

ГАПОУ "Камский строительный колледж  
имени Е.Н.Батенчука"

**Учебная дисциплина**  
**СИСТЕМЫ УПРАВЛЕНИЯ БАЗЫ ДАННЫХ**

**Лекция 3. Проектирование структуры БД**

Руководитель занятия:  
преподаватель спецдисциплин

Еремина И.И.

# Подходы проектирования БД

В зависимости от содержания исходной информации, положенной в основу проектирования, различают подходы

**«от предметной области»**

**«от запроса».**

**Подход от предметной области означает описание объектов отображаемой в ИС части реального мира и связей между ними безотносительно к потребностям пользователей.** Иногда этот подход называют объектным или непроцессным.

**В подходе от запроса основным источником информации являются запросы пользователей (задачи). Этот подход называется также процессным или функциональным.**

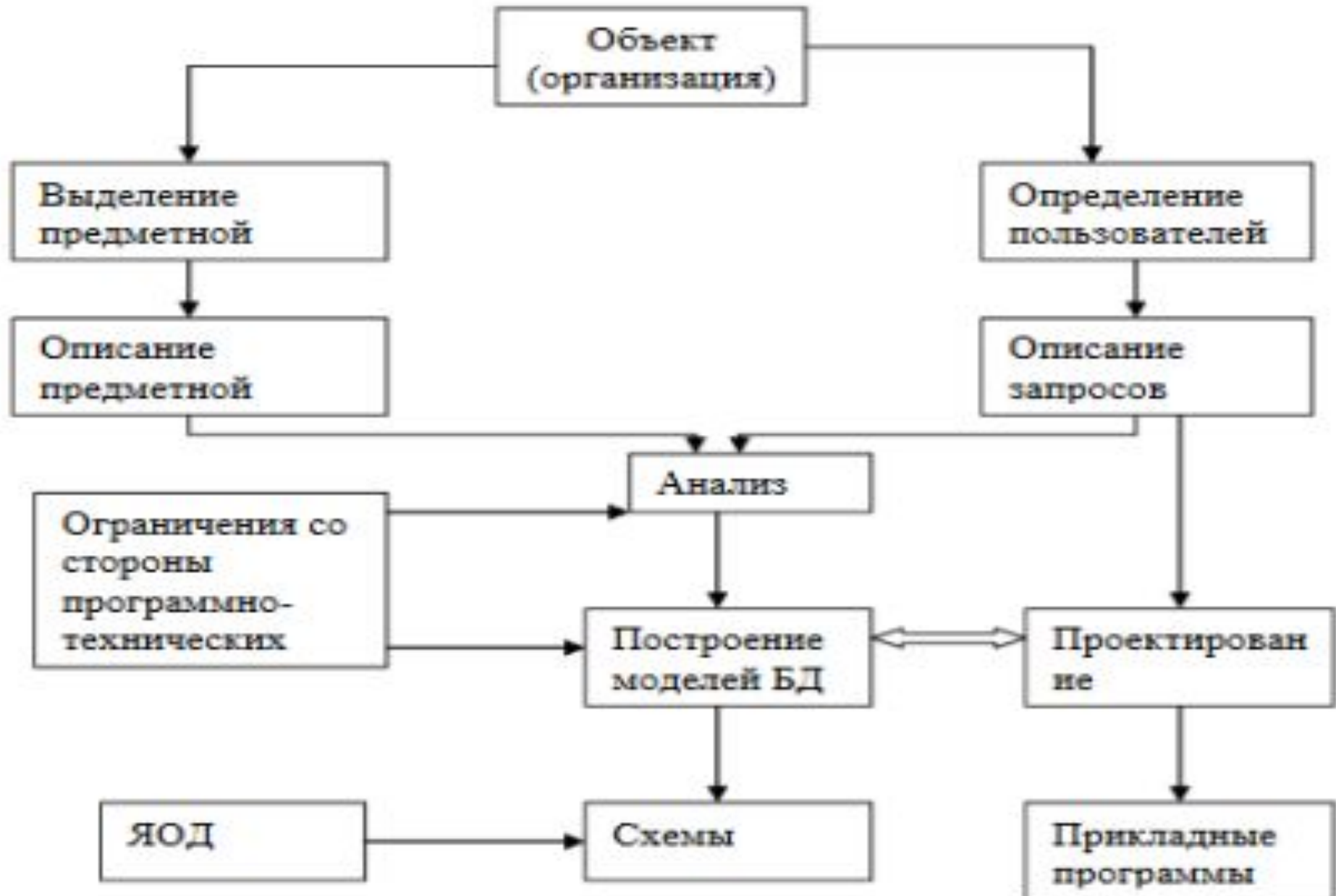
Преимуществом подхода от предметной области является его объективность, системное отображение предметной области и, как следствие, устойчивость информационной модели, возможность реализации большого числа приложений, в том числе и заранее незапланированных, на созданной базе данных.

**Недостатком подхода является трудность отбора информации,**

Функциональный подход ориентирован на реализацию текущих запросов пользователей и не учитывает перспектив развития системы. при его использовании могут возникнуть трудности в объединении взглядов различных пользователей. Однако учет запросов позволяет улучшить характеристики функционирования БД.

Отдельно взятый ни один из указанных подходов не дает достаточной информации для проектирования рациональной БД. Целесообразно совместное применение обоих подходов с ведущим положением объектного подхода.

# Укрупненная схема проектирования БД



# Взаимосвязь этапов проектирования структуры БД

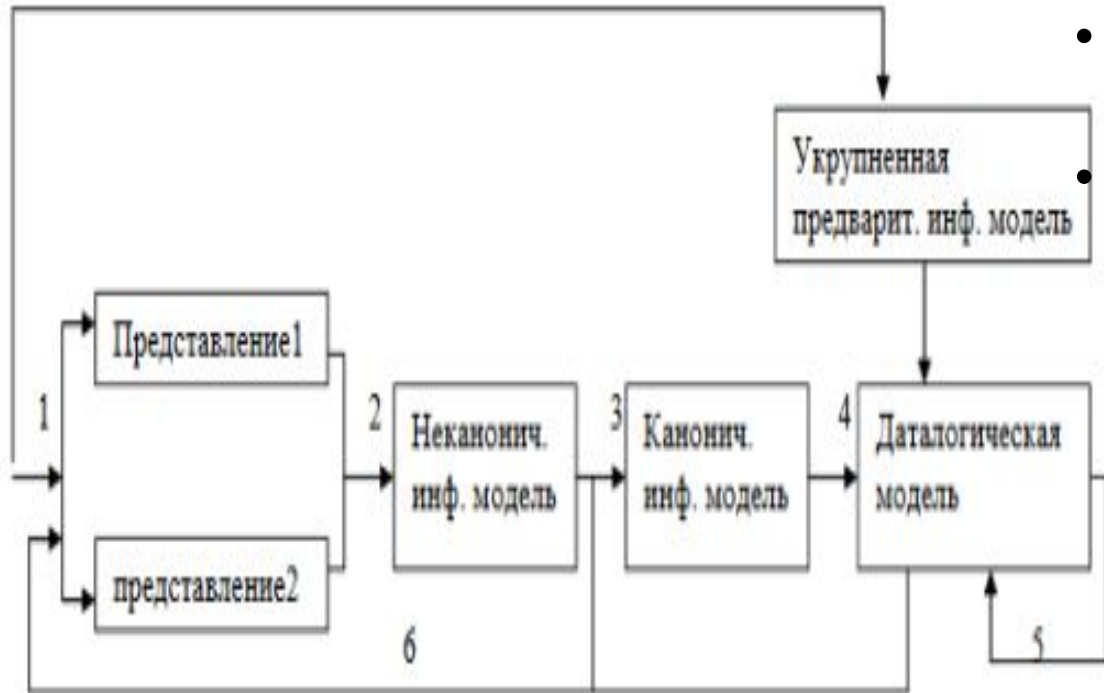


Процесс проектирования структуры БД носит итеративный характер. Обычно сначала строится

- **предварительная логическая структура**, которая в общем виде отображает предметную область и функциональные требования,
- детальное **логическое структурирование** - Даталогическая модель, построенная от предметной области (возможность и эффективность выполнения заданных функций). В процессе анализа даталогическая модель может перестраиваться,
- **физическое моделирование**. Полученный вариант физической структуры проверяют на ограничение по ресурсам. В случае несоблюдения ограничений производят перепроектирование физической структуры. Если этого оказывается недостаточно, то возвращаются к даталогическому уровню и выполняют еще одну итерацию по проектированию структуры БД.

# Схема проектирования логической структуры БД при использовании подхода от запроса

Предметная область



При реализации подхода от запроса на начальной стадии выявляются

- Локальные представления о предметной области каждого пользователя.
- Глобальная инфологическая модель (первоначальный вариант модели неканонический: он может содержать различные аномалии, например, транзитивные связи, циклы, противоречия, вызванные расхождением во взглядах различных пользователей на одни и те же данные, и др.)

**Инфологическая модель, в которой устранены аномалии, называется канонической инфологической моделью.**

Дальнейшие шаги проектирования в подходах «от

# Алгоритм проектирования логической структуры БД при использовании подхода от запроса

- 1-описание взглядов отдельных пользователей;
- 2-обобщение взглядов пользователей;
- 3-устранение противоречивости, нормализация, минимизация модели;
- 4-преобразование модели к виду, воспринимаемому СУБД;
- 5-проверка адекватности даталогической модели предметной области;
- 6-уточнение взглядов пользователей.

**Проектирование баз данных** – многовариантный процесс. Если на начальных стадиях проектирования трудно выбрать единственный вариант проектного решения, то довести до стадии моделирования и



# ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ СУБД

К основным функциям СУБД относятся:

- Ведение системного каталога, доступного конечным пользователям
- Поддержка транзакций
- Поддержка параллельной работы
- Восстановление базы данных после сбоев
- Контроль доступа к данным
- Поддержка обмена данными
- Поддержка целостности данных
- Поддержка независимости от данных
- вспомогательные функции

# Ведение системного каталога

Системный каталог, или словарь данных, является хранилищем информации, описывающей данные в базе данных (по сути, это "данные о данных", или метаданные). Обычно в системном каталоге хранятся следующие сведения:

- имена, типы и размеры элементов данных;
- Имена связей;
- Накладываемые на данные ограничения поддержки целостности;
- Имена санкционированных пользователей, которым предоставлено право доступа к данным;
- внешняя, концептуальная и внутренняя схемы и отображения между ними;
- Статистические данные, например частота транзакций и счетчики обращений к объектам базы данных.

Наличие системного каталога позволяет:

- централизованно хранить информацию о данных, что обеспечивает контроль доступа к этим данным и любому другому ресурсу;
- легко обнаружить избыточность и противоречивость описания отдельных элементов данных;
- протоколировать внесение в базу данных изменений и определить их последствия еще до их внесения, поскольку в системном каталоге зафиксированы все существующие элементы данных, установленные между ними связи, а также все их

# Поддержка транзакций

Транзакция представляет собой набор действий , выполняемых отдельным пользователем или прикладной программой с целью доступа или изменения содержимого базы данных .

Примеры транзакций

- ✓ добавление в базу данных сведений о новом сотруднике ,
- ✓ обновление сведений о зарплате некоторого сотрудника ,
- ✓ удаление сведений о сотруднике .

Если во время выполнения транзакции произойдет сбой , например из-за выхода из строя компьютера , база данных попадает в противоречивое состояние, поскольку некоторые изменения уже будут внесены , а остальные еще нет . Поэтому все частичные изменения должны быть отменены для возвращения базы данных в прежнее, непротиворечивое состояние .

**СУБД должна иметь механизм, который гарантирует выполнение либо всех операций обновления данной транзакции , либо ни одной из них .**

# Поддержка параллельной работы

*СУБД должна иметь механизм, который гарантирует корректное обновление базы данных при параллельном выполнении операций обновления многими пользователями.*

Параллельный доступ сравнительно просто организовать, если все пользователи выполняют только чтение данных, поскольку в этом случае они не могут помешать друг другу .

Однако, когда два или больше пользователей одновременно получают доступ к базе данных , легко может возникнуть конфликт с нежелательными последствиями .

# Восстановление базы данных после сбоя

СУБД должна предоставлять средства восстановления базы данных на случай какого - либо ее повреждения или разрушения . Подобный сбой может произойти в результате выхода из строя системы или запоминающего устройства , ошибки аппаратного или программного обеспечения , которые могут привести к останову СУБД . Кроме того, пользователь может обнаружить ошибку во время выполнения транзакции и потребовать ее отмены. Во всех этих случаях СУБД должна предоставить механизм восстановления базы данных и возврата ее к непротиворечивому состоянию .

Для восстановления БД нужно располагать некоторой дополнительной информацией, т.е. поддержание надежности хранения данных в БД требует избыточности хранения данных , причем та часть данных , которая используется для восстановления , должна храниться особо надежно . Наиболее распространенным методом **поддержания** такой **избыточной информации** является **ведение журнала изменений БД** .

**Журнал - это особая часть БД** , недоступная пользователям СУБД и поддерживаемая с особой тщательностью ( иногда поддерживаются две копии журнала , располагаемые на разных физических дисках ), в которую поступают записи обо всех изменениях основной части БД

# Контроль доступа к данным

## Поддержка обмена данными

**Контроль доступа к данным.** СУБД должна иметь механизм, гарантирующий возможность доступа к базе данных только санкционированных пользователей.

**Поддержка обмена данными.** СУБД в должны поддерживать работу в локальной сети, чтобы вместо нескольких разрозненных баз данных для каждого отдельного пользователя можно было бы установить одну централизованную базу данных и использовать ее как общий ресурс для всех существующих пользователей. При этом предполагается , что не база данных должна быть распределена в сети, а удаленные пользователи должны иметь возможность доступа к централизованной базе данных . Такая топология называется распределенной обработкой.

# Поддержка целостности данных

Целостность базы данных означает корректность и непротиворечивость хранимых данных. Она может рассматриваться как еще один тип защиты базы данных. Целостность обычно выражается в виде ограничений или правил сохранения непротиворечивости данных, которые не должны нарушаться в базе. СУБД должна обладать инструментами контроля за тем, чтобы данные и их изменения соответствовали заданным правилам.



# Поддержка независимости от данных

Независимость от данных обычно достигается за счет реализации механизма поддержки представлений или подсхем. Физическая независимость от данных достигается довольно просто, так как обычно имеется несколько типов допустимых изменений физических характеристик базы данных, которые никак не влияют на представления. СУБД должна обладать инструментами поддержки независимости программ от фактической структуры базы данных.



# Вспомогательные функции

- ❖ Администрирование базы данных ,
- ❖ импорт и экспорт БД ,
- ❖ мониторинг характеристик функционирования и использования базы данных ,
- ❖ статистический анализ ( оценка производительности или степени использования базы данных ),
- ❖ реорганизация индексов,
- ❖ перераспределение памяти.

Сегодня считают, что все формы , отчеты , меню , как и программный код , содержащийся в них , входят в приложение БД .

Автономные программы , вызывающие СУБД , также являются частью приложения . Все структуры , правила и ограничения , касающиеся таблицы , а также определения связей относятся к ведению СУБД и входят в состав БД .

Приложения выполняют **пять основных функций** :

1. **Создание , чтение , обновление и удаление представлений .**
2. **Форматирование представлений .**
3. **Реализация ограничений .**
4. **Обеспечение механизмов безопасности и контроля .**

# Производительность СУБД

Оценивается :

- временем выполнения запросов;
- скоростью поиска информации в неиндексированных полях;
- временем выполнения операций импортирования базы данных из других форматов;
- скоростью создания индексов и выполнения таких массовых операций, как обновление , вставка , удаление данных;
- максимальным числом параллельных обращений к данным в многопользовательском режиме;
- временем генерации отчета .

На производительность СУБД оказывают влияние два фактора:

1. **СУБД**, которые следят за соблюдением **целостности данных** , несут дополнительную нагрузку , которую не испытывают другие программы ;
2. **производительность собственных прикладных программ** сильно зависит от правильного проектирования и построения базы данных .

# Недостатки моделирования структуры БД при помощи алгоритма нормализации

1. Первоначальное размещение всех атрибутов в одном отношении является очень **неестественной операцией**. Интуитивно разработчик сразу проектирует несколько отношений в соответствии с обнаруженными сущностями. Даже если совершить насилие над собой и создать одно или несколько отношений, включив в них все предполагаемые атрибуты, то совершенно неясен смысл полученного отношения.
2. Невозможно сразу определить **полный список атрибутов**. Пользователи имеют привычку называть разными именами одни и те же вещи или наоборот, называть одними именами разные вещи.
3. Для проведения процедуры **нормализации** необходимо выделить **зависимости атрибутов**, что тоже очень нелегко, т.к. необходимо явно выделить все зависимости, даже те, которые

# **Тесты для самоконтроля**

**1. Основным объектом для хранения информации в реляционных базах данных является:**

- a) таблица
- b) запрос
- c) форма
- d) отчет

Ответ:

## 2. Проектированием БД занимается :

- a) Администратор БД
- b) Программист БД
- c) Пользователь БД
- d) Проектировщик БД
- e) Нет правильного ответа

Ответ:

### **3. Выберите правильный порядок действий при проектировании БД**

- а) Решение проблемы передачи данных**
- б) Анализ предметной области, с учетом требования конечных пользователей**
- в) Формализация представления данных в БД**
- г) Обобщенное описание БД с использованием естественного языка, математических формул, графиков и других средств**

**а) б, г, в, а**

**б) а, б, г, в**

**в) а, б, в, г**

**г) г, б, в, а**

**е) Порядок действий значения не имеет**

**Ответ:**

## 4. Реляционная база данных представлена в виде

- a) взаимосвязанных двумерных таблиц
- b) взаимосвязанных трехмерных таблиц
- c) сетевой структуры данных
- d) элементов схемы данных
- e) произвольной структуры данных



**5. Структура данных, для которой характерна подчиненность объектов нижнего уровня объектам верхнего уровня, называется**

- a) иерархической**
- b) табличной**
- c) реляционной**
- d) логической**
- e) сетевой**

## **6. Система управления базами данных (СУБД):**

- a) MS Access.
- b) MS Excel.
- c) MS Power Point.
- d) MS Outlook.
- e) MS Word.

**7. Один атрибут или минимальный набор из нескольких атрибутов, значения которых в одно и тоже время не бывают одинаковыми, то есть однозначно определяют запись таблицы - это**

- a) Первичный ключ
- b) Внешний ключ
- c) Индекс
- d) Степень отношения
- e) Нет правильного варианта

## 8. Ключ называется сложным, если состоит

- a) Из нескольких атрибутов
- b) Из нескольких записей
- c) Из одного атрибута
- d) Из одного атрибута, длина значения которого больше заданного количества символов
- e) Нет правильного варианта

**9. Средство ускорения операции поиска записей в таблице, а, следовательно, и других операций использующих поиск называется**

- a) Индекс
- b) Хеш-код
- c) Первичный ключ
- d) Внешний ключ
- e) Нет верного варианта

**10. Таблица называется индексированной, если для неё используется**

- a) Индекс
- b) Хеш-код
- c) Первичный ключ
- d) Внешний ключ
- e) Нет верного варианта