Лекция 4

Терминология реляционных ключей

Суперключ – атрибут или их множество, которые единственным образом идентифицируют кортеж данного отношения. Может содержать дополнительные атрибуты, поэтому как правило избыточный.

Потенциальный ключ – суперключ, который не содержит подмножества, также являющиеся суперключами данного отношения. Свойства потенциальных ключей:

1. Уникальность.
2. Неприводимость (никакое допустимое подмножество потенциального ключа не обладает уникальностью).
3. Отношение может иметь несколько потенциальных ключей. Если ключ состоит из нескольких атрибутов, то он называется составным.

Для назначения потенциального ключа необходимо знать смысл используемых атрибутов в реальном мире.

Первичный ключ – потенциальный ключ, который выбран для уникальной идентификации кортежа внутри отношения.

Альтернативные ключи – потенциальные ключи, которые не выбраны в качестве первичного.

Внешний ключ – атрибут или множество атрибутов внутри конкретного отношения, которое соответствует потенциальному ключу некоторого другого (возможно того же самого) отношения.

Если некий атрибут присутствует в нескольких отношениях, то его наличие отражает определённую связь между кортежами этих отношений. В таком случае говорят, что данный атрибут ссылается на первичный ключ базового отношения.

Представления связи между отношениями с помощью диаграммы «Сущность-связь» (ER-диаграмма). Рассмотренные ранее графические нотации для представления структуры БД являются избыточными. Минимально необходимой является совокупность реляционных схем. Реляционная схема – заголовок таблицы

Общепринятое обозначение включает: имя отношения, за которым в скобках располагаются имена атрибутов. Первичный ключ при этом подчёркивается. Связи между таблицами не представлены явно, но могут быть достроены на основании выделенных ключей.

Альтернативные формы хранения данных и модель данных.

Сетевые формы хранения данных

В сетевой модели данные представляются в виде коллекции записей. Связи представляются в виде наборов и реализуются с помощью указателей.

Сетевую модель можно представить в виде графа с записями в виде узлов и наборами в виде рёбер.

Такая модель используется для связи с прикладной программой в виде соответствующей структуры, задаваемой ключевым словом.

Иерархические формы хранения данных.

Иерархическая модель – ограниченный подтип сетевой модели.

В иерархической модели данные – коллекция записей, а связи – наборы. Каждый узел имеет только одного родителя. В этом случае узлы называются сегментами.

Исходный сегмент – некоторый сегмент в дереве по отношению непосредственно к нижележащему (порождённому).

Зависимый сегмент – сегмент, к которому существует путь от некоторого исходного по цепочке пар, исходной непосредственно порождённой.

Корневой сегмент – сегмент, не имеющий исходного.

Запись БД - любой экземпляр корневого сегмента и все зависимые от него сегменты.

Основной метод доступа к записи – индексно-последовательный. Т.е. имеется иерархия индексов вплоть до цилиндров и дорожек.

Каждый пункт индекса содержит физический адрес начала подчинённой группы данных и ключ последней записи, т.е. экземпляра сегмента в этой группе.

И сетевая, и иерархическая модели задаются непосредственно на физическом уровне, т.е. на уровне файлов.

Предикатная форма хранения данных.

Предикат, или пропозициональная функция, – функция P от набора переменных, которая отображает упорядоченный набор конечного числа независимых предметных переменных, каждая из которых принадлежит заданному множеству значений M1, M2 и т.д. в множество истинностных значений.

Содержательно определение предиката означает: данная сущность предметной области [название] определяется таким-то набором атрибутов [переменных], причём они могут иметь такие-то значения [доменов], и они имеют такие-то значения [значения в строке]

Таким образом, информацию о структуре реляционных таблиц можно выразить введением соответствующих предикатных символов. Предикатный символ – конструкция P от x1...xN без значений (пустая конструкция).

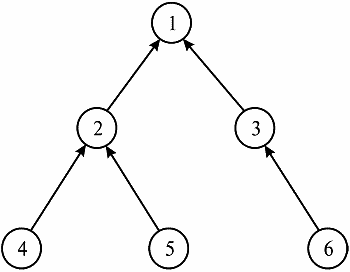
Информацию о содержимом таблиц можно выразить применением этих предикатных символов к тем и только тем конкретным наборам аргументов, на которых значения предикатов будут интерпретироваться как истинностные.

Предикатная форма содержательно практически совпадает с реляционными таблицами, однако имеет другой математический аппарат для обоснования. Аппарат называется исчисление предикатов.

Рекурсивные формы хранения данных

Все рассмотренные формы хранения представляют БД экстенсионально, т.е. путем явного перечисления всех факторов. Однако многие типы связи в БД по смыслу предполагают вложенность. Пример: начальник-подчиненный в большой компании, генеалогическое древо.

Предикатная форма позволяет ввести обобщенную связь, описываемую рекурсией.



Модель данных – интегрированный набор понятий для описания и обработки данных, связей между ними и ограничений на данные в некоторой предметной области; система типов данных, типов связей между ними и допустимых видов ограничений целостности, которые могут быть для них определены.

DM = {D, R, A} – Data Model.

D – заданное множество, носитель структуры.

R – конечный набор отношений, в которых находятся элементы множества – типовая характеристика структуры.

А – ограничивающие условия, которые накладываются на отношения или аксиомы структуры.

Множеству D соответствует реальное конечное множество элементов данных. Отношение R понимается в широком смысле как совокупность упорядоченных наборов элементов множества D. Аксиомы – то, что можно делать с данными (например, условие целостности).

Элемент данных понимает также, как в программировании, как элемент базового или абстрактного типа данных.

Запись – минимально уникально-идентифицируемая единица независимого хранения данных в конкретной системе БД.

Схема записи – описание внутренней структуры записи на языке, используемой в конкретной БД. Каждая запись должна быть объявлена вместе со своей схемой.

Поля записи – позиция в структуре хранения.

Ключи – одно или более полей записи, объявленное для её уникальной идентификации.