

Лабораторная работа № 4.

Методология моделирования IDEF1X

Цель работы: изучить теоретические основы применения концептуальной модели «сущность – связь» для создания баз данных.

Теоретические сведения

Проектирование базы данных (БД) – одна из наиболее сложных и ответственных задач, связанных с созданием информационной системы (ИС). В результате её решения должны быть определены содержание БД, эффективный для всех ее будущих пользователей способ организации данных и инструментальные средства управления данными.

Процесс проектирования БД представляет собой последовательность переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели. В общем случае можно выделить следующие этапы проектирования: системный анализ, концептуальное проектирование, даталогическое (логическое) проектирование и физическое проектирование.

1. Системный анализ и словесное описание информационных объектов предметной области.

2. Концептуальное (инфологическое) проектирование – построение семантической модели предметной области, то есть информационной модели наиболее высокого уровня абстракции. Такая модель создаётся без ориентации на какую-либо конкретную СУБД и модель данных. Термины «семантическая модель», «концептуальная модель» и «инфологическая модель» являются синонимами. Кроме того, в этом контексте равноправно могут использоваться слова «модель базы данных» и «модель пред-

метной области» (например, «концептуальная модель базы данных» и «концептуальная модель предметной области»), поскольку такая модель является как образом реальности, так и образом проектируемой базы данных для этой реальности. Конкретный вид и содержание концептуальной модели базы данных определяется выбранным для этого формальным аппаратом. Обычно используются графические нотации, подобные ER-диаграммам.

Чаще всего концептуальная модель базы данных включает в себя:

- описание информационных объектов, или понятий предметной области и связей между ними.
- описание ограничений целостности, т.е. требований к допустимым значениям данных и к связям между ними.

3. Даталогическое (логическое) проектирование – создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных. Для реляционной модели данных даталогическая модель – набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

4. Физическое проектирование – создание схемы базы данных для конкретной СУБД. Специфика конкретной СУБД может включать в себя ограничения на именование объектов базы данных, ограничения на поддерживаемые типы данных и т.п. Кроме того, специфика конкретной СУБД при физическом проектировании включает выбор решений, связанных с физической средой хранения данных (выбор методов управления дисковой памятью, разделение БД по файлам и устройствам, методов доступа к данным), создание индексов и т.д.

1. Системный анализ предметной области

Первый этап проектирования БД это системный анализ , то есть подробное словесное описание объектов предметной области и реальных связей, которые присутствуют между описываемыми объектами.

1. Функциональный подход к проектированию БД.

Этот метод является наиболее распространённым. Он реализует принцип «от задач» и применяется в том случае, когда известны функции некоторой группы лиц и комплекса задач, для обслуживания информационных потребностей которых создаётся рассматриваемая БД.

2. Предметный подход к проектированию БД.

Предметный подход применяется в тех случаях, когда у разработчиков есть чёткое представление о самой ПО и о том, какую именно информацию они хотели бы хранить в БД, а структура запросов не определена или определена не полностью. Тогда основное внимание уделяется исследованию ПО и наиболее адекватному её отображению в БД с учётом самого широкого спектра информационных запросов к ней.

Чаще всего на практике рекомендуется использовать некоторый компромиссный вариант, который, с одной стороны, ориентирован на конкретные задачи или функциональные потребности пользователей, а с другой стороны, учитывает возможность наращивания новых приложений.

Системный анализ должен заканчиваться подробным описанием информации об объектах предметной области, которая требуется для решения конкретных задач и которая должна храниться в БД, формулировкой конкретных задач, которые будут решаться с использованием данной БД с кратким описанием алгоритмов их решения, описанием выходных документов, которые должны генерироваться в системе, описанием входных до-

кументов, которые служат основанием для заполнения данными БД.

2. Инфологическое проектирование

Инфологическая модель применяется на втором этапе проектирования БД, то есть после словесного описания предметной области и постановки задачи. Инфологическая модель должна включать такое формализованное описание предметной области, которое будет «читабельно» не только для специалистов по базам данных, но и сторонних людей. Описание должно быть настолько емким, чтобы можно было оценить глубину и корректность проработки проекта БД, и не должно быть привязано к конкретной СУБД.

3. Даталогическое проектирование

Следующим шагом является выбор конкретной СУБД и отображение в ее среду спецификаций инфологической модели предметной области. Эту стадию называют даталогическим (логическим) проектированием БД.

ER-диаграмма (entity-relationship diagram, диаграмма «сущность-связь») – это графическое представление инфологической модели.

Основные элементы ER-моделей:

- сущности (объекты);
- атрибуты сущностей;
- ключ сущности;
- связи между сущностями.

Модель «сущность – связь» относится к концептуальным моделям. В современном своем варианте она также является и логической моделью данных. Модель «сущность – связь» чаще всего применяется для проектирования структур баз данных.

Важной особенностью модели «сущность – связь» является то, что по ней однозначно может быть построена физическая модель базы данных или ее схема.

Поэтому практически все системы автоматизированного проектирования баз данных (CASE – системы) используют данную модель. В результате создан язык моделирования ERD (Entity-Relationship Diagrams) – язык диаграмм «сущность – связь», применяемый в CASE – системах.

Первый вариант модели «сущность – связь» был предложен в 1976 г. Питером Ченом. В дальнейшем многими авторами были разработаны свои варианты подобных моделей (нотация Мартина, нотация IDEF1X, нотация Баркера и др.).

Кроме того, различные CASE – системы, реализующие одну и ту же нотацию, могут отличаться своими возможностями. По сути, все варианты диаграмм «сущность – связь» исходят из одной идеи – использовать графическое изображение сущностей предметной области, их свойств (атрибутов), и взаимосвязей между сущностями.

Модель «сущность-связь» не является строго формальной. Диаграммы ERD строятся согласно следующим определениям.

Сущность

Сущность – это класс однотипных объектов, информация о которых должна быть учтена в модели. Слово «класс» не имеет здесь отношения к объектно-ориентированному программированию. Однако, существует понятие подсущность. Подсущность наследует свойства сущности, которой она принадлежит.

Каждая сущность должна иметь наименование, выраженное существительным в единственном числе. Примерами сущностей могут быть такие классы объектов как «Студент», «Ад-

рес», «Человек». Каждая сущность в модели изображается в виде прямоугольника с наименованием.



Экземпляр сущности – это конкретный представитель данной сущности.

Например, представителем сущности «Человек» может быть «Иванов». Экземпляры сущностей должны быть различными, т.е. сущности должны иметь некоторые свойства, уникальные для каждого экземпляра этой сущности.

Атрибут сущности

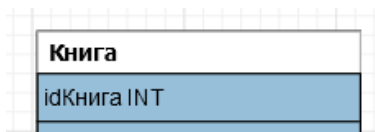
Атрибут сущности – это именованная характеристика, являющаяся некоторым свойством (параметром) сущности. Наименование атрибута должно быть выражено существительным в единственном числе (возможно, с характеризующими прилагательными). Примерами атрибутов сущности «Человек» могут быть такие атрибуты как «Фамилия», «Имя», «Отчество», «Номер паспорта» и т.п. Атрибуты изображаются в пределах прямоугольника, определяющего сущность.



Ключ сущности

Ключ сущности - это неизбыточный набор атрибутов, значения которых в совокупности являются уникальными для каждого экземпляра сущности.

Неизбыточность заключается в том, что удаление любого атрибута из ключа нарушается его уникальность. Сущность может иметь несколько различных ключей. Ключевые атрибуты изображаются на диаграмме в разделе под названием таблицы, как атрибут «idКнига».



Связь

Связь – это некоторая ассоциация между двумя сущностями. Одна сущность может быть связана с другой сущностью или сама с собою. Графически связь изображается линией, соединяющей две сущности.

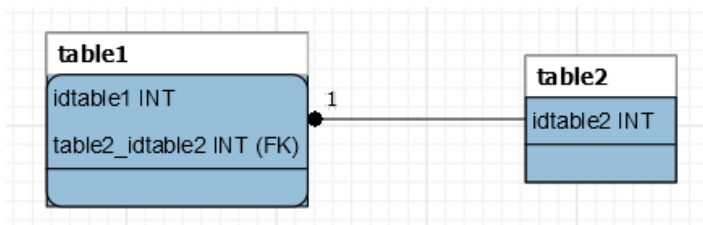
Каждая связь имеет собственное имя, два конца и два наименования. Наименование обычно выражается в неопреде-

ленной глагольной форме: «иметь», «принадлежать» и т.п. Каждое из наименований относится к своему концу связи. Наименования можно не указывать. Будучи объявленными, имя связи и наименования концов обрабатываются CASE – системой и применяются в алгоритмах обработки диаграмм «сущность – связь» для построения таблиц базы данных.

Каждая связь имеет два параметра: тип и модальность.

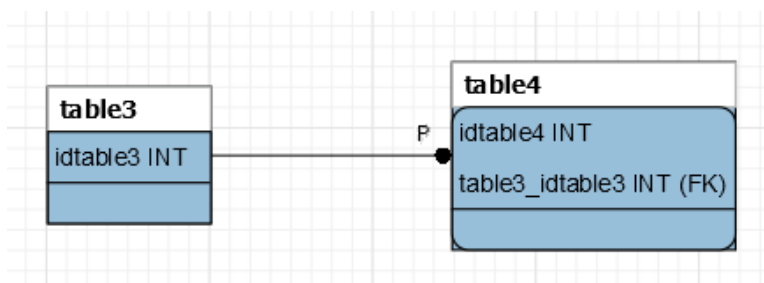
Связь типа один-к-одному

Связь типа один-к-одному означает, что один экземпляр первой сущности связан только с одним экземпляром второй сущности. Связь один-к-одному может соответствовать ситуации, когда две сущности отражают разные характеристики одного и того же объекта.



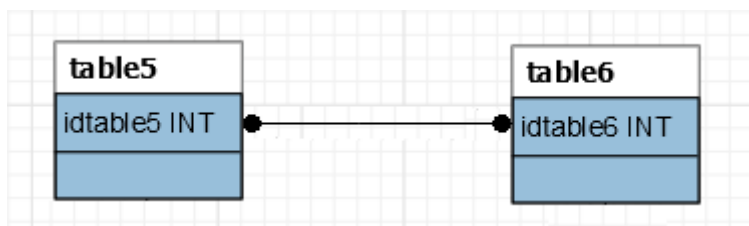
Связь типа один-ко-многим

Связь типа один-ко-многим означает, что один экземпляр первой сущности (левой) связан с несколькими экземплярами второй сущности (правой). Это наиболее часто используемый тип связи. Левая сущность (со стороны «один») называется родительской, правая (со стороны «много») – дочерней.



Связь типа много-ко-многим

Связь типа много-ко-многим означает, что каждый экземпляр первой сущности может быть связан с несколькими экземплярами второй сущности, и каждый экземпляр второй сущности может быть связан с несколькими экземплярами первой сущности.



Каждая связь может быть реализована в одном из двух видов: идентифицируемая связь (модальность «должен») и не-идентифицируемая связь (модальность «может»).

— — — — —
 Может

 Должен

Модальность «может» означает, что экземпляр одной сущности может быть связан с одним или несколькими экземплярами другой сущности, а может и не быть связан ни с одним экземпляром.

Модальность «должен» означает, что экземпляр одной сущности обязан быть связан не менее чем с одним экземпляром другой сущности.

4. Физическое проектирование

Основой для физического проектирования является схема БД, полученная на предыдущем этапе. Физическое проектирование заключается в увязке логической структуры БД и физической среды хранения с целью наиболее эффективного размещения данных.

Содержание лабораторной работы

В лабораторной работе требуется разработать диаграмму IDEF1X в соответствии с Вашим вариантом. Отчет о проделанной работе (пример оформления отчета по лабораторной работе смотри приложение 1) должен содержать: название и цель работы; номер варианта и описание задания; диаграмму соответствующую варианту; необходимые комментарии к диаграмме и выводы по проделанной лабораторной работе.

Варианты заданий

Для выполнения работы варианты задания берутся из лабораторной работы №1.