firebirdsql.org

Второй домен содержит номера всех отделов, третий домен – должности, четвертый – даты рождения. Каждый домен образует значения одного типа данных. Отношение сотрудник содержит три кортежа, каждый из которых состоит из четырех полей. Схема отношений представляет список имен атрибутов. Первичным ключом называется атрибут или группу атрибутов отношений, однозначно идентифицирующий каждый из его кортежей.

Каждое отношение имеет комбинацию атрибутов, которое может служить ключом. Ее существование гарантируется тем, что отношения – это множества без повторяющихся элементов или кортежей.

Ключи обычно юзают для достижения следующих целей:

1. исключение дублирования записей
2. упорядочения кортежей
3. ускорение работы с кортежем
4. организация связей между таблицами

Пусть отношение R1 имеет неключевой атрибут A, значения которого являются значениями ключевого атрибута B другого отношения R2, тогда говорят, что атрибут A отношения R1 – есть внешний ключ. С помощью внешних ключей устанавливаются связи между отношениями.

Имеются 2 отношения: студент (ФИО, группа, специальность) и предмет (название предмета и часы). Они связаны отношением студент\_предмет. В связующей таблице ФИО и название предмета образуют составной ключ, а сами атрибуты представляют собой внешние ключи, являющиеся первичными ключами других отношений.

Реляционная модель накладывает некоторые ограничения, называемые ссылочной целостностью. Основным ограничением является то, что каждому значению внешнего ключа должна соответствовать строка в связываемой таблице.

3.2 Индексирование

Двухуровневая система устраняет этот недостаток, поскольку ключи записей

Двухуровневая схема:

Для создания индекса для таблицы БД юзер указывает набор полей. Ключевые поля таблицы, как правило индексируются автоматически. Индексы, созданные по первичному ключу часто называют первичными, а юзерские, созданые по другим полям называют вторичными.

Основной причиной повышения скорости работы с данными в индексированных таблицах является то, что работа производится с небольшими индексными файлами, а не самими таблицами. Наибольший эффект от использования индексов, достигается при юзании больших таблиц.

3.3 Связывание таблиц

При проектировании БД инфу размещают в нескольких таблицах, которые связаны сементикой информации. В реляционные СУБД для указания связывания таблиц, производят их связывание. Связывание облегчает доступ к данным, повышает достоверность хранения инфы, а также снижает время обращения к данным. При связывании двух таблиц выделяют основную и дополнительную таблицы. Логическое связывание производится с помощью ключа связи. Суть связывания состоит в установлении соответствия полей связи основной и дополнительной таблицы. В качестве полей связи подчиненной таблицы чаще всего юзают ключевые поля, а поля связей основной таблицы могут быть как ключевыми, так и обычными. В зависимости от того, как определены поля связи основной и дополнительной таблиц, различают следующие виды связей:

1. Связь 1:1
2. Связь 1:М
3. Связь М:1
4. Связь М:М

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1:1 | 1:М | М:1 | М:М |
| Основная таблица | Являются ключевыми | Являются ключевыми | Не являются ключевыми | Не являются ключевыми |
| Дополнительная таблица | Являются ключевыми | Не являются ключевыми | Являются ключевыми | Не являются ключевыми |

3.4 Контроль целостности связей

Означает анализ содержимого двух таблиц на соблюдение следующих правил:

1. Каждой записи основной таблицы, соответствует 0 или более записей дополнительно.
2. В дополнительной таблице не должно быть записей, не имеющих родительской записи в основной таблице.
3. Каждая запись дополнительной таблицы должна иметь только одну родительскую запись.

4. Проектирование БД

Осуществляется на физическом и логическом уровне. Проектирование на физическом уровне зависит от юзаемой СУБД, а на логическом заключается в определениии структуры таблиц, формирования запросов к БД, разработки алгоритмов обработки информации и т.п.

При проектировании структуры БД выделяют 3 основных подхода:

1. Сбор инфы обо всех объектах в рамках одной таблицы и последующая декомпозиция на несколько взаимосвязанных таблиц.
2. Формулирование знаний о системе и требование к обработке данных с последующим применением case-технологий
3. Структурирование инфы на основе системного анализа

Избыточное дублирование данных и аномалии

Различают неизбыточное и избыточное дублирование данных. Неизбыточное допускается в БД, а избыточное может приводить к проблемам при обработке. Имеется отношение С\_Т с атрибутами «сотрудник» и «телефон». Для сотрудников, находящихся в одном помещении, номера телефонов совпадают, Т.о. номер телефона 4328 встречается несколько раз, хотя для каждого сотрудника телефон уникален, поэтому ни один из номеров не является избыточным. Т.о. при удалении одного из номеров будет утеряна инфа о том, как дозвониться только до одного из сотрудников.

|  |  |
| --- | --- |
| Сотрудник | Телефон |
| Иванов | 3721 |
| Петров | 4328 |
| Сидоров | 4328 |
| Егоров | 4328 |

Избыточное дублирование. Имеется отношение С\_Т\_Н (сотрудник-телефон-кабинет). В данном случае естественно, что сотрудники, находящиеся в одном кабинете имеют один и тот же номер телефона, следовательно, в отношении С\_Т\_Н имеется избыточное дублирование данных, поскольку номера телефонов Сидорова и Егорова можно узнать из записи со сведениями о Петрове.

|  |  |  |
| --- | --- | --- |
| Сотрудник | Телефон | № кабинета |
| Иванов | 3721 | 109 |
| Петров | 4328 | 111 |
| Сидоров | 4328 | 111 |
| Егоров | 4328 | 111 |

Неудачным подходом к устранению избыточности является следующее:

|  |  |  |
| --- | --- | --- |
| Сотрудник | Телефон | № кабинета |
| Иванов | 3721 | 109 |
| Петров | 4328 | 111 |
| Сидоров | - | 111 |
| Егоров | - | 111 |

При юзании такого подхода м.б. 3 негативных момента:

1. Потребуется разработка механизмов для поиска инфы для прочерков в таблице
2. Независимо от заполненности значения память для него будет выделена
3. При удалении записи о Петрове исчезает инфа и о номере телефона 111 кабинета

Возможным способом устранения избыточности является следующее: На основе существующих отношений С\_Т\_Н формируются 2 новых - Т\_Н и С\_Н.

Процедура декомпозиции отношения С\_Т\_Н на 2 отношения Т\_Н и С\_Н является основной процедурой организации отношения. Избыточное дублирование создает проблемы при обработке кортежей, названные аномалиями. Аномалией является такая ситуация в таблицах БД, которая приводит к противоречиям в БД , либо существенно усложняют обработку данных. Различают 3 вида аномалий: модификации, удаления и добавления.

Аномалии модификации проявляются в том, что изменение значения одного данного влечет за собой просмотр всей таблицы и соответствующее изменение других записей.

Аномалия удаления проявляется в том, что при удалении может пропасть другая информация, напрямую не связанная.

Аномалии добавления возникают в случаях, когда информацию нельзя поместить в таблицу пока она не полная, либо вставка требует дополнительного просмотра таблицы.

Формирование исходного отношения

Проектирование БД начинается с определения всех объектов, сведения о которых будут включены в базу. Затем определяются атрибуты объектов и они заносятся в одну таблицу, называемую исходным отношением. Рассмотрим это на примере отношения ПРЕПОДАВАТЕЛЬ.

Предположим, что для учебной части факультета создается база данных о преподавателе. Пусть было установлено, что необходимо хранение следующих атрибутов:

1. ФИО (уникальное)
2. Должность
3. Оклад
4. Стаж
5. Надбавка за стаж
6. Кафедра
7. Предмет
8. Группа
9. Вид занятий

Основным требованием является то, что все атрибуты должны быть атомарны.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ФИО | Должность | Оклад | Стаж | Надбавка за стаж | Кафедра | Предмет | Группа | Вид занятий |
| Иванов И.М. | Преп | 500 | 5 | 100 | 25 | СУБД | 256 | Практ |
| ИвановИ.М | Преп | 500 | 5 | 100 | 25 | Пл/1 | 123 | Практ |
| Петров | Ст. преп | 800 | 7 | 100 | 25 | СУБД | 256 | Лекц |
| Петров | Ст. преп | 800 | 7 | 100 | 25 |  |  |  |
| Сидоров | Преп | 500 | 10 | 150 | 25 | Паскаль | 256 | Лекц |
| Егоров | Преп | 500 | 5 | 100 | 24 | ПЭВМ | 244 | Лекц |

Полученное исходное отошение содержит избыточные данные. Различают явную и неявную избыточность.

Явная заключается в том, что строки с данными о преподавателях, проводящих занятия в нескольких группах повторяется соответствующее число раз.

Неявная избыточность проявляется в одинаковых окладах и надбавках.

Средством исключения избыточности является нормализация отношений.

4.2 Метод нормальных форм

Метод нормальных форм основан на понятии зависимости между атрибутами. Различают следующие виды зависимостей: однозначные, транзитивные и многозаначные.

Атрибут B функционально зависит от атрибута A, если каждому значению A соответствует в точности одно значение B. Это означает, что во всех кортежах с одинаковым значением атрибута A атрибут B будет иметь также одно и то же значение.

Функциональная взаимозависимость означает что если существует функциональная зависимость B от A и B от A, то между атрибутами A и B имеется функциональная взаимозависимость.

Частичная функциональная зависимость. В этом случае наблюдается зависимость неключевого атрибута от части составного ключа. Например атрибут должность функционально зависит от атрибута ФИО, являющегося частью ключа, поэтому атрибут должность частично зависит от ФИО.

Полная функциональная зависимость – это зависимость неключевого атрибута от всего составного ключа.

Транзитивная зависимость. Атрибут C зависит от атрибуата A транзитивно, если для A,B,C выполняется условие B зависит от A и C зависит от B, но обратная зависимость отсутствует.

Многозначная зависимость. Атрибут B многозначно зависит от атрибута A, если каждому значению A соответствует множество значений B. Имеют несколько видов 1:м, м:1 и м:м

Взаимонезависимые атрибуты. Атрибуты называются взаимонезависимыми, если не один из этих атрибутов не является функционально зависимым от других.