МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение   
высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра вычислительная техника

**ЭССЕ по дисциплине «Автоматизация проектирования информационных систем»**

**Глава №1 «ОСНОВЫ СЕМАНТИЧЕСКОЙ МОДЕЛИ «СУЩНОСТЬ-СВЯЗЬ»**

Отчёт составил:  
Бакалавр гр. ИВТАСбд-42  
Сулейманов М.З.

Отчёт принял:  
Профессор кафедры ВТ  
Токмаков Г.П.

Ульяновск

УлГТУ

2024

1.1 МЕТОДОЛОГИЧЕСКАЯ ОСНОВА РЕАЛИЗАЦИИ МОДЕЛИ «СУЩНОСТЬ-СВЯЗЬ»

1.1.1 МЕТОДОЛОГИИ И СТАНДАРТЫ МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ

С появлением на рынке сложных программных продуктов, предназначенных для комплексной автоматизации управления предприятием, в практику большинства аналитиков вошло понятие «моделирование бизнес-процессов». Комплексные обследования предприятий всегда являются сложными и существенно отличающимися от случая к случаю задачами. Но для решения задач моделирования сложных систем существуют хорошо обкатанные методологии и стандарты. К таким стандартам относятся методологии семейства IDEF. С их помощью можно эффективно отображать и анализировать модели деятельности широкого спектра сложных систем в различных разрезах. Одна из них IDEF1X (IDEF1 Extended) – методология построения реляционных структур.

1.1.2 МЕТОДОЛОГИЯ IDEF1X

IDEF1X использует условный синтаксис, специально разработанный для удобного построения концептуальной схемы.

Концептуальной схемой называют универсальное представление структуры данных, независимое от конечной реализации в базе данных и применяемой аппаратной платформы.

Сущность является основным объектом в методологии IDEF1X и описывает собой совокупность или набор экземпляров похожих по свойствам, но однозначно отличаемых друг от друга по одному или нескольким признакам. На диаграмме обозначается прямоугольником, где верхняя часть – ключевые поля, нижняя – не ключевые поля.

Связи типов «один к одному» и «один ко многим» в IDEF1X представляют собой ссылки, соединения и ассоциации между сущностями. Связи в соответствии со спецификацией обозначаются глаголами, показывающими, как эти сущности соотносятся между собой.

Если сущности в IDEF1X диаграмме связаны, то эта связь моделируется с помощью атрибутов дочерней сущности. Передаваемые атрибуты называются мигрирующими.

Отношения «многие ко многим» обычно используются на начальной стадии разработки диаграммы, например, в диаграмме зависимости сущностей и отображаются в IDEF1X в виде сплошной линии с точками на обоих концах. В дальнейшем такая связь должна быть заменена дополнительной сущностью.

1.1.3 ПРЕИМУЩЕСТВА IDEF1X

Основным преимуществом IDEF1X, по сравнению с другими многочисленными методами разработки реляционных БД, является жесткая и строгая стандартизация моделирования.

1.2 ОСНОВНЫЕ ЭЛЕМЕНТЫ СЕМАНТИЧЕСКОЙ МОДЕЛИ ENTITY-RELATIONSHIP (СУЩНОСТЬ-СВЯЗЬ)

Наиболее известной семантической моделью данных, реализованной на основе методологии IDEF1X, является ER-модель. Основными понятиями ER-модели являются атрибуты, сущности и связи.

1.2.1 АТРИБУТЫ

Атрибуты описывают характеристики сущностей, представляются именами существительными и служат для описания состояния экземпляра сущности. Конкретным экземпляром атрибута является значение. У атрибутов в ER-модели описание имени домена или типа данных не является обязательным. Производители CASE средств проектирования реляционных БД стремятся не связывать обеспечиваемые ими возможности семантического моделирования с конкретной реализацией СУБД. Это позволит проще реализовать концептуальную модель для разных СУБД.

1.2.1.2 КЛАССИФИКАЦИЯ АТРИБУТОВ

Атрибуты делятся на две группы и являются либо ключом, либо нет.

Ключевыми являются атрибуты, значения которых определяют значения других атрибутов. Значения ключевых атрибутов не зависят от значений никаких других атрибутов.

Ключевые атрибуты, в свою очередь, подразделяются на:

Первичный ключ. Атрибут или набор атрибутов, уникально идентифицирующий единственный экземпляр сущности. Первичный ключ должен быть статическим и неразрушаемым (не подвергается изменениям).

Составной ключ. Ключ, который состоит более чем из одного атрибута, называется составным, сложным или компонентным. Ни одна часть ключа не должна быть неопределенной (NULL).

Альтернативный ключ. Кандидат в ключи, не выбранный в качестве первичного ключа, могут быть использованы при индексировании.

Искусственные первичные ключи – это атрибуты, созданные с единственной целью – идентификации конкретных экземпляров сущности.

Внешним ключом является атрибут или группа атрибутов, составляющих первичный ключ другой сущности. Они представляют связи между сущностями.

Могут быть как идентифицирующими, так и неидентифицирующими. Неидентифицирующие внешние ключи становятся не ключевыми атрибутами. Идентифицирующие ключи становятся частью первичного ключа в той сущности, в которую они мигрировали.

Неключевыми являются атрибуты, значения которых зависят только от значений первичного ключа или составного первичного ключа.

1.2.1.3 ОБЛАСТЬ ОПРЕДЕЛЕНИЯ АТРИБУТА

Область определения атрибута задает список разрешенных значений, которые атрибут может принимать в конкретном экземпляре сущности. Область определения включает, по меньшей мере, область определения универсального типа данных и может включать область определения, заданную пользователем. Эти области определения часто специфичны для организации и должны определяться и использоваться единообразно в пределах данной организации.

1.2.2 СУЩНОСТИ

Сущности служат для визуального представления логической группировки атрибутов и соответствуют таблицам реляционной модели данных. Сущностями могут быть как вещественные объекты, такие как ПЕРСОНА или КНИГА, так и абстрактные концепции, такие как прием заказов или размещение клиентов в гостинице.

1.2.2.1 КЛАССИФИКАЦИЯ СУЩНОСТЕЙ

В модели ER различаются две группы сущностей: зависимые и независимые. Независимая сущность не нуждается в информации из другой сущности для идентификации уникального экземпляра. Она представляется на ER-диаграмме в виде прямоугольника.

Зависимая сущность должна привлекать информацию из другой сущности для идентификации уникального экземпляра. Она представляется на ER-диаграмме в виде прямоугольника с закругленными углами.

Зависимые и независимые сущности разделяются на несколько типов:

* Стержневые (Первичные) сущности. Они представляют важные объекты, информацию о которых следует хранить. Могут быть как независимыми, так и зависимыми.
* Кодовые сущности. Классификаторы, содержат строки, определяющие набор значений или область определения для атрибута, являются независимыми.
* Ассоциативные сущности. Сущности, которые содержат первичные ключи двух или более других сущностей. Ассоциативные сущности всегда зависимы.

1.2.2.2 ВЫДЕЛЕНИЕ СУЩНОСТЕЙ

Для определения сущности необходимо осуществить группировку атрибутов, используя функциональные взаимосвязи между ними.

Функциональная зависимость в реляционной модели определяет тот факт, что каждый атрибут сущности должен функционально зависеть от первичного ключа сущности и не зависеть функционально от любого другого атрибута модели. Если это условие не соблюдается, то атрибут должен быть перемещен в новую сущность, где это положение будет соблюдаться.

Для размещения атрибута в той или иной сущности используются правила нормализации. Нормализация данных означает проектирование структур данных таким образом, чтобы удалить избыточность и ограничить несвязанные структуры.

Простые правила нормализации:

1. Размещайте повторяющиеся атрибуты в зависимых сущностях. Создайте зависимую сущность, которая будет содержать набор атрибутов для представления повторяющихся атрибутов.
2. Каждый факт в модели должен быть представлен только один раз. Логическая модель не должна содержать избыточных данных.
3. Размещайте атрибуты, не зависящие от первичного ключа, в зависимых сущностях.
4. Устраняйте отношения «многие-ко-многим». Создается ассоциативная сущность, которая устраняет отношение «многие-ко-многим».

На уровне логической модели приведение к четвертой нормальной форме не имеет смысла, так как при использовании CASE-средств эта операция осуществляется автоматически.

1.2.3 ОТНОШЕНИЯ

Отношение или связь – это графически изображаемая ассоциация, устанавливаемая между двумя типами сущностей.

В ER-модели отношение представляется линией, соединяющей две сущности, и глагольной конструкцией, которая описывает, как две сущности зависят друг от друга. Глагольная конструкция – это механизм описания бизнес-правил, определяющих отношение.

1.2.3.1 СВОЙСТВА ОТНОШЕНИЙ

Отношение обладает следующими свойствами:

Степень отношения представляет собой число сущностей, ассоциированных с отношением. В основном отношения имеют степень единица (унарные отношения) или двойка (бинарные отношения). Бинарные отношения представляют случаи, когда одна сущность связана с другой. Бинарные отношения отражают наиболее распространенные взаимосвязи, присутствующие в реальном мире. Унарные, или рекурсивные, отношения представляют случаи, когда экземпляр сущности связан с другим экземпляром той же самой сущности.

Направленность отношения указывает на исходную и подчиненную сущность в отношении. Сущность, из которой отношение исходит, называется родительской сущностью. Сущность, в которой отношение заканчивается, называется дочерней сущностью.

Количество элементов отношения задает максимальное число экземпляров одной сущности, которые могут быть связаны с экземплярами другой сущности. Количество элементов определяется для обеих сторон отношения для исходной и завершающей сущностей. Количество элементов часто выражается как один или много. Один и много могут появляться в трех различных комбинациях:

* «один к одному» (1:1) – один экземпляр сущности связан с одним экземпляром другой сущности.
* «один ко многим» (1:N) – один экземпляр родительской сущности связан со многими экземплярами подчиненной сущности
* «многие ко многим» (M:N) – много экземпляров одной сущности связаны со многими экземплярами другой сущности (также называется неспецифическим отношением). Удаляется на 4-й нормальной форме.

1.2.3.2 КЛАССИФИКАЦИЯ ОТНОШЕНИЙ

В ERwin отношение между сущностями может принадлежать к одному из следующих типов:

– идентифицирующее отношение;

– неидентифицирующее отношение.

Идентифицирующим является отношение между двумя сущностями, в котором каждый экземпляр подчиненной сущности идентифицируется значениями атрибутов родительской сущности. Это означает, что экземпляр подчиненной сущности зависит от родительской сущности, и не может существовать без экземпляра родительской сущности.

Неидентифицирующим является отношение между двумя сущностями, в котором каждый экземпляр подчиненной сущности не зависит от значений атрибутов родительской сущности. Это означает, что этот экземпляр может существовать без экземпляра родительской сущности.

1.2.3.3 Роли отношений

Иногда имеют место отношения, когда связи между сущностями для разных экземпляров имеет различный смысл.

Для разрешения подобных ситуаций в модели «сущность-связь» введено понятие «роли» отношения, представляющие собой новые имена, под которыми мигрирующие атрибуты будут представлены в дочерней сущности. На диаграмме еще одна линия рисуется, но новая связь при этом не создается.

1.2.4 ПОДТИПЫ И СУПЕРТИПЫ СУЩНОСТЕЙ

Сущность может быть расщеплена на два или большее число взаимно исключающих подтипов, каждый из которых включает общие атрибуты. Эти общие атрибуты и/или связи явно определяются один раз на более высоком уровне.

Тип сущности, на основе которого определяются подтипы, называется супертипом.

1.3 ПРЕОБРАЗОВАНИЕ ER-ДИАГРАММЫ В РЕЛЯЦИОННУЮ СХЕМУ

Целью разработки концептуальной модели является ее преобразование в логическую с использованием элементов ER-модели.

1. Каждый простой тип сущности превращается в таблицу. (Простым типом сущности называется тип сущности, не являющийся подтипом и не имеющий подтипов).
2. Каждый атрибут становится столбцом таблицы с тем же именем.
3. Компоненты уникального идентификатора сущности превращаются в первичный ключ таблицы.
4. Связи «один ко многим» (и «один к одному») реализуются в виде внешних ключей.
5. Для поддержки связи «многие ко многим» между типами сущности A и B создается дополнительная таблица AB с двумя столбцами, один из которых содержит уникальные идентификаторы экземпляров сущности A, а другой – уникальные идентификаторы экземпляров сущности B.
6. Индексы создаются для первичного ключа (уникальный индекс), внешних ключей и тех атрибутов, на которых предполагается в основном базировать запросы.
7. Если в концептуальной схеме присутствуют подтипы, то возможны два способа их представления в реляционной схеме:

а) собрать все подтипы в одной таблице;

б) для каждого подтипа образовать отдельную таблицу.

Достоинствам способа a):

* соответствие логике супертипов и подтипов;
* обеспечение простого доступа к экземплярам супертипа и не слишком сложный доступ к экземплярам подтипов;
* возможность обойтись небольшим числом таблиц.

Недостатки способа а):

* прикладная программа, имеющая дело с одной таблицей супертипа, должна включать дополнительную логику работы с разными наборами столбцов (в зависимости от значения столбца ТИП) и разными ограничениями целостности (в зависимости от особенностей связей, определенных для подтипа);
* общая для всех подтипов таблица потенциально может стать узким местом при многопользовательском доступе по причине возможности блокировки таблицы целиком;
* для индивидуальных столбцов подтипов должна допускаться возможность содержать неопределенные значения (потребуется много памяти).

Достоинства способа б):

* действуют более понятные правила работы с подтипами (каждому подтипу соответствует одноименная таблица);
* упрощается логика приложений; каждая программа работает только с нужной таблицей.

Недостатки способа б):

* в общем случае требуется слишком много отдельных таблиц;
* работа с экземплярами супертипа на основе представления, объединяющего таблицы супертипов, может оказаться недостаточно эффективной;
* поскольку множество экземпляров супертипа является объединением множеств экземпляров подтипов, не все СУБД могут обеспечить выполнение операций модификации экземпляров супертипа.