность представления подобной информации, необходимо организовать соответствующие связи между сущностями Поставщик и Товар (рис. 3.4).

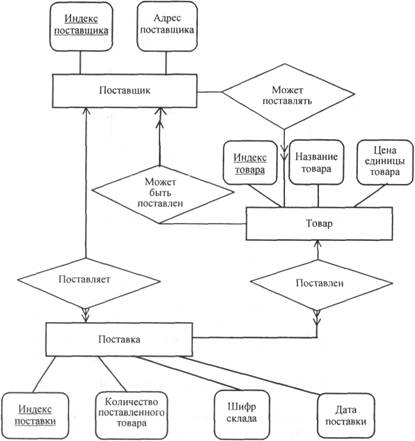


**Рис. 3.3.** Введение в модель сущности ТОВАР

Для локального представления, рассмотренного в примере, выбираем последний вариант модели, поскольку он оказался более гибким для представления информации, т.е. позволяет представлять не только информацию о поставках, но также ее отдельные фрагменты: о поставщиках, товарах, возможностях поставщиков, распределении поставщиков по видам товаров. Следовательно, БД, реализующая это представление, окажется более гибкой в обработке данных и будет обладать большими возможностями по обработке произвольных запросов.

Итак, для данного локального представления целесообразно сформулировать такие сущности, как Поставка, Поставщик, Товар.

Каждой выбранной сущности должно быть присвоено четкое наименование. Желательно, чтобы оно отражало смысловое содержание вводимого понятия. Расплывчатые наименования, наличие синонимов (одно и то же понятие имеет различные наименования) и омонимов (различные понятия имеют одно и то же наименование) приводят к ошибкам в проектировании и являются недопустимыми.



**Рис. 3.4.** Организация связей между сущностями

Обобщение категорий сущностей на этом шаге обычно не выполняется. При моделировании локального представления необходимо выполнить распознавание этих категорий и представить каждую в виде самостоятельной сущности. Распознавание выполняется с использованием концепции типа или роли. Например, содержание сущности Учащийся вуза может быть разделено по категориям типов: Студент, Аспирант, Слушатель факультета повышения квалификации и т.д. (Обобщение этих типов в родовую сущность Учащийся вуза выполняется на этапе объединения локальных представлений).

Как уже отмечалось, общее количество сформулированных сущностей в отдельном локальном представлении должно быть ограниченным. 94

Большое число типов сущностей в одном локальном представлении говорит о том, что его область слишком обширна и ее необходимо пересмотреть с целью ограничения, разбив на несколько более мелких локальных областей.

**Выбор идентифицирующего атрибута для каждой сущности**

Для каждой сущности необходимо указать индентификатор, служащий для однозначного распознавания экземпляров сущности. Идентификатором может служить один атрибут или совокупность из нескольких атрибутов (в этом случае идентифицирующий атрибут является составным атрибутом), набор значений которых является уникальным, т.е. значения идентифицирующего атрибута находятся во взаимно однозначном соответствии с экземплярами сущности.

В системах обработки данных атрибут или некоторая совокупность атрибутов, значения которых уникально идентифицируют каждый объект в наборе объектов, называется ключом.

В системах обработки данных используется гипотеза: каждый объект в наборе объектов различим (т.е. его описание отличается от описания других объектов). В соответствии с этой гипотезой каждый набор объектов имеет ключ. Однако, если для описания набора объектов выбирается совокупность атрибутов, не содержащих ключа, в этом случае в состав атрибутов вводится специальный атрибут, выступающий в качестве ключа. Во многих случаях это некоторый последовательный номер (например, в АСУ СБЕРКАССА в качестве ключа для идентификации счета вводят атрибут Номер счета, представляющий собой следующий свободный порядковый номер в реестре счетов сберкассы).

У одного и того же набора объектов может оказаться несколько ключей. Один из них назначается первичным ключом набора объектов и служит в дальнейшем ключом соответствующего этому набору файла (его еще называют ключом записи, кортежа). Все остальные ключи набора в этом случае называются возможными ключами.

Выбор ключа является важным моментом в проектировании моделей данных. Это связано с тем, что, с одной стороны, ключ должен выполнять свою главную задачу – задачу однозначной идентификации, а с другой стороны, он должен включать в свой состав минимально необходимое (для выполнения задачи идентификации) количество атрибутов. На заключительных этапах проектирования БД для некоторых приложений может оказаться возможным уточнение второй задачи: из множества возможных ключей в качестве первичного выбирают тот, для хранения которого требуется меньший объем памяти ЭВМ.

Кроме понятий «ключ», «первичный ключ», «возможный ключ» существует еще понятие «составной ключ» – для обозначения ключа, состоящего из двух атрибутов и более.

**Назначение сущности описательных атрибутов**

Выделенным сущностям в дополнение к идентифицирующим атрибутам назначаются описательные атрибуты, которые описывают представляющие для приложений интерес свойства сущностей.

Например, сущность Служащий может иметь такие описательные атрибуты, как Дата рождения, Образование, Домашний адрес.

Спецификация атрибутов должна быть закончена указанием для каждого атрибута множества значений или домена, которые этот атрибут может принимать. Если это множество является бесконечным, то оно задается указанием типа принимаемых значений – алфавитно-цифровой, цельный, вещественный и т.п. – и диапазона значений для числовых величин либо количества символов для представления алфавитно-цифровых значений.

В системах обработки данных используется понятие «вторичный ключ» – это атрибут или совокупность атрибутов, которые идентифицируют не уникальный объект в наборе, а все объекты, имеющие определенные значения этих атрибутов, т.е. вторичный ключ позволяет при необходимости выделять из набора те объекты, которые обладают интересующими нас свойствами. Вторичный ключ еще называют ключом поиска. Ограничения на состав вторичных ключей определяются только логикой запроса. В качестве вторичного ключа может выступать любой атрибут или любая совокупность атрибутов, описывающих набор объектов.

**Спецификация связей**

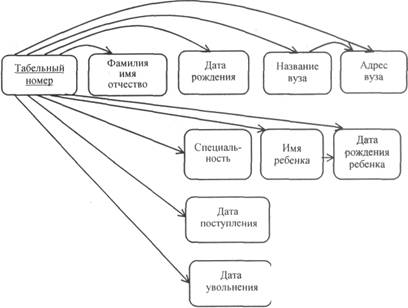
В локальном представлении выявляются зависимости между двумя (и более) сущностями. Устанавливается, какие связи являются важными, а какие – избыточными. Для выявленных связей определяются их характеристики, каждый тип связи «сущность–сущность» именуется.

Кроме спецификации связей типа «сущность–сущность» выполняется спецификация связей типа «сущность–атрибут» (т.е. указываются атрибут и сущность, для описания которых он используется), а также спецификация связей типа « атрибут–атрибут».

Связь типа «атрибут–атрибут» представляет собой отношение между атрибутами, которые относятся к одной и той же сущности или к одной и той же связи типа «сущность–сущность».

**Пример.** В локальном представлении для сущности Служащий были назначены следующие атрибуты: Табельный номер, Фамилия имя отчество, Дата рождения, Название вуза, Адрес вуза, Специальность, Имя ребенка, Дата рождения ребенка, Дата поступления, Дата увольнения.

Между атрибутами существуют определенные связи. Во-первых, поскольку все эти атрибуты описывают одну и ту же сущность Служащий, которая имеет идентифицирующий атрибут Табельный номер, следовательно, все атрибуты этой сущности имеют зависимость от идентифицирующего атрибута, т.е. при описании конкретных экземпляров сущности Служащий описательные атрибуты не могут принимать произвольные значения. Их значения зависят от значений идентифицирующего атрибута. Кроме этой зависимости среди атрибутов в рассматриваемом примере наблюдается и ряд других зависимостей между описательными атрибутами: значение атрибута Адрес вуза определяется значением атрибута Название вуза, значение атрибута Дата рождения ребенка – значением атрибута Имя ребенка.



**Рис. 3.5.** Диаграмма бинарных связей между сущностями

Эту информацию целесообразно представить в виде отдельной диаграммы бинарных связей между атрибутами. На рис. 3.5 связи представлены направленными дугами.

Анализ диаграммы связей между атрибутами сущности позволяет судить о том, насколько удачно был сделан выбор основных конструкций модели (в данном случае – сущность и атрибут) с точки зрения простоты представления смысловых связей между используемыми конструкциями при представлении конкретной порции информации.

Целесообразно стремиться к случаю, когда наблюдаются только зависимости описательных атрибутов от идентифицирующего и никаких иных зависимостей нет, т.е. зависимость «атрибут–атрибут» для сущности имеет вид, представленный на рис. 3.6.

С этой точки зрения в нашем примере целесообразно сущность Служащий представить с помощью следующих конструкций, представленных графической диаграммой (рис. 3.7). При этом исходная диаграмма бинарных связей типа «атрибут–атрибут» распадается на три диаграммы:

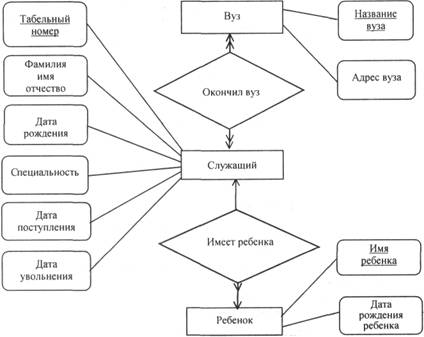
для сущности Служащий (рис. 3.8).

для сущности Вуз (рис. 3.9).

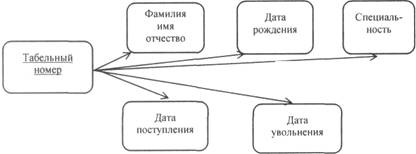
для сущности Ребенок (рис. 3.10).



**Рис. 3.6.** Зависимость «атрибут–атрибут»



**Рис. 3.7.** Пример графической диаграммы локального представления



**Рис. 3.8.** Диаграмма связей атрибутов сущности Служащий



**Рис. 3.9.** Диаграмма связей атрибутов сущности Вуз



**Рис. 3.10.** Диаграмма связей атрибутов сущности Ребенок

Возможны случаи, когда экземпляры сущности различают не по атрибутам, а по их связям с экземплярами сущности другого типа. Например, создается АС некоторого производственного объединения, включающего в состав несколько предприятий. Каждое предприятие имеет свою независимую систему присвоения табельных номеров служащим, причем структура табельных номеров на всех предприятиях одинакова, например пять десятичных знаков. В этом случае в БД, хранящей информацию о кадрах объединения, табельный номер потерял свою уникальность (может встретиться несколько одинаковых значений табельных номеров у различных сотрудников). Чтобы однозначно идентифицировать в БД описание конкретного служащего, придется помимо его табельного номера использовать также шифр предприятия, т.е. использовать связь между сущностями Служащий и Предприятие. При спецификации таких связей указывается, что они используются для идентификации сущностей и каких именно.

Моделирование локального представления заканчивается графическим оформлением всех выявленных сущностей, связей между ними и атрибутами и составлением всех перечисленных выше спецификаций.