

Методология IDEF1X.
Проектирование логической модели системы

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНОЙ РАБОТЕ
ПО КУРСУ: «ПРОЕКТИРОВАНИЕ АСОИУ»

Составители: А. В. Челпанов
И. М. Мамонов

Москва 2018

Методология IDEF1X. Проектирование логической модели системы / Сост.:
А. В. Челпанов, И. М. Мамонов: Метод. указания. — МАИ, 2018. — 14 с.

© А. В. Челпанов, И. М. Мамонов, составление 2018

© МАТИ, 2018

ВВЕДЕНИЕ

Данные методические указания предназначены для проведения лабораторной работы по курсу «Проектирование автоматизированных систем обработки информации и управления» со студентами специальности 230102 «Автоматизированные системы обработки информации и управления» на базе компьютерной лаборатории с IBM/PC совместимыми ЭВМ в среде операционной системы семейства Microsoft Windows. Для выполнения лабораторной работы необходимо провести установку и настройку системы моделирования CA ERwin Data Modeler компании Computer Associates.

В лабораторной работе излагаются основные теоретические положения, что не исключает изучение учебников и рекомендованных федеральным агентством образования учебных пособий.

Материал изложен в следующем порядке. Сначала указывается цель работы, постановка задачи и теоретические сведения, необходимые для понимания и проведения работы; затем приводится описание интерфейса системы, инструментария, необходимого для выполнения работы; далее излагается порядок проведения лабораторной работы и задания.

В конце указаний приведен перечень основных контрольных вопросов для подготовки к лабораторным работам и их последующей сдаче (защите) преподавателю. Некоторые вопросы, связанные с темой данной работы, могут быть и не отражены в ее описании, но все работы проводятся в компьютерной лаборатории после прочтения лекций по соответствующему разделу, таким образом, студент должен при подготовке к работе проработать лекционный материал для более глубокого освоения курса.

ЛАБОРАТОРНАЯ РАБОТА №2

МЕТОДОЛОГИЯ IDEF1X.

ПРОЕКТИРОВАНИЕ ЛОГИЧЕСКОЙ МОДЕЛИ СИСТЕМЫ

Цель работы:

- ознакомиться с технологией построения логической модели в ERWin;
- изучить методы определения ключевых атрибутов сущностей;
- освоить метод проверки адекватности логической модели;
- изучить типы связей между сущностями.

Первым шагом при создании логической модели БД является построение диаграммы ERD (Entity Relationship Diagram). ERD-диаграммы состоят из трех частей: сущностей, атрибутов и взаимосвязей. Сущностями являются существительные, атрибуты - прилагательными или модификаторами, взаимосвязи - глаголами.

ERD-диаграмма позволяет рассмотреть систему целиком и выяснить требования, необходимые для ее разработки, касающиеся хранения информации.

ERD-диаграммы можно подразделить на отдельные куски, соответствующие отдельным задачам, решаемым проектируемой системой. Это позволяет рассматривать систему с точки зрения функциональных возможностей, делая процесс проектирования управляемым.

Краткие сведения из теории

1. ERD-диаграммы

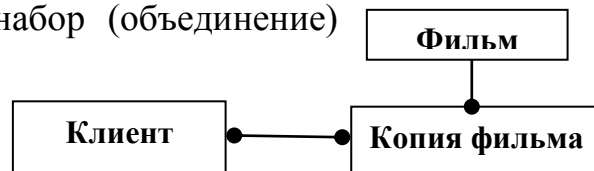
Как известно, основным компонентом реляционных БД является таблица. Таблица используется для структуризации и хранения информации. В реляционных БД каждая ячейка таблицы содержит одно значение. Кроме того, внутри одной БД существуют взаимосвязи между таблицами, каждая из которых задает совместное пользование данными таблицы.

ERD-диаграмма графически представляет структуру данных проектируемой информационной системы. Сущности отображаются при помощи прямоугольников, содержащих имя. Имена принято выражать существительными в единственном числе, взаимосвязи - при помощи линий,

соединяющих отдельные сущности. Взаимосвязь показывает, что данные одной сущности ссылаются или связаны с данными другой.

1.1. Определение сущностей и атрибутов

Сущность - это субъект, место, вещь, событие или понятие, содержащие информацию. Точнее, сущность - это набор (объединение) объектов, называемых экземплярами. В приведенном на рисунке примере сущность «клиент» представляет всех возможных клиентов. Каждый экземпляр сущности обладает набором характеристик. Так, каждый клиент может иметь имя, адрес, телефон и т. д. В логической модели все характеристики называются атрибутами сущности.



1.2. Логические взаимосвязи

Логические взаимосвязи представляют собой связи между сущностями. Они определяются глаголами; показывающими, как одна сущность соотносится к другой.

Некоторые примеры взаимосвязей:

- команда включает много игроков.
- самолет перевозит много пассажиров,
- продавец продает много продуктов.

Во всех этих случаях взаимосвязи отражают взаимодействие между двумя сущностями, называемое «один-ко-многим». Это означает, что экземпляр первой сущности взаимодействует с несколькими экземплярами другой сущности. Взаимосвязи отображаются линиями, соединяющими две сущности с точкой на одном конце и глаголом, располагаемым над линией.

Кроме взаимосвязи «один-ко-многим» существует еще один тип - это «многие-ко-многим». Этот тип связи описывает ситуацию, при которой экземпляры сущностей могут взаимодействовать с несколькими экземплярами других сущностей. Связь «многие-ко-многим» используют на первоначальных стадиях проектирования. Этот тип взаимосвязи отображается сплошной линией с точками на обоих концах.

Связь «многие-ко-многим» может не учитывать определенные

ограничения системы, поэтому может быть заменена на «один-ко-многим» при последующем пересмотре проекта.

1.3. Проверка адекватности логической модели

Если взаимосвязи между сущностями были правильно установлены, то можно составить предложения, их описывающие. Например, по модели, показанной на рис. 1, можно составить следующие предложения:

- «Самолет перевозит пассажиров»;
- «Много пассажиров перевозятся одним самолетом».

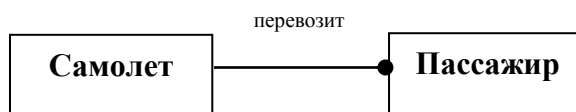


Рис. 1. Пример логической модели со связью «один-ко-многим»

Составление таких предложений позволяет проверить соответствие полученной модели требованиям и ограничениям создаваемой системы

2. Модель данных, основанная на ключах

Каждая сущность содержит горизонтальную линию, разделяющую атрибуты на две группы. Атрибуты, расположенные над линией, называются первичным ключом. Первичный ключ предназначен для уникальной идентификации экземпляра сущности.

2.1. Выбор первичного ключа

При создании сущности необходимо выделить группу атрибутов, которые потенциально могут стать первичным ключом (потенциальные ключи); затем произвести отбор атрибутов для включения в состав первичного ключа, следуя следующим рекомендациям:

- Первичный ключ должен быть подобран таким образом, чтобы по значениям атрибутов, в него включенных, можно было точно идентифицировать экземпляр сущности.
- Никакой из атрибутов первичного ключа не должен иметь нулевое

значение.

- Значения атрибутов первичного ключа не должны меняться. Если значение изменилось, значит, это уже другой экземпляр сущности.

При выборе первичного ключа можно внести в сущность дополнительный атрибут и сделать его ключом. Так, для определения первичного ключа часто используют уникальные номера, которые могут автоматически генерироваться системой при добавлении экземпляра сущности в БД. Применение уникальных номеров облегчает процесс индексации и поиска в БД.

Первичный ключ, выбранный при создании логической модели, может быть неудачным для осуществления эффективного доступа к БД и должен быть изменен при проектировании физической модели.

Потенциальный ключ, не ставший первичным, называется альтернативным ключом (Alternate Key). ERWin позволяет выделить атрибуты альтернативных ключей, и по умолчанию в дальнейшем при генерации схемы по этим атрибутам будет генерироваться уникальный индекс. При создании альтернативного ключа на диаграмме рядом с атрибутом появляются символы (AK).

Атрибуты, участвующие в неуникальных индексах, называются инверсионными входами (Inversion Entries). Инверсионные входы - это атрибут или группа атрибутов, которые не определяют экземпляр уникальным образом, но часто используются для обращения к экземплярам сущности. ERWin генерирует неуникальный индекс для каждого инверсионного входа.

При проведении связи между двумя сущностями в дочерней сущности автоматически образуются внешние ключи (Foreign Key). Связь образует ссылку на атрибуты первичного ключа в дочерней сущности, и эти атрибуты образуют внешний ключ в дочерней сущности. Атрибуты внешнего ключа обозначаются символами (FK) после своего имени.

3. Пример

Рассмотрим процесс построения логической модели на примере БД студентов системы «Успеваемость студентов в рамках вуза». Первым этапом является определение сущностей и атрибутов. В базе данных будут храниться записи о студентах, следовательно, сущностью будет студент (см. табл. 1).

Таблица №1. Атрибуты сущности «Студент»

Атрибут	Описание
Номер студента	Уникальный идентификатор
ФИО студента	Фамилия, имя и отчество студента
Дата рождения	Дата рождения студента
E-mail	Адрес электронной почты
Телефон	Номера телефонов студента (рабочий и домашний)
Адрес	Домашний адрес студента (прописка)
Кафедра	Название кафедры, на которой обучается студент
Специальность	Специальность, получаемая студентом по окончании учебного заведения
Специализация	Направление специальности, по которому обучается студент (Специалист, бакалавр, аспирант, магистр)
Основа обучения	Основа обучения студента (бюджетная, платная)
Форма обучения	Форма обучения студента (очная, заочная, очно-заочная)
Группа	Название группы, в которой обучается студент
Оценки	Список предметов с соответствующими оценками
Преподаватели	Список преподавателей дисциплин

В полученном выше списке существуют атрибуты, которые нельзя определить в виде одного поля БД. Такие атрибуты требуют дополнительных определений и должны рассматриваться как отдельные сущности, состоящие, в свою очередь, из атрибутов. К таковым относятся: оценки и преподаватели. Определим ниже их атрибуты (см. табл. 2,3).

Таблица №2. Атрибуты сущности «Оценки»

Атрибут	Описание
Номер предмета	Уникальный идентификатор
Название предмета	Название предмета за который студент получил оценку
Дата	Дата получения оценки
Оценка	Оценка, полученная студентом

Таблица №3. Атрибуты сущности «Преподаватели»

Атрибут	Описание
Номер преподавателя	Уникальный идентификатор
ФИО преподавателя	Фамилия, имя, отчество преподавателя
Должность	Должность преподавателя
Звание и степень	Ученое звание и ученая степень преподавателя

Составим ERD-диаграмму, определяя типы атрибутов и организуя связи между сущностями. Связь между сущностями «Студенты» и «Оценки» будет типа «один-ко-многим», а между сущностями «Оценки» и «Преподаватели» «Неидентифицирующая».

На полученной диаграмме рядом со связью отражается ее имя, показывающее соотношение между сущностями.

Следующим этапом при построении логической модели является определение ключевых атрибутов и типов атрибутов (см. рис. 2, табл. 4).

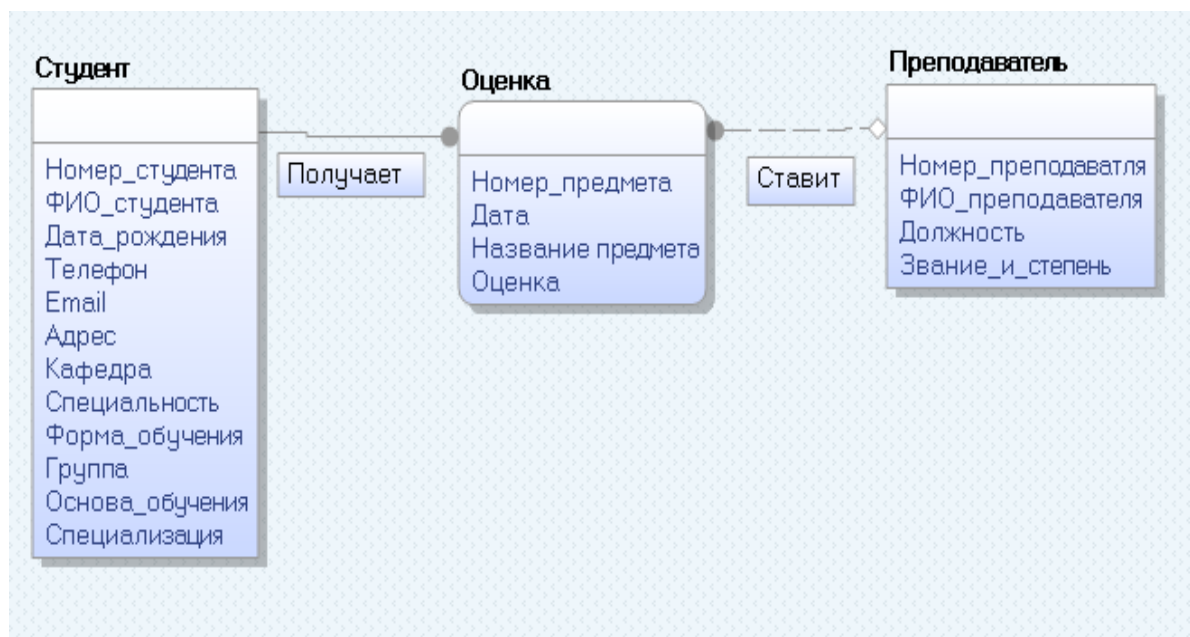


Рис. 2. Схема логической модели.

Таблица №4. Типы атрибутов

Атрибуты	Тип
Номер студента	Number
ФИО студента	String
Дата рождения	Datetime
E-mail	String
Телефон	String
Адрес	String
Кафедра	String
Специальность	String
Специализация	String
Форма обучения	String
Основа обучения	String
Группа	String
Номер преподавателя	Number
ФИО преподавателя	String
Должность	String
Звание и степень	String
Номер предмета	Number

Название предмета	String
Оценка	Number
Дата	Datetime

Выберем для каждой сущности ключевые атрибуты, однозначно определяющие сущность. Для сущности «Студент» и «Преподаватель» это будет соответствующий уникальный номер. Сущность «Оценка» определяется уникальным номером и датой, так как студент по одному предмету может иметь только одну оценку за один день.

Получим новую диаграмму, где ключевые атрибуты будут находится над горизонтальной чертой внутри рамки, изображающей сущность. При этом первичный ключ мигрирует в дочернюю сущность.

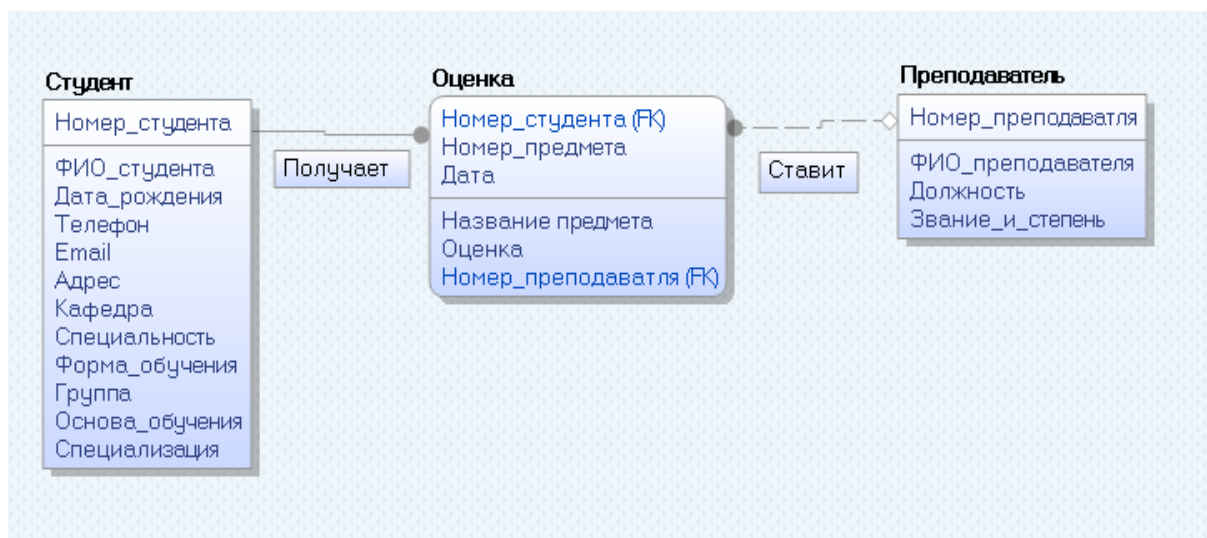


Рис. 3. Модифицированная схема логической модели.

4. Задания

1. Создать новый проект в ERWin.
2. Сформировать модель БД системы согласно перечню информационных объектов.
3. Включить в модель имена сущностей и определить их атрибуты.
4. Сохранить модель в файл.
5. Изменить масштаб модели.
6. Распечатать полученную модель.

7. Выбрать сервер БД.
8. Сгенерировать схему БД для выбранного сервера.

5. Контрольные вопросы

1. Назовите уровни методологии IDEF1X.
2. Из каких моделей состоит логический уровень?
3. Из каких моделей состоит физический уровень?
4. Что включает в себя диаграмма «сущность-связь»?
5. Что включает в себя модель данных, основанная на ключах?
6. Какую информацию содержит трансформационная модель?
7. Что включает в себя полная атрибутивная модель?
8. Сформулируйте требования, в которых необходимо убедиться перед началом проектирования БД.
9. Что называется моделью СУБД?
10. Перечислите преимущества от использования CASE-средства ERWin
11. Как вызвать диалоговое окно Report Browser?
12. Какие кнопки панели инструментов позволяют изменить уровень просмотра модели?
13. Как сгенерировать схему БД?
14. Каким образом осуществляется выбор сервера для генерации схемы БД?
15. Как добавить сущность на диаграмму?
16. Как добавить категорию в сущность?
17. Назовите виды связей.
18. Как перемещать атрибуты внутри сущности?
19. Как добавить текст в диаграмму?
20. С помощью какой кнопки на панели инструментов переключаются области модели?

ЛИТЕРАТУРА

1. Вендров А. М. CASE-технологии. Современные методы и средства проектирования информационных систем [Электронный документ]. – М, 2006 (<http://www.citforum.ru>).
2. Маклаков С. В. “ERwin и BPwin. CASE-средства разработки информационных систем.” – М.: ДИАЛОГ-МИФИ, 2000
3. Рамбо Д., Якобсон А., Буч Г. UML. Специальный справочник. СПб.: Питер, 2002.
4. Калянов Г. Н. CASE. Структурный системный анализ (автоматизация и применение). М., "Лори", 1996.
5. Марка Д. А., МакГоуэн К. Методология структурного анализа и проектирования. – М: "МетаТехнология", 1993.

Содержание

ВВЕДЕНИЕ.....	3
ЛАБОРАТОРНАЯ РАБОТА №2.....	4
МЕТОДОЛОГИЯ IDEF1X.	4
ПРОЕКТИРОВАНИЕ ЛОГИЧЕСКОЙ МОДЕЛИ СИСТЕМЫ.....	4
КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ	4
1. ERD-диаграммы	4
<i>1.1. Определение сущностей и атрибутов.....</i>	<i>5</i>
<i>1.2. Логические взаимосвязи.....</i>	<i>5</i>
<i>1.3. Проверка адекватности логической модели.....</i>	<i>6</i>
2. Модель данных, основанная на ключах.....	6
<i>2.1. Выбор первичного ключа</i>	<i>6</i>
3. ПРИМЕР	8
4. ЗАДАНИЯ	11
5. КОНТРОЛЬНЫЕ ВОПРОСЫ	12
ЛИТЕРАТУРА.....	13