

Тема: Системы Управления Базами Данных (СУБД). Терминология. Структура БД.

Содержание

Системы Управления Базами Данных (СУБД)	2
Что такое База Данных (БД)	2
Системы Управления Базами Данных (СУБД).	2
Функции СУБД	3
Терминология.....	4
Объекты и данные	4
Структура БД	5
Двумерная структура	5
Иерархическая структура	6
Используемая литература:	7

Системы Управления Базами Данных (СУБД)

Что такое База Данных (БД)

БД - это совокупность сведений о конкретных объектах в какой-либо предметной области.

Например,

- база данных по твердым сплавам (металлургия),
- база данных по театральным постановкам (культура),
- база данных поликлиники (медицина) и т. п.

Синоним термина БД — банк данных.

Системы Управления Базами Данных (СУБД).

Для взаимодействия пользователя с базами данных используются **системы управления базами данных (СУБД).**

Современные СУБД обеспечивают:

- набор средств для поддержки таблиц и соотношений между связанными таблицами;
- развитый пользовательский интерфейс, который позволяет вводить и модифицировать информацию, выполнять поиск и представлять информацию в текстовом или графическом виде;
- средства программирования высокого уровня, с помощью которых можно создать собственные приложения.

В мире насчитывается более 50 типов СУБД для IBM PC и совместимых с ними компьютеров.

По своему назначению СУБД делятся на:

- СУБД, предназначенные для выполнения специализированных задач.

Например,

для хранения юридических и экономических документов – Консультант, Гарант или для обработки данных в системах реального времени используются специализированные СУБД, которые оптимизированы для решения этих задач – 1-С, Инфо-бухгалтер, Парус.

- СУБД общего применения, предназначенные для решения широкого круга задач обработки данных.

СУБД общего применения можно далее разделить на:

- СУБД, предназначенные для работы на персональных компьютерах (например, Paradox, dBase, Access, FoxPro).
- СУБД, функционирующие в режиме клиент/сервер (например, Oracle, Informix, Ingres, Visual C или Visual Basic + SQL-server), требующие компьютеры другого класса.

СУБД, предназначенные для работы на персональных компьютерах могут работать в локальной сети, но в этом случае файл-сервер используется прежде всего в качестве дополнительного общего жесткого диска, к которому могут обращаться одновременно несколько пользователей. Если рабочей станции необходимо получить данные, размещенные на файл-сервере, на него посылается запрос. Файл-сервер передает требуемые данные на рабочую станцию, где они в дальнейшем обрабатываются. Одним из существенных недостатков использования файл-сервера при работе с базами данных является большой поток данных в сети, что приводит к снижению ее производительности.

В отличие от файл-сервера, сервер базы данных в режиме клиент/сервер используется не только для хранения информации, но и для обработки запросов к базе данных. Сервер базы данных устанавливается на файл-сервере локальной сети. Запросы рабочей станции обрабатываются сервером базы данных и обратно возвращается только результат выполнения запроса. Такой подход уменьшает поток данных в сети. Кроме того, обработка запросов сервером базы данных осуществляется быстрее, чем на рабочей станции, поскольку:

- ✓ в качестве сервера базы данных, как правило, используется гораздо более мощный компьютер;
- ✓ СУБД, используемая в качестве сервера базы данных, обладает более совершенными средствами обработки данных.

СУБД можно также классифицировать по степени их сложности и средствам взаимодействия с пользователем.

- СУБД для обработки небольших объемов информации.
Использование этих СУБД не требует предварительного обучения. Примерами таких СУБД являются базы данных, входящие в состав интегрированных пакетов типа Microsoft Works. Основным достоинством таких СУБД является их простота и мощные средства встроенной помощи, которые помогают пользователю шаг за шагом создавать таблицы, формы, отчеты и содержат большое количество встроенных шаблонов.
- СУБД, ориентированные на конечного пользователя, который не умеет программировать или не желает тратить на это время.
Наиболее яркими примерами таких систем являются Microsoft Office Access.
- СУБД, ориентированные на разработку законченного приложения (например, FoxPro, dBase и почти все СУБД типа клиент/сервер).
Данные системы требуют умения программировать на языке конкретного СУБД и используются при создании достаточно сложных систем. От пользователей законченных приложений не требуется знание СУБД, а только умение пользоваться приложением.
- СУБД, сочетающие ориентированность на конечного пользователя с мощными средствами программирования. Среди них лидирующее положение занимает Paradox, который сочетает «дружественный» пользовательский интерфейс и простоту его изучения с широкими возможностями.

Наличие большого количества разнообразных СУБД затрудняет выбор оптимальной системы для разработки конкретного приложения.

Функции СУБД

Мы так много говорили о различных аспектах СУБД, что пришла пора обобщить выполняемые ей функции, чтобы осознать в целом ее задачи, возможности и ограничения. Все функции разделим на две группы:

- **Основные**
- **Вспомогательные.**

Основные функции СУБД

1. **Хранение информации.** СУБД хранит структурированную и неструктурированную информацию об объектах конкретной предметной области, исполняя как архивную роль, так и роль поставщика оперативных (меняющихся) сведений.
2. **Просмотр и поиск.** СУБД обеспечивает *просмотр* информации по заданной вами совокупности полей, значения которых упорядочены по возрастанию или убыванию (в частном случае — по алфавиту).
Поиск — это частный случай просмотра, когда используя упорядоченность файла, система мгновенно указывает вам на запись (или на первую запись в блоке), которая содержит заданную вами совокупность критериев поиска и всю информацию об искомом объекте..

Например,

поиск в студентов по фамилии Петров.

3. **Выборка.** СУБД обеспечивает выборку определенной совокупности записей файла, которая удовлетворяет заданным вами условиям.
После фиксации первой (или единственной) искомой записи процесс поиска становится процессом *выборки*.

Например,

выбрать студентов по фамилии, начинающейся с буквы «П».

В общем случае выборку обслуживает аппарат установки фильтра, который реализован во всех программных системах.

4. **Отчеты.** СУБД обеспечивает определенные виды отчетов по файлам БД (суммирование числовых полей по файлу, по фильтру, по записи; преобразование значений полей, составление словарей и др.). Для решения многих оперативных задач (например, бухгалтерского учета) универсальные ИС приходится дополнять или заменять специализированными системами, которые реализуют сложные расчетные операции, не предусмотренные в универсальных СУБД.

Вспомогательные функции СУБД

5. **Ввод и редактирование информации.** СУБД обеспечивает разнообразный спектр приемов ввода и редактирования информации.
6. **Контроль информации.** Многие СУБД обеспечивают (должны, или не обеспечивают) автоматизированный логический контроль введенной информации.
7. **Отображение информации.** Все СУБД обеспечивают разнообразные способы представления данных в удобочитаемой форме как на дисплее, так и на печати.
8. При работе с СУБД пользователь часто пользуется еще двумя вспомогательными операциями: **архивированием файлов** и **созданием страховых копий**. Эти операции чаще всего не встраиваются в СУБД и выполняются по усмотрению пользователя с помощью широко известных программ.

Терминология

Объекты и данные

Мы можем сказать, что целью любой информационной системы является обработка **данных** об **объектах** реального мира.

Объект — это нечто существующее и различимое, отличающееся от другого объекта.

Например,

каждая фирма — это объект. Объектами являются также человек, стол, улица и т. д. Увиденную нами, например, бабочку на лугу мы сможем считать объектом, если укажем способ отличить ее от другой бабочки. В противном случае мы не воспринимаем бабочку как объект.

Объектами могут быть не только материальные предметы, но и более абстрактные понятия, отражающие реальный мир.

Например,

произведения искусства — книги (не как полиграфическая продукция, а как произведения), театральные постановки, кинофильмы; правовые нормы, философские теории, увлечения человека и проч.

Группа всех подобных объектов образует набор объектов.

Например,

наборами объектов могут быть заводы какой-то отрасли, товары на бирже, люди, работающие на предприятии, кинофильмы, т.п.

Информационная система в общем оперирует наборами объектов, спроектированными применительно к данной предметной области, используя при этом конкретные значения **данных** о тех или иных объектах.

Данное – это некоторый показатель, который характеризует объект и принимает для конкретного числовое или текстовое значение.

Например,

Набор объектов - группа фирм, выпускающих пластмассу:

Объект - каждая фирма,

Данное - годовой оборот фирмы, которое принимает числовое значение (у одной фирмы 250 миллионов руб., у другой — 1 миллиард и т. д.).

Данное - название фирмы, принимающее текстовое значение (у одной — «Сатурн», у другой — «Гелиотроп» и проч.).

Данные - название, специализация, адрес, фамилия директора, количество работающих и т. п.

Объект - продукция, производимая фирмой. Обычно каждый вид продукции обозначается цифровым кодом, позволяющим отличать один экземпляр этого объекта от другого. (так, например, код 445 может обозначать трубы медные, 446 — прутки медные, 455 — трубы латунные и т. П),

Данные вида продукции могут быть ее код, название, технико-экономические показатели, рассчитанные на единицу продукции (например, на тонну или штуку) — себестоимость, трудоемкость изготовления в станко-часах или человеко-часах и многие другие показатели.

То есть с другой стороны, вид продукции может выступать как данное по отношению к другому объекту — фирме.

Таким образом, *важнейшей задачей проектировщика СУБД является выбор объектов и описывающих их данных, а также установление связей между данными.* В зависимости от целей и масштабов информационной системы объекты характеризуются разным количеством данных, которое может меняться в весьма широких пределах.

Например,

обычные кадровые системы отличаются сравнительно небольшим набором данных о главном объекте СУБД — человеке (табельный номер, фамилия, год рождения, пол, национальность, образование, должность, оклад), однако, существуют специализированные библиографические системы с информацией о научных работниках, которые содержат широкий круг дополнительных данных (печатные труды, изобретения, открытия, подробные сведения о семейном положении, увлечениях и т. п.).

Структура БД

Двумерная структура

Разобравшись с понятиями «объект» и «данные», поставим себе следующую цель: научиться составлять наборы однородных объектов и структурировать их данные. Для начала попробуем составить телефонный справочник, — так, чтобы его могла обработать машина.

Справочник состоит из однородных объектов — номеров телефонов. Выберем для объекта следующие данные:

- номер телефона;
- населенный пункт;
- имя абонента (просто имя, или фамилия, или название);
- адрес абонента;
- категория абонента (друзья, родственники, сослуживцы, сервис, магазин и т. д.).

Все данные будем считать текстовыми. Присвоим этим данным следующие имена и длины (указаны в скобках рядом с именем):

Тогда справочник можно представить следующей таблицей:

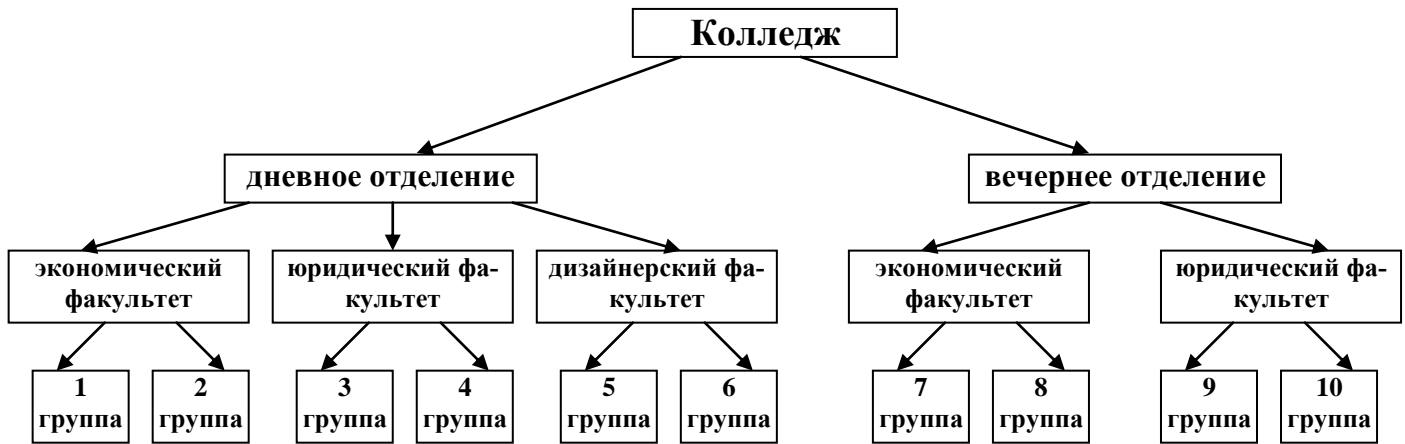
Номер	Населенный пункт	Имя абонента	Адрес	Категория
915-233-08-19	г.Москва	Петров Мих.Вас.	Садовая, 18	Сослуживцы
926-265-04-15	г.Москва	Дядя Коля	Зеленая, 11	Родственники
915-570-14-20	г.Москва	Химчистка	Кольшева, 5	Сервис
295-981-23-19	г.Химки	Эдик	Свободы, 57	Магазины
926-487-18-20	г.Москва	Терехов Анат.Дм.	Киевская, 2	Сослуживцы

Заметим, что эту базу данных образует одна простая двумерная таблица с фиксированным числом столбцов-данных и переменным числом строк-объектов (это число зависит от масштаба ваших контактов и меняется по мере появления новых знакомых или исчезновения старых).

Такое представление данных является наиболее простым и наглядным для человека и наиболее удобным для машины. Однако, нетрудно заметить, что мы умышленно выбрали для рассмотрения набор независимых (не связанных между собой) объектов — телефонов, каждый из которых описывается группой несвязанных между собой данных.

Хотя такие структуры характерны для большого числа приложений (культура, искусство, медицина, кадры и проч.), во многих случаях, особенно в технике, данные объектов связаны между собой, и тогда исходная структура оказывается более сложной.

Иерархическая структура



Описанную структуру данных можно представить себе как дерево, ствол которого — это набор объектов.

Перед нами типичная иерархическая структура, в которой исходный элемент (колледж) порождает другие элементы (отделения), причем эти элементы в свою очередь порождают следующие элементы (факультеты) и т. д.

Можно ли такое дерево представить в виде простой таблицы, подобной телефонному справочнику? Можно, — если мы не поленимся пройтись от каждого узла по всем веткам до каждого листочка персонально.

При этом каждый листочек (группа) займет отдельную строку в таблице:

Колледж	Отделение	Факультет	Группа
Колледж	дневное отделение	экономический факультет	1 группа
Колледж	дневное отделение	экономический факультет	2 группа
Колледж	дневное отделение	юридический факультет	3 группа
Колледж	дневное отделение	юридический факультет	4 группа
Колледж	дневное отделение	дизайнерский факультет	5 группа
Колледж	дневное отделение	дизайнерский факультет	6 группа
Колледж	вечернее отделение	экономический факультет	7 группа
Колледж	вечернее отделение	экономический факультет	8 группа
Колледж	вечернее отделение	юридический факультет	9 группа
Колледж	вечернее отделение	юридический факультет	10 группа

Заметим, что значения некоторых данных (колледж, отделение, факультет) пришлось повторить в таблице несколько раз.

Итак, мы познакомились с двумя типами информационных структур — простой двумерной (телефонный справочник) и иерархической (справочник колледжи), причем иерархическую структуру мы механически свели к простой двумерной.

Существуют и более сложные — *сетевые структуры*, в которых каждый порожденный элемент может иметь более одного порождающего элемента. Мы не рассматриваем здесь эти структуры, однако, каждый желающий может попробовать свести к двумерной таблице, например, генеалогическое дерево Романовых типичную сетевую структуру.

Используемая литература:

1. Гаврилов, М. В. Информатика и информационные технологии: учебник для СПО / М. В. Гаврилов, В. А. Климов. — 4-е изд., пер. и доп. — М.: Издательство Юрайт, 2018. — 383 с. — (Серия: Профессиональное образование). — ISBN 978-5-534-03051-8. — Режим доступа: www.biblio-online.ru/book/1DC33FDD-8C47-439D-98FD-8D445734B9D9.
2. Трофимов, В. В. Информатика в 2 т. Том 1: учебник для СПО / В. В. Трофимов; под ред. В. В. Трофимова. — 3-е изд., пер. и доп. — М.: Издательство Юрайт, 2018. — 553 с. — (Серия: Профессиональное образование). — ISBN 978-5-534-02518-7. — Режим доступа: www.biblio-online.ru/book/87EC2130-3EBB-45B7-B195-1A9C561ED9D9.
3. Трофимов, В. В. Информатика в 2 т. Том 2: учебник для СПО / В. В. Трофимов; отв. ред. В. В. Трофимов. — 3-е изд., пер. и доп. — М.: Издательство Юрайт, 2018. — 406 с. — (Серия: Профессиональное образование). — ISBN 978-5-534-02519-4. — Режим доступа: www.biblio-online.ru/book/14FE5928-69CF-41EC-A00B-3979EC8273C8.