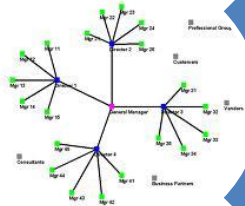


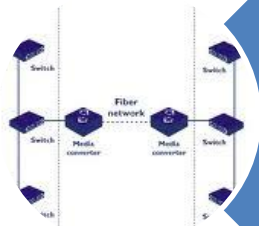
# **ТЕМА №5. МЕХАНИЗМЫ ОБЕСПЕЧЕНИЯ ЦЕЛОСТНОСТИ ДАННЫХ, РЕАЛИЗУЕМЫЕ РЕЛЯЦИОННЫМИ СУБД**

- 1. Типы управляемой базы данных СУБД;*
- 2. Обеспечение целостности базы данных.*

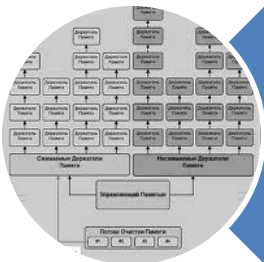
По типу управляемой базы данных СУБД  
разделяются на:



## Иерархические СУБД

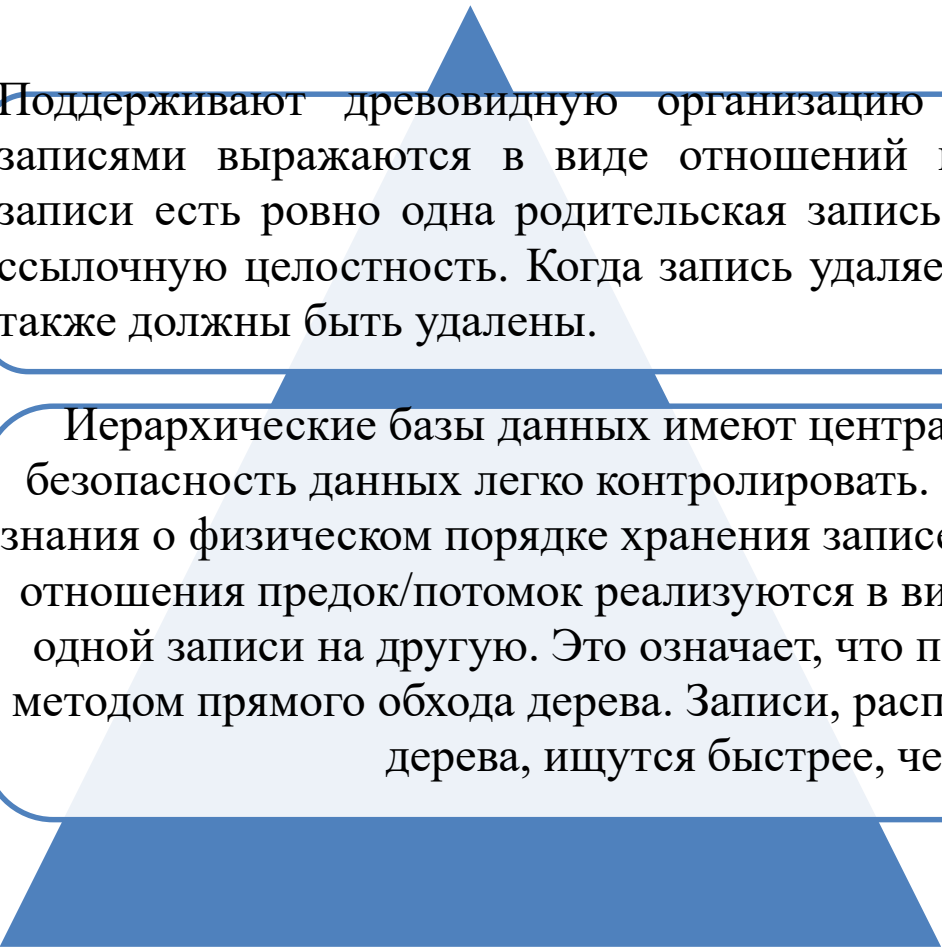


## Сетевые СУБД



## Реляционные СУБД


# Иерархические СУБД



Поддерживают древовидную организацию информации. Связи между записями выражаются в виде отношений предок/потомок, а у каждой записи есть ровно одна родительская запись. Это помогает поддерживать ссылочную целостность. Когда запись удаляется из дерева, все ее потомки также должны быть удалены.

Иерархические базы данных имеют централизованную структуру, т.е. безопасность данных легко контролировать. К сожалению, определенные знания о физическом порядке хранения записей все же необходимы, так как отношения предок/потомок реализуются в виде физических указателей из одной записи на другую. Это означает, что поиск записи осуществляется методом прямого обхода дерева. Записи, расположенные в одной половине дерева, ищутся быстрее, чем в другой.

# Сетевые СУБД



Сетевая модель расширяет иерархическую модель СУБД, позволяя группировать связи между записями в множества. С логической точки зрения связь — это не сама запись. Связи лишь выражают отношения между записями. Как и в иерархической модели, связи ведут от родительской записи к дочерней, но на этот раз поддерживается множественное наследование.

Следуя спецификации CODASYL, сетевая модель поддерживает DDL (Data Definition Language — язык определения данных) и DML (Data Manipulation Language — язык обработки данных). Это специальные языки, предназначенные для определения структуры базы данных и составления запросов. Несмотря на их наличие программист по-прежнему должен знать структуру базы данных.

В сетевой модели допускаются отношения "многие ко многим", а записи не зависят друг от друга. При удалении записи удаляются и все ее связи, но не сами связанные записи.

В сетевой модели требуется, чтобы связи устанавливались между существующими записями во избежание дублирования и искажения целостности. Данные можно изолировать в соответствующих таблицах и связать с записями в других таблицах.

Оптимальную структуру базы данных сложно сформировать, а готовую структуру трудно менять. Если вид таблицы претерпевает изменения, все отношения с другими таблицами должны быть установлены заново, чтобы не нарушилась целостность данных. Сложность подобной задачи приводит к тому, что программисты зачастую отменяют некоторые ограничения целостности ради упрощения приложений.

# Реляционные СУБД

В сравнении с рассмотренными выше моделями реляционная модель требует от сервера СУБД гораздо более высокого уровня сложности. В ней делается попытка избавить программиста от выполнения рутинных операций по управлению данными, столь характерных для иерархической и сетевой моделей.

В реляционной модели база данных представляет собой централизованное хранилище таблиц, обеспечивающее безопасный одновременный доступ к информации со стороны многих пользователей. В строках таблиц часть полей содержит данные, относящиеся непосредственно к записи, а часть — ссылки на записи других таблиц. Таким образом, связи между записями являются неотъемлемым свойством реляционной модели.

Каждая запись таблицы имеет одинаковую структуру. Например, в таблице, содержащей описания автомобилей, у всех записей будет один и тот же набор полей: производитель, модель, год выпуска, пробег и т.д. Такие таблицы легко изображать в графическом виде.

В реляционной модели СУБД достигается информационная и структурная независимость. Записи не связаны между собой настолько, чтобы изменение одной из них затронуло остальные, а измененная структура СУБД, базы данных не обязательно приводит к перекомпиляции работающих с ней приложений.

В реляционных СУБД применяется язык SQL, позволяющий формулировать произвольные, нерегламентированные запросы. Это язык четвертого поколения, поэтому любой пользователь может быстро научиться составлять запросы. К тому же, существует множество приложений, позволяющих строить логические схемы запросов в графическом виде. Все это происходит за счет ужесточения требований к производительности компьютеров. К счастью, современные вычислительные мощности более чем адекватны.

# ***Обеспечение целостности базы данных***

---

Важнейшим вопросом, возникающим при проектировании и эксплуатации систем обработки данных, является обеспечение целостности данных. Нарушение целостности может быть связано не только с ошибками оператора при вводе данных, но и со сбоями в работе системы. Отказ в работе системы может возникнуть, когда одни операции были выполнены, а другие, связанные с ними определенными ограничениями операции, не успели выполниться.

---

Решение проблемы целостности заключается в обеспечении в любой момент времени правильности данных в базе данных, для этого используют правила проверки достоверности данных гарантирующие, что недействительные данные не попадут в таблицы БД. Целостность данных обеспечивается набором специальных предложений, называемых ограничениями целостности.

---

Ограничения – это правила, применимые во время или после создания таблиц, распространяемые на то, как эти таблицы могут заполняться.

---

Ограничения целостности представляют собой утверждения о допустимых значениях отдельных информационных единиц и связях между ними.

---

Ограничение целостности устанавливает правила на уровне БД, определяя набор проверок для таблиц системы. Эти проверки автоматически выполняются всякий раз, когда вызывается оператор вставки, модификации или удаления данных в таблице. Если какие-либо ограничения нарушены, операторы отменяются.

---

Ограничения целостности определяются в большинстве случаев особенностями предметной области, хотя могут отражать и чисто информационные характеристики. Ограничения целостности могут относиться к разным информационным объектам атрибутам (полям), кортежам (строкам, записям), отношениям (таблицам, файлам), связям между файлами и т. п. Для простоты изложения будем далее пользоваться в основном терминологией реляционных СУБД. Однако все сказанное ниже относится и к СУБД других классов.

Для полей чаще всего используются следующие виды ограничений.

