

МАЛАХОВ Е.В.

ОРГАНИЗАЦИЯ БАЗ ДАННЫХ

Конспект лекций

**Кафедра
математического обеспечения компьютерных систем**

Последняя редакция: 09.2018

Содержание

Содержание	3
Понятие данных. Обработка данных. Информационная система	5
Структуры данных	8
Системы баз данных	10
Архитектура системы БД	11
Моделирование предметной области	13
Модели данных	17
Сетевая модель данных	18
Иерархическая модель данных	20
Реляционная модель данных	21
Реляционная алгебра	23
Объектная модель	29
Объектно-реляционная модель	30
Объектно-ролевая модель	30
Комбинированные модели данных. Переход от одной модели к другой	31
Проектирование БД	34
Нормализация отношений	35
Вторая нормальная форма	37
Третья нормальная форма	38
Нормальная форма Бойса-Кодда	39
Четвертая нормальная форма	41
Пятая нормальная форма	42
Шестая нормальная форма	43
Формализация связей между отношениями	44
Демонстрационный пример	50
SQL	53
Реализации моделей данных	53
Существующие <i>SQL</i> -ориентированные СУБД	54
Существующие СУБД других моделей данных	55
Средства разработки	55
Инсталляция и администрирование СУБД	56
СУБД Firebird	56
СУБД PostgreSQL	57
Язык определения данных (ЯОД, DDL) SQL. Часть 1	58
Типы данных	58
Домены	59
Поддержка пользовательских типов данных в СУБД PostgreSQL	59
Создание новых типов	59
Составные типы данных	60
Дополнительные структурные элементы	63
Таблицы	66
Логические операторы	68
Регулярные выражения	69
Потенциальные ключи	74

Первичный ключ.....	75
Внешние ключи.....	75
Создание объектно-реляционных связей в PostgreSQL.....	78
Индексы.....	79
Представления.....	80
Демонстрационный пример.....	83
Язык манипулирования данными (ЯМД, DML) SQL.....	85
Основные команды ЯМД SQL.....	85
Агрегатные функции.....	88
Аналитические функции.....	91
Реализация операций соединения.....	98
Демонстрационный пример.....	106
Подзапросы.....	110
Объединение.....	118
Иерархические (рекурсивные) запросы PostgreSQL.....	118
Поддержка механизма объектно-реляционных связей в PostgreSQL.....	120
Демонстрационный пример.....	121
ЯОД (DDL) SQL. Часть 2.....	123
Модифицируемые представления.....	123
Хранимые процедуры (ХП).....	124
Особенности построения хранимых процедур в СУБД PostgreSQL.....	127
Язык программирования PLpg/SQL СУБД PostgreSQL.....	127
Структура подпрограмм PLpg/SQL.....	128
Переменные в PLpg/SQL.....	129
Атрибуты.....	130
Управляющие структуры.....	131
Обработка ошибок.....	133
Курсоры.....	133
Триггеры.....	135
Поддержка хранилищ данных средствами PLpg/SQL.....	140
Демонстрационный пример.....	141
Сценарии.....	144
Язык управления данным SQL (ЯУД, DCL).....	145
Управление доступом.....	145
Особенности управления доступом в СУБД PostgreSQL.....	147
Роли и управление ролями.....	147
Управление схемами данных.....	149
Управление представлениями.....	151
Управление транзакциями.....	153
Блокировка.....	156
Сжатие базы данных.....	159
Приложение А Схема системы баз данных к разделу „SQL“.....	161
Приложение Б Пример сценария для построения таблиц к разделу „SQL“.....	164
Приложение В Примеры таблиц к разделу „SQL“.....	167
Список литературы.....	170

Понятие данных. Обработка данных. Информационная система

Для информационной поддержки управленческих решений в разных областях науки и техники создаются разные информационные системы.

На сегодняшний день не существует единого и общепринятого определения информационной системы. Так „Словарь по кибернетике“ утверждает, что „*информационные системы* — системы обработки данных о любой предметной области со средствами накопления, хранения, обновления, поиска и выдачи данных...“ [15]. Однако это определение не охватывает ряд аспектов обработки и передачи данных и не отражает место предметной области, для которой создаётся система.

Более корректным и полным представляется определение М.Р. Когаловского: „Автоматизированной информационной системой называется комплекс, который содержит вычислительное и коммуникационное устройства, программное обеспечение, лингвистические средства и информационные ресурсы, а также системный персонал, обеспечивающий поддержку динамической информационной модели некоторой части реального мира для удовлетворения информационных нужд пользователей“ [10].

Мы считаем, что и это определения следовало бы дополнить, сделав акцент на том, что **информационная система** — это совокупность связанных в единое целое системой каналов передачи информации баз данных, информационных хранилищ, баз знаний, а также информационных технологий, которые поддерживают процессы обработки, анализа и передачи информации различного уровня интеграции. При этом обмен информацией осуществляется как между объектами и подобластями выделенной предметной области, так и между данной предметной областью и остальной частью окружающего её мира.

В зависимости от объёма решаемых задач, используемых технических средств и организации функционирования информационные системы делятся на группы (классы) [15](рис. 1).

В определении информационной системы используются два понятия: данные и предметная область (подобласть).

Данные — цифровые и графические сведения об объектах окружающего мира [14].

Под термином **обработка данных** мы будем понимать последовательность действий, необходимых для выполнения некоторой задачи. Различают *числовую* и *нечисловую* обработки.

Для числовой обработки характерен большой объем вычислений, состоящих из ряда итераций (например, решение различных нелинейных уравнений, операций с матрицами и векторами и т.п.) с обязательным сохранением высокой точности результатов.

При нечисловой обработке не нужна высокая точность и большой объем вычислений. Однако имеется очень большой объем обрабатываемых данных и, кроме того, при нечисловой обработке требуется выполнение таких специфических операций, как поиск конкретных данных и их сортировка.

В само понятие „данные“ как объекты при числовой и нечисловой обработках вкладывается разное содержание.

Так при числовой обработке мы манипулируем переменными, векторами, матрицами, константами и др. При этом нас не интересует их содержимое. Например, при выполнении какой-нибудь арифметической операции или операции ввода/вывода необходимо знать адреса переменных (их имена), и не важно, что находится в этих переменных.

При нечисловой обработке объектами являются файлы, записи, поля, сетки, отношение и др. В этом случае нас больше интересует информация, которая содержится в конкретной записи, чем местоположение этой записи в файле или оперативной памяти.



Рис. 1. Классификация информационных систем

Термин **база данных (БД)** обозначает совокупность структурированных данных, предназначенных для общего использования при решении массовых проблем, существующих или сформулированных в определённой предметной области.

Оперируя какой-нибудь информацией, можно рассматривать данные, относящиеся к одной или общие для множества организаций или сфер деятельности человека.

Реальный мир, который должен быть отображён в БД, называют **предметной областью (ПрО)** [8].

Любая часть реального мира является результатом взаимодействия определённой части физического мира и интеллектуального мира, который возник в выделенной части физического мира [18].

Физическим миром будем называть Вселенную. Часть физического мира — это некоторая его k -мерная область D , заданная в виде множества k - мерных точек в m -мерном пространстве, которое описывает нашу Вселенную. Область D задаётся системой k -мерных отношений. То есть это множество всех k -мерных точек соответствующего пространства, которые удовлетворяют заданной системе математических соотношений.

Интеллектуальным миром будем называть биологическую систему, которая достигла такого уровня развития в некоторой физической части мира, при котором она в состоянии изменить часть физического мира, в котором она существует и продолжает развиваться, влияя на законы его эволюции.

Кроме того, в процессе развития интеллектуального мира на некотором этапе возникает систематическая потребность во все более совершенных формах анализа, обработки и организации информации и знаний, не связанных непосредственно с умственными процессами, но которые позволяют под управлением человека принимать желательные для него формы этих процессов.

Часть информационного пространства, которая имеет необходимую степень независимости, самостоятельности и способности к трансформации в необходимые формы и передачи их согласно человеческим законам, но, в то же время, независимую от человека, будем называть виртуальным миром.

Для уточнения понятия виртуальный мир или виртуальная реальность, или виртуальность (от лат. *virtus* — потенциальный, возможный, доблесть, энергия, сила, а также мнимый, воображаемый; лат. *realis* — вещественный, действительный, существующий) отметим два определения, приведенных в [6].

„Виртуальная реальность в постклассической науке — понятие, с помощью которого описывается совокупность объектов следующего (относительно нижележащей, порождающей их реальности) уровня. Эти объекты онтологично равноправны с порождающей их „константной“ реальностью и автономны; при этом их существование полностью определено перманентным процессом их воспроизведения порождающей реальностью...

Виртуальная реальность — технически конструируемая с помощью компьютерных средств интерактивная среда порождения и оперирования объектами, подобными реальным или воображаемым, на основе их трёхмерного графического представления, симуляции их физических свойств (объем, движение и т.п.), моделирования их способности влияния и самостоятельного присутствия в пространстве“.

Фактически, **виртуальный мир** — это воображаемый мир, для которого определены конкретные физические и математические законы и который, при соответствующих условиях, может быть с помощью технических средств сделан доступным для восприятия органами чувств человека. То есть виртуальным миром является „физический“ мир, созданный в пределах интеллектуального мира.

В каждом с миров: физическом, интеллектуальном, виртуальном, в свою очередь, можно выделить предметную область. Предметная область реального мира является продуктом взаимодействия соответствующих предметных областей каждого из миров.

В [21] определено, что предметная область или домен (от англ. *domain* — область) — чётко очерченный реальный, гипотетический или абстрактный мир, населённый взаимозависимым набором объектов, которые ведут себя согласно характерным для предметной области правилам и линиям поведения.

Построение любой информационной системы начинается с создания математической, информационной, онтологической и т.п. модели ПрО, которая может быть реализована в виде баз и хранилищ данных.

Поэтому для того, чтобы некоторую предметную область представить в БД, нужно выделить существенные понятия, необходимые пользователю, и связи между ними.

(Например, книги в библиотеке: читателю нужны автор и название, а не количество рисунков или качество бумаги).

Классификацию понятий предметной области, конкретное наполнение которых будет выполнено в БД, называют *логическим проектированием* данных или баз данных. С математической точки зрения можно сказать, что **логическое проектирование — это абстрагирование информации о предметной области.**

В связи с этим необходимо определить следующее:

- *физические данные* — данные, хранящиеся в памяти компьютера (дисковой или оперативной).
- *логическое представление данных* соответствует пользовательскому представлению физических данных.

Например: в файле БД хранятся 2 множества символьных строк — это физические данные. Вариант логического представления: авторы и названия книг.

Структуры данных

При использовании компьютера для сохранности и обработки данных необходимо хорошо знать тип и структуру данных, и определить способ их представления.

Программу, предназначенную для достижения некоторой цели, можно рассматривать как результат объединения структур данных и алгоритма.

Чаще всего существуют различные алгоритмы решения одной и той же задачи, которые зависят от способа представления и упорядоченности данных. Реализовать структуру данных необходимо так, чтобы обеспечить эффективность их обработки большим количеством алгоритмов.

Как правило, структурирование данных требует точного определения типа каждого элемента, входящего в структуру.

Математические принципы концепции типа, которые положены в основу языков программирования высокого уровня следующие [1]:

1. Любой тип данных определяет множество значений, к которым может относиться некоторая константа, которая может принимать любая переменная или выражение и которое может формироваться операцией или функцией.
2. Тип любой величины, обозначаемой константой, переменной или выражением, может быть выведен по её виду или из её описания (без необходимости каких-либо вычислений).
3. Каждая операция или функция требует аргументов определённого типа и даёт результат также фиксированного типа.

В вычислительной технике существует ряд так называемых **простейших стандартных типов** данных: целый, логический, символьный и т.п.

Кроме того, существует возможность определения новых типов путём *перечисления* множества их значений. Поэтому эти типы ещё зовут **перечисляемыми**. Они также являются простейшими, но, конечно, не стандартными. Например:

цвет {синий, красный, зелёный }

Структуры данных иначе называют *составными типами*, так как они представляют собой совокупности компонентов, относящихся к определённым ранее типам.

Если все компоненты относятся к одному типу, то такой тип называют *базовым*.

Очевидно, наиболее широко известная структура данных — одномерный или многомерный **массив**. Эта структура представляет собой отображение некоторого конечного множества данных на множество данных другого типа. Здесь областью определения является множество данных типа *индекс*, а областью значений — *множество элементов массива*. Массив состоит из элементов одного базового типа, поэтому структура массива *однородна*.

Наиболее общий метод получения составных типов состоит в объединении элементов разных типов. Причём сами элементы могут быть составными.

Пусть у нас есть следующие элементы:

$$a_i \in A_1, a_j \in A_2.$$

Если рассмотреть множество вида:

$$\{(a_i, a_j) | a_i \in A_1, a_j \in A_2\},$$

то мы получим составной тип, который называется **прямым (декартовым) произведением**. Каждый элемент такой структуры носит название *кортеж*.

$$A_1 = \{1, 3, 8, 9\}, A_2 = \{2, 3\}$$

$$A_1 \times A_2 = \{(1, 2), (3, 2), (8, 2), (9, 2), (1, 3), (3, 3), (8, 3), (9, 3)\}.$$

Ещё одним важным типом структурированных данных являются **последовательности**. Все элементы последовательности имеют один и тот же тип. Структура последовательности очень похожа на структуру массива. Существенное отличие заключается в том, что в массиве количество элементов фиксируется в его описании, а количество элементов последовательности (длина) конечно, но не фиксировано. Типичные примеры последовательного типа данных — *файл* и *стек*.

Последовательным файлом называется последовательность, для которой определены следующие операции:

1. Формирование пустой последовательности;
2. Выборка начального элемента последовательности;
3. Добавление элемента в конец последовательности,

Стек — последовательность, над которой возможно выполнение таких операций как:

1. Создание пустой последовательности.
2. Получение первого элемента последовательности.
3. Добавление элемента в начало последовательности.

Системы баз данных

Для создания, хранения и использования данных были разработаны специальные программные средства — *системы управления базами данных (СУБД)*, где БД, как мы уже говорили, это структурированная совокупность данных, хранимая в вычислительной системе.

Совокупность БД и СУБД представляют собой *банк данных (БнД)*.

Банк данных (БнД) является современной формой организации хранения и доступа к информации. Существует множество определений банка данных.

Например: „*Банк данных — это система специальным образом организованных данных (баз данных), программных, технических, языковых, организационно-методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования данных*“ [Общепромышленные руководящие материалы по созданию банков данных. - М.: ГКНТ, 1982].

Нельзя сказать, что термин „банк данных“ является общепризнанным. В некоторой англоязычной литературе в последнее время используется термин „система баз данных“ (Database System), который по своему смыслу близок к приведённому понятию банка данных (система баз данных включает базу данных, систему управления базами данных, соответствующее оборудование и персонал). Согласно семантике украинского и русского языков понятие „система баз данных“ воспринимается уже, чем его истинное обозначение [5]. Итак, слово „банк“ в этом смысле является лучшим, потому что „банк“ обычно обозначает не только то, что хранится в нем, но и всю инфраструктуру. Поэтому нельзя отождествлять понятие „база данных“ и „банк данных“.

СУБД — набор программных средств, позволяющих [14]:

- а) обеспечить пользователей языковыми средствами определения и манипуляции данными (выборка, обновление, удаление). Такими средствами является язык определения данных (ЯОД) и язык манипулирования данными (ЯМД). „Язык данных“, обозначающий или один, или оба эти языка, может быть включён в универсальный язык программирования (C, Pascal, C++, Delphi, C#, Java, ...) и называется *тогда подязыком* данных, или быть автономным и называемым *языком запросов*;
- б) обеспечить поддержку моделей данных пользователя.

При логическом проектировании априорно определённые рамки, которые используются для абстрагирования предметной области, называют моделью данных. То есть, модель данных — это средство для определения логического представления физических данных, относящихся к некоторому приложению. Модель данных позволяет представить разные связи или свойства объектов реального мира, определить способы получения новых понятий на основе имеющихся и т.п.;

- в) обеспечить программу, реализующую функции ЯОД и ЯМД, в которых возможное определение, создание и манипуляция данными.
- г) обеспечить защиту и целостность данных. Имеется в виду защита от несанкционированного доступа или несанкционированных действий над данными со стороны пользователей в многопользовательском и/или сетевом режимах.

Целостность имеет смысл при работе в тех же режимах. То есть, если один из пользователей внёс изменения в общие данные, то эти изменения должны быть доступны другим пользователям или получены ими. В противном случае возникают коллизии, типа знакомых ситуаций продажи железнодорожных билетов на одно и то же место.