МИНИСТЕРСТВО ОБРАЗОВАНИЯ НАУКИ

 РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «Тверской государственный университет»

Факультет иностранных языков и международной коммуникации

Направление «Лингвистика»

Кафедра английского языка

РЕФЕРАТ

на тему: «Реляционная модель организации базы данных»

по дисциплине «Информационные технологии в лингвистике»

|  |  |
| --- | --- |
|   | Выполнила:студентка 1курса группы 13дневного отделенияШеламова Диана Владимировна |
|  |  |
|  | Проверила:к.ф.н., доцент,доцент кафедры английского языкаЕ. М. Масленникова  |

ТВЕРЬ

2021

**Содержание**

|  |  |
| --- | --- |
| Содержание ……………………………………………………… | 2 |
| Общее понятие реляционной модели организации базы данных …………………………………………………………… | 3 |
| Реляционная модель данных: кем, когда и для чего создана .... | 3 |
| Структура реляционных баз данных …………………………… | 4 |
| Структура данных в реляционной модели данных ………….... | 4 |
| Организация информации в реляционной базе данных …….... | 5 |
| Состав частей реляционной модели данных …………………... | 6 |
| Достоинства и недостатки реляционной модели данных …….. | 7 |
| Список литературы …………………………………………….... | 8 |

**Общее понятие реляционной модели организации базы данных**

Реляционные базы данных представляют собой базы данных, которые используются для хранения и предоставления доступа к взаимосвязанным элементам информации. Реляционные базы данных основаны на реляционной модели — интуитивно понятном, наглядном табличном способе представления данных. Каждая строка, содержащая в таблице такой базы данных, представляет собой запись с уникальным идентификатором, который называют ключом. Столбцы таблицы имеют атрибуты данных, а каждая запись обычно содержит значение для каждого атрибута, что дает возможность легко устанавливать взаимосвязь между элементами данных.

**Реляционная модель данных: кем, когда и для чего создана**

Реляционная модель данных - созданная Эдгаром Коддом логическая модель данных, описывающая:

* структуры данных в виде (изменяющихся во времени) наборов отношений;
* теоретико-множественные операции над данными: объединение, пересечение разность и декартово произведение;
* специальные реляционные операции: селекция, проекция, соединение и деление;
* специальные правила, обеспечивающие целостность данных.

Эдгар Франк «Тед» Кодд - (23 августа 1923 —18 апреля 2003) — британский учёный, работы которого заложили основы теории реляционных баз данных. Работая в компании IBM, он создал реляционную модель данных. В 1970 издал работу «A Relational Model of Data for Large Shared Data Banks», которая считается первой работой по реляционной модели данных.

Реляционная модель данных - это способ рассмотрения данных, то есть предписание для способа представления данных (посредством таблиц) и для способа работы с таким представлением (посредством операторов). Она связана с тремя аспектами данных: структурой (объекты), целостностью и обработкой данных (операторы).

В 2002 журнал Forbes поместил реляционную модель данных в список важнейших инноваций последних 85 лет.

Целями создания реляционной модели данных являются следующие:

* обеспечение более высокой степени независимости от данных;
* создание прочного фундамента для решения семантических вопросов и проблем непротиворечивости и избыточности данных;
* засширение языков управления данными за счёт включения операций над множествами.

**Структура реляционных баз данных**

Реляционная модель подразумевает логическую структуру данных: таблицы, представления и индексы. Логическая структура отличается от физической структуры хранения. Такое разделение дает возможность администраторам управлять физической системой хранения, не меняя данных, содержащихся в логической структуре. Например, изменение имени файла базы данных не повлияет на хранящиеся в нем таблицы.

Разделение между физическим и логическим уровнем распространяется в том числе на операции, которые представляют собой четко определенные действия с данными и структурами базы данных. Логические операции дают возможность приложениям определять требования к необходимому содержанию, в то время как физические операции определяют способ доступа к данным и выполнения задачи.

Чтобы обеспечить точность и доступность данных, в реляционных базах должны соблюдаться определенные правила целостности. Например, в правилах целостности можно запретить использование дубликатов строк в таблицах, чтобы устранить вероятность попадания неправильной информации в базу данных.

**Структура данных в реляционной модели данных**

При табличной организации данных отсутствует иерархия элементов. Строки и столбцы могут быть просмотрены в любом порядке, поэтому высока гибкость выбора любого подмножества элементов в строках и столбцах. Любая таблица в реляционной базе состоит из строк, которые называют записями, и столбцов, которые называют полями. На пересечении строк и столбцов находятся конкретные значения данных. Для каждого поля определяется множество его значений [(см. рис. 1)](#рис1).



Рис. 1. Таблица данных в реляционной модели

В реляционной модели данных применяются разделы реляционной алгебры, откуда и была заимствована соответствующая терминология. В реляционной алгебре поименованный столбец отношения называется ***атрибутом***, а множество всех возможных значений конкретного атрибута – ***доменом***. Строки таблицы со значениями разных атрибутов называют ***кортежами***. Атрибут, значение которого однозначно идентифицирует кортежи, называется ключевым (или просто ***ключом***). Так ***ключевое поле*** – это такое поле, значения которого в данной таблице не повторяется. В отличие от иерархической и сетевой моделей данных в реляционной отсутствует понятие группового отношения. Для отражения ассоциаций между кортежами разных отношений используется дублирование их ключей. Сложный ключ выбирается в тех случаях, когда ни одно поле таблицы однозначно не определяет запись.

Записи в таблице хранятся упорядоченными по ключу. Ключ может быть простым, состоящим из одного поля, и сложным, состоящим из нескольких полей. Сложный ключ выбирается в тех случаях, когда ни одно поле таблицы однозначно не определяет запись.

Кроме ***первичного ключа*** в таблице могут быть ***вторичные ключи***, называемые еще внешними ключами, или индексами. ***Индекс*** – это поле или совокупность полей, чьи значения имеются в нескольких таблицах и которое является первичным ключом в одной из них. Значения индекса могут повторяться в некоторой таблице. Индекс обеспечивает логическую последовательность записей в таблице, а также прямой доступ к записи.

По первичному ключу всегда отыскивается только одна строка, а по вторичному – может отыскиваться группа строк с одинаковыми значениями первичного ключа. Ключи нужны для однозначной идентификации и упорядочения записей таблицы, а индексы для упорядочения и ускорения поиска.

Индексы можно создавать и удалять, оставляя неизменным содержание записей реляционной таблицы. Количество индексов, имена индексов, соответствие индексов полям таблицы определяется при создании схемы таблицы.

Индексы позволяют эффективно реализовать поиск и обработку данных, формирую дополнительные индексные файлы. При корректировке данных автоматически упорядочиваются индексы, изменяется местоположение каждого индекса согласно принятому условию (возрастанию или убыванию значений). Сами же записи реляционной таблицы не перемещаются при удалении или включении новых экземпляров записей, изменении значений их ключевых полей.

С помощью индексов и ключей устанавливаются связи между таблицами. Связь устанавливается путем присвоения значений внешнего ключа одной таблицы значениям первичного ключа другой. Группа связанных таблиц называется схемой данных. Информация о таблицах, их полях, ключах и т.п. называется метаданными.

**Организация информации в реляционной базе данных**

 Информация в реляционной базе данных организуется по следующему принципу: пары таблиц объединяются между собой при помощи совпадающих ключей (одинаковых столбцов), которые называются информационными связями. Выделяют информационные связи трех типов:

* **«один к одному»**. Связи данного типа предполагают наличие в двух связанных таблицах только одного одинакового атрибута;
* **«один ко многим»**. Это означает, что при данном типе связи один атрибут первой таблицы совпадает с несколькими атрибутами во второй;
* **«многие ко многим».** В данном случае связи между двумя таблицами устанавливаются через несколько соответствующих друг другу атрибутов.

Чтобы информация в таблицах не дублировалась и не возникало затруднений ее обновления из-за необходимости редактирования каждой записи, реляционные базы данных, базу данных требуется нормализовать. Под нормализацией понимается организация данных в БД - создание таблиц и построение связей между ними.

Чаще всего при работе с БД выполняются три основных правила нормализации, что относит базу данных к:

* **первой нормальной форме**. В таких БД исключаются повторяющиеся группы в отдельных таблицах, для каждого набора связанных данных создаются отдельные таблицы, каждый набор связанных данных идентифицируется при помощи первичного ключа;
* **второй нормальной форме**. БД, соответствующая второму правилу нормализации, имеет отдельные таблицы, связанные при помощи внешнего ключа и содержащие наборы значений, которые применяются к нескольким записям;
* **третьей нормальной форме**. Третье правило нормализации исключает из БД не связанные с ключами поля.

Благодаря такой организации сокращается объем избыточных данных в БД, уменьшаются затраты на ее ведение, устраняется противоречивость хранимой в базе информации и обеспечивается ее безопасность.

**Состав частей реляционной модели данных**

Наиболее распространенная трактовка реляционной модели данных, принадлежит Дейту, который воспроизводит ее (с различными уточнениями) практически в своих книгах. Согласно Дейту реляционная модель состоит из трех частей, описывающих разные аспекты реляционного подхода: ***структурной части, манипуляционной части и целостной части.***

1. ***Структурная часть:***

Структурная часть (аспект), отвечает за принцип построения структуры реляционной базы данных на нормализированном наборе n-арных отношений, в форме таблиц. Важно, что реляционная база данных, структурно может представляться только в виде отношений;

1. ***Манипуляционная часть:***

В манипуляционной части модели утверждаются операторы манипулирования отношениями - реляционная алгебра и реляционное исчисление. Первый механизм базируется в основном на классической теории множеств (с некоторыми уточнениями), а второй - на классическом логическом аппарате исчисления предикатов первого порядка. Основной функцией манипуляционной части реляционной модели является обеспечение меры реляционности любого конкретного языка реляционных БД: язык называется реляционным, если он обладает не меньшей выразительностью и мощностью, чем реляционная алгебра или реляционное исчисление;

1. ***Целостная часть:***

В целостной части реляционной модели данных фиксируются два базовых требования целостности, которые должны поддерживаться в любой реляционной СУБД. Первое требование называется ***требованием целостности сущностей.*** Объекту или сущности реального мира в реляционных БД соответствуют кортежи отношений. Конкретно требование состоит в том, что любой кортеж любого отношения отличим от любого другого кортежа этого отношения, т.е. другими словами, любое отношение должно обладать первичным ключом. Как мы видели в предыдущем разделе, это требование автоматически удовлетворяется, если в системе не нарушаются базовые свойства отношений.

Второе требование ***называется требованием целостности по ссылкам*** и является несколько более сложным. Очевидно, что при соблюдении нормализованности отношений сложные сущности реального мира представляются в реляционной БД в виде нескольких кортежей нескольких отношений. Требование целостности по ссылкам, или требование внешнего ключа состоит в том, что для каждого значения внешнего ключа, появляющегося в ссылающемся отношении, в отношении, на которое ведет ссылка, должен найтись кортеж с таким же значением первичного ключа, либо значение внешнего ключа должно быть неопределенным (т.е. ни на что не указывать).

**Достоинства и недостатки реляционной модели данных**

***Достоинства:***

* Изложение информации в простой и понятной для пользователя форме (таблица).
* Реляционная модель данных основана на строгом математическом аппарате, что позволяет лаконично описывать необходимые операции над данными.
* Независимость данных от изменения в прикладной программе при изменении.
* Позволяет создавать языки манипулирования данными не процедурного типа.
* Для работы с моделью данных нет необходимости полностью знать организацию БД.

***Недостатки:***

* Относительно медленный доступ к данным.
* Трудность в создании БД основанной на реляционной модели.
* Трудность в переводе в таблицу сложных отношений.
* Требуется относительно большой объем памяти.

**Список используемой литературы:**

1. <https://function-x.ru/sql_relation_data_model.html>
2. <https://www.oracle.com/ru/database/what-is-a-relational-database/>
3. [https://ru.bmstu.wiki/Реляционная\_модель\_данных](https://ru.bmstu.wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)