

Логическое проектирование базы данных

Назначение логического проектирования

- Логическое проектирование используется для перенесения концептуального проекта на внутреннюю модель выбранной системы управления базой данных (СУБД)
- Логическое проектирование переводит программно-независимую концептуальную модель в программно-зависимую модель определения подходящих доменов, необходимых таблиц и соответствующих ограничений на доступ к ним.

Технология логического проектирования

- Все объекты логической модели (сущности, атрибуты и др.) отображаются на модель с определенной структурой, используемой СУБД. Например, реляционная СУБД – реляционная модель.
- В процессе логического проектирования должны быть определены права на пользование базой данных: кто будет иметь право использовать таблицы, какие участки таблицы или какие таблицы доступны и каким пользователям. Для этого в реляционной среде необходимо определить соответствующие представления и права доступа.

Для реляционной СУБД логическое проектирование включает в себя проектирование следующих элементов:

- таблицы
- индексы
- представления
- транзакции
- авторизация доступа и др.

Алгоритм преобразования ER-модели в реляционную базу данных



Шаг 1

Для каждой «сильной» сущности модели «сущность-связь» создать базовое отношение в соответствии с требованиями:

- каждому простому атрибуту сущности соответствует столбец таблицы;
- ключевой атрибут сущности становится первичным ключом отношения;
- если ключ составной, то для каждой его части создается отдельный столбец, а затем этот набор столбцов объявляется первичным ключом отношения.

Шаг 2

Для каждой слабой сущности создается отношение, состоящее из всех простых атрибутов этой сущности и дополнительных столбцов первичных ключей сущностей, от которых зависит ее существование.

Шаг 3

Если две сущности участвуют в связи «многие-ко-многим», необходимо создать отношение из ключей двух отношений, представляющих сущности-участники.

Шаг 4

Если две сущности участвуют в связи «один-к-одному», то в отношении, представляющее одну из этих сущностей, необходимо включить столбец внешнего ключа. Не следует помещать внешние ключи в оба отношения, так как это создает возможность для записи несогласованных данных.

Шаг 5

Если сущность имеет многозначный атрибут, для его представления следует создать отдельное отношение. Один столбец этого отношения будет внешним ключом к отношению, представляющему сущность, а второй будет представлять многозначный атрибут. Первичным ключом будет являться комбинация этих столбцов.