

Структуры базы данных. Часть 1.

Наумов Д.А., доц. каф. ИТГД, доц. каф. КТ

Основы компьютерных наук (3 часть), 2019

Содержание лекции

1 Реляционная модель данных

2 Другие подходы и модели данных

- Модель Сущность-Атрибут-Значение
- Неполно структурированные модели данных
- Документ-ориентированные модели данных

Немного истории

- Эдгар Кодд, "Реляционная модель данных для больших совместно используемых банков данных"[1];
- Основа - теория отношений (XIX век);

Эволюция концепции реляционной модели данных

- версия 1, опубликованная в статье в CACM в 1970 г.;
- версия 2, определена в статье по поводу Тьюринговской премии в 1981 г.;
- версия 3, доопределена "дюжиной Кодда"— двенадцатью правилами и оценочной системой в 1985 году [2,3];
- версия 4, определена в опубликованной в 1990 году книге Кодда [4].

Немного истории

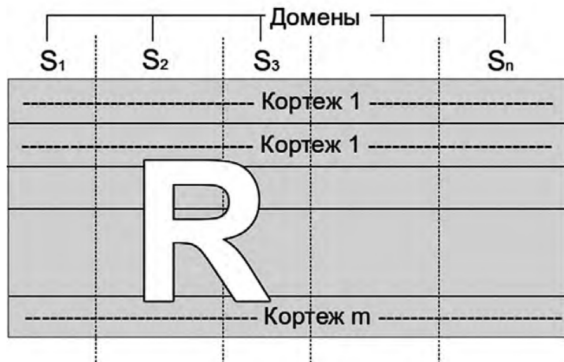
Выводы

- Существуют два мира с условными названиями "реляционная модель" и "промышленные РСУБД" и эти два мира пересекаются, но не полностью, создавая, в частности, проблемы переносимости приложений между СУБД разных производителей.
- Реляционная модель относится к логическим, то есть определённые в рамках модели структуры, операции над ними и задаваемые ограничения не зависят от способов реализации физической организации данных и управления ими.
- Каждая СУБД имеет свои особенности физической организации, поэтому одна и та же схема данных в рамках реляционной модели может иметь специфические параметры и опции, оптимизирующие работу с данными.

Определение

Отношение

Для заданных множеств S_1, S_2, \dots, S_n (не обязательно различных) R является отношением на этих n множествах, если представляет собой множество кортежей степени n , у каждого из которых первый элемент взят из множества S_1 , второй — из множества S_2 и т.д.



Определение

Графический смысл отношения

таблица с зафиксированным типом данных для каждой колонки.

Реляционные СУБД предлагают своим пользователям оперировать не определениями реляционной алгебры, а их адаптированными аналогами (таблицы, строки, столбцы и т. п.).

Свойства отношения

- в таблице нет двух одинаковых строк;
- таблица имеет столбцы, соответствующие атрибутам отношения;
- каждый атрибут в отношении имеет уникальное имя;
- порядок строк в таблице произвольный.

Определение

Отношение

является динамической моделью некоторого объекта реального мира.

| R1 | | |
|------------------|----------|--------|
| Дисциплина | Фамилия | Оценка |
| Теория автоматов | Крылов | 5 |
| Теория автоматов | Степанов | 5 |
| Теория автоматов | Иванов | 4 |
| Базы данных | Иванов | 3 |
| Базы данных | Степанов | 4 |

Схема отношения

перечень атрибутов данного отношения с указанием домена, к которому они относятся.

$$S_R = (A_1, A_2, A_3, \dots, A_n), A_i \subset D_i$$

Экземпляр отношения

отражает состояние данного объекта в текущий момент времени.

Преимущества реляционной модели данных

- теоретическая основа. Формально определяет базовые понятия модели, язык описания и операции над отношениями;
- стандартизация. Стандарты SQL-NN (SQL-89, SQL-92, SQL-99 и т. д.), имеющие несколько уровней полноты реализации, позволяют создавать приложения, переносимые между СУБД разных поставщиков;
- полное разделение доступа к данным от способа их физической организации;
- универсальность. Информационное моделирование сущностей реального мира в виде набора связанных таблиц является достаточно хорошим подходом в большинстве случаев;
- простота манипуляции данными с точки зрения конечного пользователя;
- SQL — развитый стандартизованный декларативный язык 4-го поколения.

Недостатки реляционной модели данных

- в общем случае, более низкое быстродействие по сравнению с сетевыми и иерархическими СУБД или другими подходами, обеспечивающими доступ к данным непосредственно на уровне их физической организации, например, индексированные файлы;
- неполнота реализации стандартов SQL-NN, а также специфические языковые и процедурные расширения СУБД разных поставщиков, осложняющие переносимость приложений (так называемый vendor lock);
- необходимость учёта некоторых особенностей модели на концептуальном уровне (ключи — идентификаторы сущностей), отсутствующая, например, в сетевой модели.

САЗ (Сущность-Атрибут-Значение, EAV (Entity-Attribute-Value))

подход, заключающийся в максимальном упрощении структуры хранения данных, за счет "вертикального хранения атрибутов что обеспечивает высокую гибкость изменения логической схемы данных.

| | | |
|--------------|---------------|--------------------|
| Товар №795 | Наименование | Мука пшеничная в/с |
| Товар №795 | Вес упаковки | 50 |
| Клиент №123 | Наименование | "Пирожки" ООО |
| Доставка №24 | Дата создания | 2014-06-15 |
| Доставка №24 | Статус | В пути |
| Продажа №867 | Дата создания | 2014-06-12 |
| ... | ... | ... |

| | Наименование | Вес упаковки | Дата создания | Статус |
|--------------|--------------------|--------------|---------------|--------|
| Товар №795 | Мука пшеничная в/с | 50 | | |
| Клиент №123 | "Пирожки" ООО | | | |
| Доставка №24 | | | 2014-06-15 | В пути |
| Продажа №867 | | | 2014-06-12 | |

САЗ даёт возможность менять логическую схему данных, что называется, "на лету". Действительно, достаточно добавить в метаданные описание нового атрибута и связать его с типами сущностей, как все приложения могут считывать и записывать значения данного атрибута, не принимая во внимание структурные изменения.

Преимущества модели САЗ

- гибкость схемы данных и возможность её изменения в динамике;
- высокая степень унификации и повторного использования структур хранения;
- возможность выполнения контекстно-независимых запросов, например, не принимающих в расчёт тип сущностей;
- возможность реализации на основе реляционной СУБД;
- относительно простая базовая концепция.

Недостатки модели САЗ

- сложность манипуляций данными вне CRUD-запросов;
- необходимость вводить собственный язык манипуляции данными;
- более низкая производительность за счёт большего числа соединений в запросах;
- необходимость оптимизации физического хранения.

Неполно структурированные модели данных

(НСМД, semi-structured data models) - способы организации данных, в которых схема и данные не разделены.

Случаи применения:

- исходные данные неполно структурированы, а возможности по их анализу и классификации ограничены;
- обмен информацией в гетерогенной среде.

```
<!-- Случай 1. Заказы списком -->
<orders>
  <order>
    <number>123</number>
    <items />
  </order>
</orders>
...
<!-- Случай 2. Заказы в составе продаж -->
<sales>
  <sale>
    <order>
      <number>123</number>
    </order>
  </sale>
</sales>
```

JSON

JavaScript Object Notation, формат хранения НСМД, менее удобный для восприятия человеком, но более прост для обработки анализаторами и более экономичный.

```
{
  "номер_заказа": "123-45",
  "дата": "2014-02-15",
  "состояние": "выполнен",
  "адрес_доставки": {
    "улица": "Ивановский пр. 5",
    "город": "Крыжополь",
    "индекс": "333222"
  },
  "телефоны": [
    { "тип": "домашний", "номер": "921-1112233" },
    { "тип": "мобильный", "номер": "921-1114455" }
  ]
}
```

Концепция самодокументированных данных: информация передаётся между системами в НСМД, при этом структура считается самодостаточной для распознавания и интерпретации смысловых и контекстных связей в документе без предварительного согласования

В документ-ориентированной модели данных (ДОМ)

основным элементом являются документы, организованные в линейные списки — коллекции, которые могут содержать другие коллекции.

- В отсутствии строгого определения, понятие "документ" соответствует иерархическим XML, JSON-документам и подобным структурам.
- ДОМ представляет собой синтез иерархической и неполно-структурированной моделей данных.

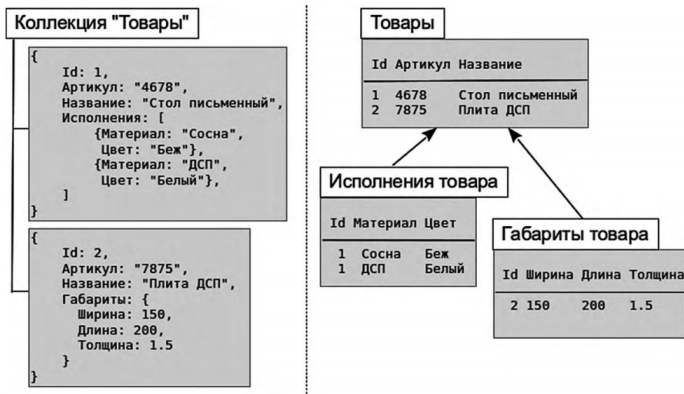
СУБД, реализующие документ-ориентированную модель, получили название NoSQL.

Реализации NoSQL СУБД

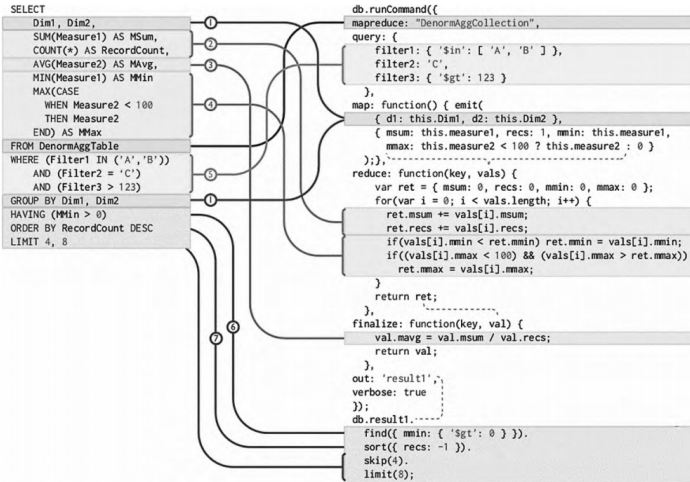
Cassandra, CouchDB, MongoDB.

Отличия от РМД и сетевой МД

В отличие от реляционной модели, где каждая строка таблицы имеет строго оговорённое число полей, определяемых заданными колонками и их типами, или от канонической иерархической модели, где каждый сегмент содержит список записей одного типа, документы списка в ДОМ могут иметь структуру с общей и различающимися частями.



Слабым звеном NoSQL является отсутствие стандартизованного декларативного языка запросов.



Рекомендуемая литература:

- ① Эдгар Кодд, Реляционная модель данных для больших совместно используемых банков данных. Перевод Communications of the ACM, Volume 13, Number 6, June, 1970. — Журнал "СУБД №1/1995;
- ② Codd, E.F. Is Your DBMS Really Relational?; Computerworld, October 14th, 1985.
- ③ Codd, E.F. Does Your DBMS Run By The Rules?; Computerworld, October 21st, 1985.
- ④ Codd, E.F. The Relational Model For Database Management Version 2. Reading, Mass.: Addison-Wesley, 1990.