



Ответы на экзаменационные вопросы по проектированию БД

Уровни моделей и этапы проектирования БД. Взаимосвязь этапов проектирования

На практике выделяют три уровня модели БД: инфологический (концептуальный), даталогический (логический) и физический ¹ ². На инфологической стадии создаётся формальная модель предметной области (концептуальная схема) без учёта СУБД ¹. Далее при даталогическом проектировании строится логическая схема (на уровне отношений) уже с учётом выбранного типа СУБД, но без привязки к физическим носителям ². И затем на физической стадии уточняются детали физической организации (файловая структура, индексы, т.д.) ³. Выход каждого этапа служит входной информацией для следующего ⁴ ⁵, обеспечивая сквозную связь между этапами.

Стандарты семейства IDEF. Детали IDEF1 и IDEF1X

IDEF (Integrated Definition) – семейство методологий моделирования, разработанных по программе ICAM (авиационная промышленность США) для структурированного описания различных аспектов системы. Стандарты IDEF охватывают функциональное моделирование (IDEF0), информационное (IDEF1), динамическое (IDEF2), процессное (IDEF3), объектно-ориентированное (IDEF4), онтологическое (IDEF5) и др. IDEF1 – методология моделирования информационных потоков внутри системы, позволяющая отобразить их структуру и взаимосвязи ⁶. IDEF1X (IDEF1 Extended) – расширение IDEF1 для проектирования реляционных БД (Entity-Relationship), то есть методология создания реляционных структур на основе ER-моделирования ⁷. Оба стандарта служат для концептуального и логического моделирования данных, но IDEF1X специально оптимизирован под проектирование СУБД.

Технологическая сеть проектирования (даталогическое и физическое проектирование)

Для стадии даталогического проектирования существует «технологическая сеть», описывающая входные и выходные артефакты. Входными данными служат результат инфологического проектирования (концептуальная модель), документация по СУБД и средствам проектирования, допустимые даталогические конструкции, операторы языка описания схем и ограничения СУБД ⁴. Выходом является готовая даталогическая модель (схема таблиц с атрибутами и связями). Аналогично, для физической стадии сеть включает входные данные: логическую модель, документацию по СУБД, выбранному оборудованию и прочие ограничения (технические, пользовательские) ⁸. Выходом физической стадии является физическая схема БД (таблицы с конкретными типами данных, индексы, и др.). В этих сетях результат предыдущего этапа (инфологического или даталогического) используется как вход для следующего ⁵.

Исходная и результатная информация на стадии инфологического проектирования. Компоненты концептуальной модели

На этапе инфологического (концептуального) проектирования исходной информацией является знание предметной области: непосредственное наблюдение процессов, изучение существующих документов и опрос участников бизнес-процессов ⁹. Результатом этого этапа становится концептуальная (инфологическая) модель – ER-схема предметной области. Компонентами такой концептуальной модели являются: описание классов объектов и связей между ними, описание информационных потребностей пользователей, описание существующих процедур и документов, алгоритмические зависимости показателей, интегральные характеристики и ограничения целостности, а также функциональная структура системы и требования к ней ¹⁰.

Понятия «класс объектов» и «объект» в предметной области. Использование в ER-модели. Разновидности объектов

- «Класс объектов» (сущность) – это множество реальных или абстрактных объектов предметной области, имеющих одинаковый набор свойств (атрибутов) ¹¹. * «Объект» – конкретный экземпляр (сущность), на который ссылается класс (например, «студент Иванов» – объект класса «Студент»). В ER-модели проектировщик оперирует именно классами-сущностями, а не отдельными объектами ¹². Объекты могут быть простыми или сложными. Простейший объект описывается одним классом. Сложные объекты бывают трёх видов: **составные** (целое–часть, например «Аудитория состоит из Стульев»), **обобщённые** (род–вид, например «Учащийся» – супер-класс для «Студент», «Школьник», «Аспирант») и **агрегированные** (объединяющие объекты процесса, например «Поставка» объединяет Поставщик, Продукция и Потребитель) ¹³ ¹⁴.

Изображение и описание свойств простых объектов. Разновидности свойств

Свойства объекта (его атрибуты) изображаются в ER-диagramмах подписи (имена) внутри прямоугольника класса или отдельными графическими элементами. В ERWin свойства указываются отдельными атрибутами сущности. Типы свойств включают: – **однозначные и многозначные** – одно значение на объект (обозначается одинарной стрелкой) или множество значений (двойная стрелка) ¹⁵; – **статические (С) и динамические (Д)** – соответственно неизменяемые или изменяемые со временем ¹⁶; – **обязательные и необязательные** – требующие наличия значения (NOT NULL) или допускающие NULL ¹⁷; – **составные** – когда атрибут разбивается на под поля (например адрес = город, улица и т.д.) ¹⁸. Графически в ERWin у атрибута можно установить флаг Required для обязательности ¹⁹, а множественные атрибуты оформляются отдельными таблицами-связками.

Связи объектов. Типы и отображение классов принадлежности

Связи (отношения) между сущностями в ER-модели задают взаимосвязи объектов. По кардинальности они бывают «один-к-одному», «один-ко-многим» или «многие-ко-многим» ²⁰.

Класс принадлежности (участия) указывает, обязательно ли каждое экземпляр класса должно участвовать в связи (полное участие) или может не участвовать (частичное участие). Обязательное участие помечается заполненной точкой у соответствующей сущности, необязательное – пустым кружком у конца связи ²¹. Выбор режима (CASCADE/RESTRICT и т.д.) при удалении/обновлении также настраивается в ERWin (вкладка RI Actions) ²².

Сложные объекты: классификация и отображение

Сложные объекты описываются несколькими классами. **Составной объект** соответствует отношению «целое-часть» ¹³; его части либо включаются в одну таблицу с добавлением внешнего ключа, либо хранятся в отдельных связанных таблицах (с FK на целое) ²³. **Обобщённый объект** отражает иерархию «род-вид» ¹³. Его можно отображать одним универсальным отношением для всех подклассов или же тремя способами: одной таблицей для супер-класса со всеми атрибутами, либо отдельными таблицами для каждого подкласса (каждая наследует ключ супер-класса) ²⁴ ²⁵. **Агрегированный объект** обычно моделируется как отдельная ассоциативная сущность (реляционная таблица) с внешними ключами на участвующие классы и собственными атрибутами процесса ¹⁴.

Возможности ERwin Data Modeler при логическом проектировании БД

ERwin Data Modeler поддерживает моделирование на логическом и физическом уровнях. Для логической модели доступно несколько режимов отображения: *Entity* (имена сущностей), *Definition* (описание), *Attribute* (список атрибутов), *Primary Key* (ключевые атрибуты) и *Icon* (иконки) ²⁶. Пользователь выбирает шаблон Logical/Physical и целевую СУБД при создании модели ²⁷. Сущности в ERwin отображаются прямоугольником с именем, а её атрибуты делятся горизонтальной линией на первичный ключ и остальные поля ²⁸. ERwin позволяет графически устанавливать связи между сущностями, определять ключи, а затем автоматически переводить логическую модель в конкретную схему целевой СУБД.

Даталогическое проектирование для реляционных БД: критерии оценки БД и возможности Erwin

При оценке качества спроектированной БД учитывают **адекватность** (соответствие структуре предметной области) и **полноту** (удовлетворение текущих и будущих потребностей пользователей) ²⁹ ³⁰. Также важны минимизация избыточности (нормализованность), простота поддержки и эффективность запросов. Erwin Data Modeler упрощает эти задачи: он поддерживает прямое и обратное проектирование баз ³¹, позволяя генерировать SQL-схему из ER-модели или строить ER-модель из существующей базы. Инструмент автоматически генерирует DDL и может проверить целостность и ключи, облегчая достижение нормализованной и целостной логической схемы.

Алгоритмы перехода от базовой ER-модели к схеме реляционной БД. Отображение простых объектов

Простейшие объекты (классы-сущности) переводятся в реляционные отношения таблиц по простым правилам ³². Каждой сущности соответствует отношение (таблица); атрибуты сущности становятся столбцами этого отношения ³². Первичный ключ сущности становится первичным

ключом таблицы ³³. Если сущность зависит от другой (существует идентифицирующая связь), то ключ «родительской» сущности добавляется в таблицу «дочерней» как внешний ключ ²³. Если связь необязательная, допускается NULL в поле FK, если обязательная — у FK ставят NOT NULL ²³.

Алгоритмы перехода от ER-модели к реляционной схеме. Отображение сложных объектов

Составные объекты: дочерние части, как правило, отображаются отдельными таблицами с внешним ключом на таблицу «целого» ²³. **Обобщённые объекты:** можно создать либо одну таблицу для каждого подкласса (включив туда все атрибуты суперкласса) ²⁴, либо использовать одну общую таблицу для суперкласса и дополнительные таблицы для атрибутов каждого подкласса ²⁵. **Агрегированные объекты:** реализуются как отдельные ассоциативные таблицы, содержащие FK на участвующие сущности и собственные атрибуты процесса (например, таблица «Поставка» с FK на «Поставщик», «Продукция» и «Потребитель») ¹⁴.

Возможности и особенности проектирования логической модели данных в ERwin

В ERwin Data Modeler логическая модель строится на понятии ER-диаграммы с нотацией IDEF1X ³¹. Инструмент позволяет создавать сущности, определять ключи (первичные и альтернативные) и атрибуты, задавать связи «один-ко-многим» или «многие-ко-многим», а также транслировать модель в SQL для конкретной СУБД ³⁴. ERwin автоматически проверяет правильность схемы, поддерживает нормализацию (например, позволяет построить модель в НФ) и документирует модель для анализа. Кроме того, система может генерировать базу данных (таблицы, индексы, триггеры и др.) из логической модели ³⁵, значительно ускоряя проектирование.

Возможности и особенности проектирования физической модели данных в ERwin

Физическая модель в ERwin зависит от выбранной СУБД ³⁶. На этом уровне задаются конкретные типы данных, размер полей, спецификации хранения (табличные пространства, файлы), индексы, ограничения и триггеры. ERwin позволяет настроить все физические параметры и автоматически преобразует логическую модель в набор SQL-команд создания объектов БД ³⁵. Он хранит информацию обо всех объектах (таблицах, индексах, представлениях и т.д.) как в системном каталоге СУБД ³⁶ и позволяет генерировать готовую базу (DDL) с учётом особенностей сервера.

Средства обеспечения целостности БД в ERwin и их применение

Для обеспечения целостности ERwin предоставляет встроенные средства. Атрибуты можно помечать как **Required** – тогда для них будет задано ограничение NOT NULL ¹⁹ (обеспечение сущностной целостности). Для связей в редакторе Relationship Editor задаются действия RI Actions при удалении/обновлении (NONE, RESTRICT, CASCADE, SET DEFAULT/NONE) ²², что позволяет задать правила ссылочной целостности между таблицами (ON DELETE/UPDATE) при генерации

СУБД. Таким образом, ERwin модель может включать все основные ограничения целостности, которые будут перенесены в физическую базу автоматически.

Прямое и обратное проектирование в ERwin и оценка изменения размера БД

ERwin поддерживает *прямое проектирование* (forward-engineering) – генерацию структуры базы данных из модели, и *обратное проектирование* (reverse-engineering) – создание ER-модели на основе уже существующей БД ³¹. Это позволяет синхронизировать модель и реальную базу. ERwin также умеет рассчитывать объём занимаемого пространства: при физическом проектировании задаётся число строк и типы, после чего система вычисляет размеры таблиц/индексов и может учесть «временные» изменения для планирования роста БД. Это даёт возможность оценить изменение ёмкости БД во времени и спланировать ресурсы.

Сравнительный анализ программных средств проектирования БД

К популярным CASE-средствам моделирования данных относятся CA ERwin Data Modeler, SAP PowerDesigner, IDERA ER/Studio, Oracle SQL Developer Data Modeler и другие ³⁷. Большинство из них используют схожие нотации (обычно IDEF1X) и поддерживают полный цикл от концептуального ER-моделирования до генерации физической СУБД ³⁷. Различия заключаются в функциональности: PowerDesigner дополнительно интегрирован с корпоративными архитектурами, ERwin специализируется на моделировании данных и имеет мощный репозиторий (ModelMart), Oracle Designer тесно связан с Oracle БД. Инструменты также отличаются интерфейсом, поддержкой командной работы и стоимостью. В целом многие обладают схожими возможностями по графическому построению моделей, проверке целостности и генерации DDL ³⁷.

Источники: Описания уровней и стадий проектирования БД и их взаимосвязей ² ⁴; стандарты IDEF1 и IDEF1X ⁶; технологические сети проектирования ⁴ ⁵; компоненты концептуальной модели ⁹ ¹⁰; понятия «класс» и «объект» ¹¹ ¹²; типы объектов ¹³ ¹⁴; свойства объектов ¹⁵ ¹⁷; связи и кардинальности ²⁰ ²²; алгоритмы преобразования ER→реляционная модель ³² ²³; возможности ERwin (логический/физический дизайн, прямое/обратное проектирование) ³¹ ³⁵.

¹ ² ³ 1.5. Уровни моделей и этапы проектирования бд

<https://studfile.net/preview/6354061/page:16/>

⁴ ⁵ ⁸ ⁹ ¹⁰ Базы данных. Ч.1.

https://shpora1.do.am/_Id/2/241_-_-2004.pdf

⁶ ⁷ УП - ТСиСА ПИ бакалавриат

https://dgunh.ru/content/glavnay/ucheb_deyatel/uposob/up-fgos-14-15-pmit-9.pdf

¹¹ ¹² 2.1.4. Преимущества использования er-моделирования

<https://studfile.net/preview/6354061/page:21/>

¹³ ¹⁴ 2.2.2. Разновидности объектов

<https://studfile.net/preview/6354061/page:22/>

15 16 17 18 2.2.4. Описание свойств объекта. Разновидности свойств

<https://studfile.net/preview/6354061/page:23/>

19 22 4.3. Задание ограничений целостности в erWin

<https://studfile.net/preview/6354061/page:57/>

20 21 21. Типы связи, их представление на ег-диаграмме. Класс принадлежности сущности его представление на ег-диаграмме.

<https://studfile.net/preview/5407664/page:9/>

23 32 33 Алгоритм перехода от ег к реляционной модели данных.

<https://studfile.net/preview/16527149/page:5/>

24 25 37 Диго С.М. Базы данных. Проектирование и создание - Стр 9

<https://studfile.net/preview/6382849/page:9/>

26 27 28 4. Режимы отображения модели

<https://studfile.net/preview/2049063/page:12/>

29 30 3.2. Критерии оценки бд

<https://studfile.net/preview/7638483/page:22/>

31 34 35 36 ERwin: логическое моделирование данных

<https://studfile.net/preview/9497884/page:19/>