

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет радиофизики и электроники
Кафедра системного анализа

В. В. Скакун

**Системы управления
базами данных**

Учебное пособие

Основные понятия

Оглавление

<u>1.ОСНОВНЫЕ ПОНЯТИЯ</u>	2
<u>1.1.Отличие от файловой системы хранения данных</u>	6
<u>1.2.Архитектура построения баз данных</u>	7
<u>1.3.Свойства баз данных</u>	9
<u>1.3.1.Независимость данных</u>	9
<u>1.3.2.Контролирование избыточности данных</u>	9
<u>1.3.3.Обеспечение целостности и правильности данных</u>	10
<u>1.3.4.Обеспечение безопасности и секретности</u>	12

Минск 2006

1. ОСНОВНЫЕ ПОНЯТИЯ

Термин «Базы данных» неразрывно связан с понятиями данных и информации. Поэтому логично было бы начать изложение материала из формулировки этих понятий.

Информация – это совокупность сведений, воспринимаемых из окружающей среды, передаваемых в окружающую среду, либо сохраняемых внутри информационной системы. Под информационной системой будем понимать любую систему, осуществляющую сбор, накопление и обработку информации. Понятие информации предполагает обязательное наличие источника и приемника информации. Когда источником и приемником являются люди, то говорят, что они обмениваются сообщениями. В случае, если источником информации является объект наблюдения, то наблюдатель получает информацию либо путем наблюдения либо в процессе активного воздействия на объект наблюдения. Если источником и приемником информации являются технические устройства, то говорят, что они обмениваются сигналами. И если приемником информации является некоторая информационная система, то говорят, что она получает, выдает и преобразует данные.

Производным от информации являются данные. Данные – это запись в соответствующем коде наблюдения, факта, объекта и т.п., пригодное для интерпретации, передачи, обработки и получения информации. Следовательно, понятие информации предполагает существование материального носителя информации и форму, способ, которым она связана со своим носителем. Таким образом, данные – это информация, представленная в виде, позволяющим автоматизировать ее сбор, хранение и дальнейшую обработку человеком или информационным средством.

Хранящиеся в информационной системе данные должны быть доступны в том виде, в каком они нужны для конкретной деятельности человека. Сегодня мы можем встретить систему обработки данных традиционного типа, в которой служащий вручную заполняет формы и помещает их в скоросшиватель, а рядом с ней современную систему с применением ЭВМ. Несмотря на поразительную несхожесть, обе эти системы обязаны предоставить достоверную информацию в определенное время, в определенном месте и с ограниченными затратами.

Информация получается из данных в процессе решения какой либо задачи. Однако не вся информация может быть напрямую выведена из данных. Например, человеческие эмоции, отраженные в художественных произведениях, могут быть восприняты совершенно по разному. Даже простая совокупность чисел в виде (25-05-99) может пониматься разным

способом: как номер телефона (25-05-99), либо как дата (25-05-99). Следовательно, для получения информации из данных необходимы определенные соглашения, называемые интерпретацией данных. Другая проблема состоит в том, что не существует определенного компактного кода для записи любой (в общем случае разнородной) информации.

Так как мы будем говорить в дальнейшем о технических информационных системах, то будет правомерно говорить только о той информации, которая может быть переведена на язык, понимаемый компьютером. Другим доводом в пользу выбора компьютера как информационной системы является стоимость хранения единицы информации (в настоящее время стоимость хранения единицы информации на компьютерных носителях минимальна). Зачастую на язык, понимаемый компьютером, переводится только часть информации (набор данных) о какой-то конкретной предметной деятельности человека. Такой набор данных, например, телефонный справочник, список товаров, адреса клиентов некоторой фирмы и называют базой данных (БД). В общем смысле термин База Данных можно применить к любой совокупности связанной информации, объединенной вместе по определенному признаку.

Приведем классическое определение БД, данное Дж. Мартиным: «БД – это совокупность взаимосвязанных данных при такой их минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений в определенной предметной области человеческой деятельности». Под предметной областью мы будем понимать часть реального мира, представляющего интерес для исследования. Таким образом, БД является динамической информационной моделью некоторой предметной области деятельности человека.

Когда речь идет о БД, то под ней очень часто понимают не сам набор связанных данных, а совокупность, состоящую из набора данных и программы обслуживания, осуществляющей взаимодействие пользователя с набором данных, а также проведение операций с данными. К таким операциям можно отнести операции поддержки структуры данных, обеспечения целостности данных, добавление, удаление данных, поиск, сортировку, фильтрацию и т.д. Обычно минимальный набор таких операций осуществляется ядром (процессором) базы данных. В таком случае правомерно говорить о совокупности, состоящей из набора данных, ядра базы данных и программы, осуществляющей интерфейс с пользователем (смотри рис. 1).

Полный набор операций с данными предоставляют Системы Управления Базами Данных (СУБД). Под СУБД понимают совокупность языковых и программных средств, обеспечивающих создание, поддержание

и доступ к данным как со стороны пользователей, так и со стороны приложений.

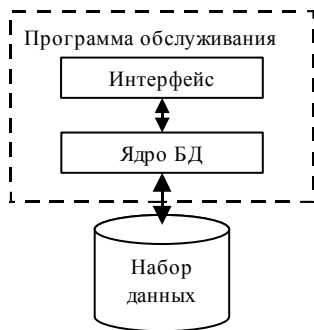


Рис. 1. Структура БД

Кроме средств поддержки структуры данных и операций с данными СУБД также предоставляет развитый пользовательский интерфейс, средства программирования высокого уровня, средства администрирования, обеспечения секретности и безопасности информации.

Таким образом, под БД будем понимать набор взаимосвязанных данных, а под СУБД – программную среду, позволяющую управлять многими БД и имеющую средства их разработки. В то же самое время не всякий набор данных

можно назвать термином БД. Давайте рассмотрим требования к набору данных, чтобы его можно было бы интерпретировать как БД:

1. Связность и логическая упорядоченность. Набор беспорядочных данных не может быть признан за БД (например, слова в некотором документе, хотя они объединены местом хранения). Логическая упорядоченность также требует хранения только однородных данных. Например, в поле «Фамилия» должны быть записаны только фамилии, а не имена или отчества, и тем более, не названия предприятия, где работает данный человек;
2. Контролируемая избыточность данных (Controlling Redundancy). Каждый объект должен быть описан только один раз. Данные должны быть избыточными настолько, насколько это требуется для нормального функционирования системы (некоторая избыточность требуется для повышения эффективности работы, ускорения поиска и восстановления информации после сбоев);
3. Целостность (Data Integrity) подразумевает правильность данных (Data Validity) в любой момент времени;
4. Физическая и логическая независимость (Physical and Logical Independence). Логическая независимость предполагает независимость приложений (представления, т.е. интерфейса данных) от изменения логической структуры данных. Например, добавление или удаление некоторых характеристик описываемых объектов и связей между ними не должно привести к необходимости переделки значительной части БД. Физический уровень независимости данных предполагает

независимость данных от их конкретного размещения на физических носителях, их типа, организации и способа доступа;

5. Безопасность и секретность. Под безопасностью данных понимают защиту данных от случайного или преднамеренного доступа к ним лиц, не имеющих на это право, от неавторизированной модификации данных или от их разрушения. Секретность определяется как право отдельных лиц или организаций решать, как, когда и какое количество соответствующей информации может быть передано другим лицам или организациям.

Кроме классических БД, предназначенных в основном для хранения текстовой и числовой информации, в последнее время получили развитие новые направления применения баз данных:

- БД мультимедиа (multimedia databases);
- географические информационные системы (geographic information systems - GIS) – предназначенные для хранения карт, снимков со спутников и погодных данных;
- аналитическая обработка данных (Data warehouses, Data mining and on-line analytical processing – OLAP). Системы, используемые для выделения и анализа необходимой информации из очень больших БД для принятия решений;
- системы реального времени и активные базы данных (Real-time and active database technology) используются для контроля проектирования и производства;
- интернет БД включающие серверные БД и поисковые системы.

Работу с БД можно условно разбить на несколько этапов:

Первый этап заключается в определении структуры данных и имплементации ее с помощью некоторой СУБД. В реляционных БД информация сохраняется в связанных таблицах.

Второй этап заключается во вводе данных. Использование специальных форм упрощает ввод-вывод отдельных записей, ограничивая «рабочую зону» пользователя, сконцентрировав его внимание на требуемых данных.

Третий этап – поиск информации либо ее выборка. Осуществляется на основе критериев поиска. Совокупность критериев поиска, предназначенных для поиска и выделения информации, называется запросом.

Четвертый этап заключается в оформлении выбранных записей. Записи могут быть представлены в виде таблиц, форм, графиков, диаграмм и могут быть оформлены в виде отчетов.

Таким образом, можно выделить основные компоненты БД. Это таблицы (для реляционных БД), запросы, формы и отчеты. Сюда также необходимо добавить и программы, связывающие эти компоненты воедино в приложение.

1.1. Отличие от файловой системы хранения данных

Базы данных появились тогда, когда традиционная файловая система хранения перестала удовлетворять нарастающему потоку данных и требованиям по их обработке. Давайте рассмотрим, чем же отличается способ хранения данных в БД от принятого в файловой системе.

Первое отличие состоит в том, что БД хранит данные в упорядоченном и специально организованном виде, исключающим совместное хранения разнородной информации. К примеру, в текстовых файлах на порядок размещения данных не накладывается сколько-нибудь серьезных ограничений, и данные могут быть расположены произвольно. В электронных таблицах данные уже располагаются упорядоченно по строкам и столбцам, но все еще достаточно произвольно. Человек сам решает в момент создания таблицы как лучше и нагляднее разместить данные. И лишь в базах данных структура данных строго фиксирована (если определен столбец «Фамилия», то в него должны записываться только фамилии, а не имена и т.п.) и определяется стандартом используемой модели данных.

Второе отличие состоит в том, что БД хранит не только данные, но и описание структуры данных (мета-данные). Мета-данные хранятся отдельно от самих данных в так называемом словаре (системном каталоге) данных. Т.о. любая СУБД может работать с разными наборами данных, поскольку структура хранения данных доступна при чтении этих данных. В файловой системе способ хранения данных – дело каждой программы, осуществляющей хранение и обработку данных. Структура данных встроена в программу доступа и не может быть прочитана другими программами.

В объектно-ориентированных и объектно-реляционных БД можно определить операции над данными как часть определения данных. Операция (также называемая функцией) состоит из двух частей: интерфейса (или подпись – signature) включающего ее имя со списком аргументов и тела (имплементации – method). Имплементация хранится отдельно и может быть изменена независимо от интерфейса. Таким образом, использование мета-данных проявляется в:

- 1) независимости программ и данных (program-data independence);
- 2) независимости программ и операций (program-operation independence).

Третьим серьезным отличием БД от файловой системы хранения является наличие расширенных средств поиска информации. Практически всегда в БД применяют индексированное хранение информации. Наряду с упорядоченностью данных индексированное хранение информации дает многократное повышение скорости поиска. Фактически появление БД было вызвано невозможностью дальнейшего повышения скорости поиска в файловых системах с ростом объема хранимой информации.

Соответственно, БД снимают следующие ограничения, присущи файловой системе хранения данных:

- разделение и изоляция данных;
- дублирование данных;
- зависимость от данных;
- несовместимость файлов и программ доступа.

1.2. Архитектура построения баз данных

Так как любая упорядоченность накладывает серьезные ограничения на способ хранения и использования данных, то были предприняты действия, направленные на повышение гибкости доступа к данным. Результатом этих действий стала предложенная в 1978 году (Tsichritzis and Klug) трехуровневая архитектура (смотри рис. 2) построения баз данных. Данная схема была разработана как стандарт представления данных (ANSI/SPARC) и в настоящий момент ее поддерживают большинство коммерческих СУБД. Цель трехуровневой архитектуры заключается в отделении пользовательского представления базы данных от ее физического представления, т.е. обеспечении независимости от данных.

Первый уровень. Внутренний (internal) уровень. Определяется физической моделью данных, которая описывает размещение данных на физических носителях и способы доступа к ним, структуру файлов, индексов и отдельных информационных единиц;

Второй уровень. Концептуальный (conceptual) уровень. Определяется концептуальной моделью данных, которая описывает логическую структуру данных без указания деталей их физического хранения;

Третий уровень. Внешний уровень или уровень представлений (интерфейса) выводит необходимые данные в требуемом формате, скрывая остальную часть БД от пользователей. Внешнее представление – это содержимое БД, каким его видит определенный пользователь. Одному пользователю нужны сведения о товарах и их размещение на складе и он понятия может не иметь, что в БД еще хранится информация о клиентах, поставщиках и т.д. Пользователь может также изменять свое представление, не оказывая влияние на представления других пользователей. Внешний

уровень предоставляет также свободу выбора языка общения с БД. Конечный пользователь может пользоваться языком интерфейса, т.е. использовать меню и другие запрограммированные действия. Опытный пользователь может воспользоваться языком запросов SQL. Системный программист может использовать расширение SQL и т.д.

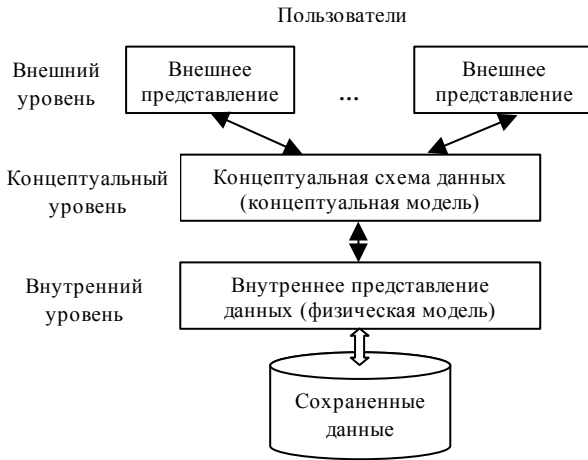


Рис. 2. Трехуровневая архитектура построения БД

Все эти уровни связаны между собой программами отображения одного уровня в другой путем трансляции запросов. Запрос от конечного пользователя на требуемые данные должен быть интерпретирован на концептуальном уровне и затем преобразован в конечный запрос на извлечение требуемых данных

на физическом уровне. Затем эти данные должны быть преобразованы к виду, запрашиваемому пользователем. Все эти уровни связаны между собой программами отображения одного уровня в другой путем трансляции запросов. Запрос от конечного пользователя на требуемые данные должен быть интерпретирован на концептуальном уровне и затем преобразован в конечный запрос на извлечение требуемых данных на физическом уровне. Затем эти данные должны быть преобразованы к виду, запрашиваемому пользователем.

Таким образом, можно выделить пять основных характеристик, определяющие БД как способ хранения и доступа к данным:

1. Хранение однородных упорядоченных данных;
2. Использования мета-данных для хранения описания БД.
3. Независимость программ и данных. (независимость программ и данных и независимость программ и операций);
4. Поддержка разнообразных способов отображения (представления) и выборки данных.
5. Наличие расширенных средств поиска информации.

1.3. Свойства баз данных

1.3.1. Независимость данных

Независимость данных должна обеспечиваться на логическом и физическом уровнях. Логическая независимость предполагает независимость приложений (внешнего представления) от изменения логической структуры данных. Схема, представленная на хорошо поясняет независимость данных. Видим, что независимость данных на логическом уровне обеспечивается тем, что внешнее представление данных развязано от способа организации данных. Поэтому изменения на первом и втором уровнях не повлияют на конечное представление данных, т.е. однажды разработанный интерфейс не придется переписывать заново.

Физический уровень независимости данных предполагает независимость данных от их конкретного размещения на физических носителях, их типа, организации и способа доступа. Независимость данных на этом уровне обеспечивается развязкой первого и второго уровней. При изменении физической модели не потребуется производить изменения на концептуальном и внешнем уровнях. В настоящее время этот уровень независимости обеспечивается СУБД и операционной системой, хотя на ранних порах развития баз данных представлял собой большую проблему. Для хранения данных применялись накопители на магнитных лентах, причем не существовало единых стандартов размещения данных на лентах и стандартов управления накопителями. Поэтому при смене типа носителя приходилось полностью переписывать все программы доступа к данным.

1.3.2. Контролирование избыточности данных

Избыточность данных в первую очередь связана с хранением повторяющейся информации. Наличие повторяющейся информации не только приводит к увеличению размера БД, но и приводит к возможности появления несогласованного состояния данных. Рассмотрим пример. Нам необходимо хранить информацию о покупках, сделанных в кредит. Тогда для каждой покупки отдельного покупателя придется повторно вводить его персональные данные, например ФИО и адрес. При покупке одинаковых товаров различными покупателями придется каждый раз дублировать описание товара. Такое дублирование информации приводит к следующим серьезным проблемам:

- а) приходится тратить лишние усилия на ввод повторяющихся данных;
- б) при вводе повторяющихся данных возрастает вероятность ошибки набора;
- в) при изменении сведений об объекте необходимо корректировать все записи, содержащие сведения об объекте;

- d) согласованность данных может быть нарушена вследствие наличия отличающихся копий одинаковых данных, либо при обновлении не всех повторяющихся данных (аномалия модификации), либо при непреднамеренном удалении информации, которая напрямую не связана с удаляемыми данными (аномалия удаления); при добавлении неполных данных (аномалия добавления).
- e) неоправданное увеличение размера данных приведет к снижению скорости выполнения запросов, поиска и т.п.

Основное требование контролируемой избыточности данных заключается в том, что каждая логическая единица данных должна быть сохранена лишь однажды. Данные должны быть избыточными настолько, насколько это требуется для нормальной работы системы, так как некоторая избыточность требуется для повышения эффективности работы, ускорения поиска и восстановления информации после сбоев. Хранение согласованных данных лежит в основе следующего свойства БД.

1.3.3. Обеспечение целостности и правильности данных

Обеспечение целостности БД составляет необходимое условие успешного функционирования БД. Целостность БД есть свойство базы данных, означающее, что в ней содержится полная, непротиворечивая, согласованная и адекватно отражающая предметную область информация. Мы привыкли доверять данным, помещаемым в печатных изданиях. При подготовке книги размещаемые в ней данные проверяют несколько редакторов. Они же стараются сделать так, чтобы книга была написана грамотно и соответствовала нормам. При общении с компьютером соблюдается все тот же принцип, т.е. компьютер сравнивается с известным издательством. Если что-то сделано на компьютере, то люди этому слепо верят, мотивируя тем, что компьютер не может ошибаться. Но это не совсем так. О достоверности данных придется забыть, если в компьютер будут помещены некорректные данные или они могут стать некорректными вследствие сбоев и ошибок.

Поддержание целостности БД включает проверку целостности и ее восстановление в случае обнаружения противоречий в БД. Целостное состояние БД описывается с помощью ограничителей целостности в виде условий, которым должны удовлетворять хранимые в базе данные. Ограничители целостности (constraints) бывают трех типов:

- ограничители значений. К ним относятся:
 - задание типа и формата, которые позволяют ввод только определенных данных;
 - задание диапазона значений;

- задание списка значений;
- ссылочная целостность. Обеспечивается контролем отношений между связанными данными и введением каскадного удаления и обновления связанных записей;
- целостность записи. Обеспечивается проверкой на уникальность некоторых данных и объявлением обязательных данных.

Другим важным механизмом поддержания целостности является введение транзакций. Транзакцией называется некоторая неделимая последовательность операций над данными БД, которая отслеживается от начала и до завершения. Если по каким-либо причинам (сбои и ошибки) транзакция остается незавершенной, то производится отмена всех операций, входящих в ее состав. Транзакции присущи следующие свойства:

- атомарность (выполняются все входящие в транзакцию операции или ни одна);
- согласованность (любая транзакция должна переводить БД из одного согласованного состояния в другое согласованное состояние);
- изолированность (транзакции выполняются независимо друг от друга);
- безопасность (даже аварийное завершение работы не приводит к потери данных).

Контроль транзакций особенно важен в многопользовательских БД, где транзакции могут быть запущены параллельно. Так как компьютер не может обрабатывать параллельно выполняемые процессы вследствие ограниченности его ресурсов (один центральный процессор), то обычно прибегают к разбиению выполняемых процессов на сравнительно небольшие части и к их поочередному выполнению. Если две или более транзакции читают или модифицируют разные данные, то это не приводит к возникновению каких-либо проблем. Другое дело, когда доступ осуществляется к одним и тем же данным. Тогда порядок выполнения частей транзакций может играть важную роль, так как от него будет зависеть конечное состояние данных.

В последнем случае говорят о сериализации транзакций, т.е. о составлении такого плана их выполнения (сериального плана), при котором суммарный эффект реализации транзакций эквивалентен эффекту их последовательного выполнения. При параллельном выполнении транзакций возможно возникновение конфликтов (блокировок). При обнаружении таких случаев обычно производится откат одной или нескольких транзакций.

1.3.4. Обеспечение безопасности и секретности

Практически всегда БД представляет собой важный ресурс, который должен быть надежно защищен. Потенциальными опасностями являются:

- похищение и фальсификация данных;
- утрата конфиденциальности;
- утрата целостности;
- потеря доступности;
- непредумышленное и умышленное повреждение данных.

В отношении опасностей могут быть предприняты самые разные контрмеры, начиная от компьютерных систем наблюдения, защиты и восстановления, и заканчивая правовыми и административными процедурами.

Обеспечение безопасности достигается защитой объектов БД и программного кода от модификации, запрещением редактирования по умолчанию, поддержкой блокировок, уровней изолированности транзакций и средств восстановления информации после сбоев (создание контрольных точек, ведение журнала, протоколирование и создание архивных копий).

Обеспечение секретности достигается шифрованием программ и данных, защитой паролем, авторизации и аутентификации пользователей, поддержкой различных уровней доступа к БД и отдельным ее элементам (таблицам, формам, отчетам и т.д).

Контрольные вопросы и задания

1. Дайте определение БД.
2. Перечислите свойства БД.
3. Опишите пять основных компонентов БД и кратко поясните, для чего они предназначены.
4. Перечислите основные преимущества использования БД перед традиционной файловой системой хранения данных.
5. Нарисуйте архитектуру построения баз данных ANSI-SPARC. На ее основе поясните понятия логической и физической независимости данных.
6. Сформулируйте основные характеристики, определяющие БД как способ хранения и доступа к данным.
7. С помощью каких средств обеспечивается поддержка целостности данных в БД. Перечислите виды ограничителей целостности.
8. Сформулируйте основное требование контролируемой избыточности данных. Перечислите проблемы, появляющиеся при его нарушении.