

Содержание:

image not found or type unknown



Введение

В статье, опубликованной в журнале "Computer World", Тэд Кодд сформулировал двенадцать правил, которым должна соответствовать настоящая реляционная база данных. Они являются полуофициальным определением понятия реляционная база данных. Перечисленные правила основаны на теоретической работе Кодда, посвященной реляционной модели данных.

12 правил Кодда (Codd's 12 rules) — 12 правил (на самом деле их 13), которым должна удовлетворять каждая система управления реляционными базами данных.

Предложены английским математиком Эдгаром Коддом (Edgar Codd) ^[1] ^[2].

В действительности правила столь строги, что все популярные т. н. «реляционные» СУБД не соответствуют многим критериям.

Основная часть

Правила

правило 0: Основное правило (Foundation Rule): **Реляционная СУБД должна быть способна полностью управлять базой данных, используя связи между данными.:**

Чтобы быть реляционной системой управления базами данных (СУБД), система должна использовать исключительно свои *реляционные* возможности для *управления базой данных*.

правило 1: Явное представление данных (The Information Rule):

Информация должна быть представлена в виде данных, хранящихся в ячейках. Данные, хранящиеся в ячейках, должны быть атомарны. Порядок строк в реляционной таблице не должен влиять на смысл данных. Т.е. вся информация в базе данных должна быть предоставлена исключительно на логическом уровне и только одним способом – в виде значений, содержащихся в таблицах.

*правило 2: **Гарантированный доступ к данным** (Guaranteed Access Rule):*

Доступ к данным должен быть свободен от двусмысленности. К каждому элементу данных должен быть гарантирован доступ с помощью комбинации имени таблицы, первичного ключа строки и имени столбца. Т.е. логический доступ ко всем и каждому элементу данных (атомарному значению) в реляционной базе данных должен обеспечиваться путём использования комбинации имени таблицы, первичного ключа и имени столбца.

*правило 3: **Полная обработка неизвестных значений** (Systematic Treatment of Null Values):*

Неизвестные значения NULL, отличные от любого известного значения, должны поддерживаться для всех типов данных при выполнении любых операций. Например, для числовых данных неизвестные значения не должны

рассматриваться как нули, а для символьных данных — как пустые строки. Т.е. в настоящей реляционной базе данных должна быть реализована поддержка недействительных значений, которые отличаются от строки символов нулевой длины, строки пробельных символов, и от нуля или любого другого числа и используются для представления отсутствующих данных независимо от типа этих данных.

*правило 4: **Доступ к словарю данных в терминах реляционной модели** (Active On-Line Catalog Based on the Relational Model):*

Словарь данных должен сохраняться в форме реляционных таблиц, и СУБД должна поддерживать доступ к нему при помощи стандартных языковых средств, тех же самых, которые используются для работы с реляционными таблицами, содержащими пользовательские данные. Т.е. Описание базы данных на логическом уровне должно быть представлено в том же виде, что и основные данные, чтобы пользователи, обладающие соответствующими правами, могли работать с ним с помощью того же реляционного языка, который они применяют для работы с основными данными.

правило 5: Полнота подмножества языка (Comprehensive Data Sublanguage Rule):

Система управления реляционными базами данных должна поддерживать хотя бы один реляционный язык, который

(а) имеет линейный синтаксис,

(б) может использоваться как интерактивно, так и в прикладных программах,

(в) поддерживает операции определения данных, определения представлений, манипулирования данными (интерактивные и программные), ограничители целостности, управления доступом и операции управления транзакциями (begin, commit и rollback). Т.е. реляционная система может поддерживать различные языки и режимы взаимодействия с пользователем (к примеру, режим вопросов и ответов). При этом должен существовать по крайней мере один язык, операторы которого можно представить в виде строк символов в соответствии с некоторым четко определенным синтаксисом и который в полной мере поддерживает следующие элементы:

определение данных;

определение представлений;

обработку данных (интерактивную и программную);

условия целостности;

идентификация прав доступа;

границы транзакций (начало, завершение и отмена).

правило 6: Возможность модификации представлений (View Updating Rule):

Каждое представление должно поддерживать все операции манипулирования данными, которые поддерживают реляционные таблицы: операции выборки, вставки, модификации и удаления данных. Ну а это просто : все представления, которые теоретически можно обновить, должны быть доступны для обновления.

правило 7: Наличие высокоуровневых операций управления данными (High-Level Insert, Update, and Delete):

Операции вставки, модификации и удаления данных должны поддерживаться не только по отношению к одной строке реляционной таблицы, но по отношению к любому множеству строк. Т.е. возможность работать с отношением как с одним операндом должна существовать не только при чтении данных, но и при добавлении, обновлении и удалении данных.

*правило 8: **Физическая независимость данных** (Physical Data Independence):*

Приложения не должны зависеть от используемых способов хранения данных на носителях, от аппаратного обеспечения компьютеров, на которых находится реляционная база данных. Т.е. прикладные программы и утилиты для работы с данными должны на логическом уровне оставаться нетронутыми при любых изменениях способов хранения данных или методов доступа к ним.

*правило 9: **Логическая независимость данных** (Logical Data Independence):*

Представление данных в приложении не должно зависеть от структуры реляционных таблиц. Если в процессе нормализации одна реляционная таблица разделяется на две, представление должно обеспечить объединение этих данных, чтобы изменение структуры реляционных таблиц не сказывалось на работе приложений. Т.е. прикладные программы и утилиты для работы с данными должны на логическом уровне оставаться нетронутыми при внесении в базовые таблицы любых изменений, которые теоретически позволяют сохранить нетронутыми содержащиеся в этих таблицах данные.

*правило 10: **Независимость контроля целостности** (Integrity Independence):*

Вся информация, необходимая для поддержания целостности, должна находиться в словаре данных. Язык для работы с данными должен выполнять проверку входных данных и автоматически поддерживать целостность данных.

Т.е. должна существовать возможность определять условия целостности, специфические для конкретной реляционной базы данных, на подязыке реляционной базы данных и хранить их в каталоге, а не в прикладной программе.

*правило 11: **Дистрибутивная независимость** (Distribution Independence):*

База данных может быть распределённой, может находиться на нескольких компьютерах, и это не должно оказывать влияние на приложения. Перенос базы данных на другой компьютер не должен оказывать влияния на приложения.

Т.е. реляционная СУБД не должна зависеть от потребностей конкретного клиента.

*правило 12: **Согласование языковых уровней** (The Nonsubversion Rule):*

Если используется низкоуровневый язык доступа к данным, он не должен игнорировать правила безопасности и правила целостности, которые поддерживаются языком более высокого уровня. Т.е. в случае если в реляционной системе есть низкоуровневый язык (обрабатывающий одну запись за один раз), то должна отсутствовать возможность использования его для того, чтобы обойти правила и условия целостности, выраженные на реляционном языке высокого уровня (обрабатывающем несколько записей за один раз).

Заключение:

Правило 1 напоминает неформальное определение реляционной базы данных, приведенное ранее.

Правило 2 указывает на роль первичных ключей при поиске информации в базе данных. Имя таблицы позволяет найти требуемую таблицу, имя столбца позволяет найти требуемый столбец, а первичный ключ позволяет найти строку, содержащую искомый элемент данных.

Правило 3 требует, чтобы отсутствующие данные можно было представить с помощью недействительных значений (NULL), которые описаны в главе 5.

Правило 4 гласит, что реляционная база данных должна сама себя описывать. Другими словами, база данных должна содержать набор *системных таблиц*, описывающих структуру самой базы данных. Эти таблицы описаны в главе 16.

Правило 5 требует, чтобы СУБД использовала язык реляционной базы данных, к примеру SQL, хотя явно SQL в правиле не упомянут. Такой язык должен поддерживать все основные функции СУБД — создание базы данных, чтение и ввод данных, реализацию защиты базы данных и т.д.

Правило 6 касается *представлений*, которые являются виртуальными таблицами, позволяющими показывать различным пользователям различные фрагменты структуры базы данных. Это одно из правил, которые сложнее всего реализовать на практике.

Правило 7 акцентирует внимание на том, что базы данных по своей природе ориентированы на множества. Оно требует, чтобы операции добавления, удаления и обновления можно было выполнять над множествами строк. Это правило

предназначено для того, чтобы запретить реализации, в которых поддерживаются только операции над одной строкой.

Правила 8 и 9 означают отделение пользователя и прикладной программы от низкоуровневой реализации базы данных. Они утверждают, что конкретные способы реализации хранения или доступа, используемые в СУБД, и даже изменения структуры таблиц базы данных не должны влиять на возможность пользователя работать с данными.

Правило 10 гласит, что язык базы данных должен поддерживать ограничительные условия, налагаемые на вводимые данные и действия, которые бывают выполнены над данными.

Правило 11 гласит, что язык базы данных должен обеспечивать возможность работы с распределенными данными, расположенными на других компьютерных системах.

И, наконец, *правило 12* предотвращает использование других возможностей для работы с базой данных, помимо языка базы данных, поскольку это может нарушить ее целостность.

Список литературы

<http://referatwork.ru>

wikipedia.org

<http://wreferat.baza-referat.ru>

Более про 12 правил Кодда я не знаю что написать, итак всё понятно☺

Спасибо.

ВБОУ-201

Щуров Никита