Тема 1. Функциональные возможности ГИС

Занятие № 3. Базы данных и системы управления ими в ГИС

Цель занятия: **студент должен получить представление о базах данных в ГИС и системе управления ими.**

ПЛАН ЛЕКЦИИ

Ведение

1. База данных в ГИС и требования к ней.

2. Структуры баз данных в ГИС.

3. Системы управления базами данных в ГИС.

4. Заключение.

Список литературы

1. Капралов, Е.Г. Основы геоинформатики. В двух книгах. [Текст] / Е. Г. Капралов, А.В. Кошкарев, В.А. Тикунов / Под ред. В.С. Тикунова. – М.: Академия, 2004. Книга 1, гл. 4, § 4.

ПЕРЕЧЕНЬ ВОПРОСОВ К ЗАЧЕТУ

Понятие базы данных. Разновидности баз данных.

Понятие и функции СУБД. Особенности применения СУБД в ГИС.

ВВЕДЕНИЕ

Совокупность цифровых данных о пространственных объектах образует множество пространственных данных и составляет содержание баз географических данных. Сегодня вы познакомитесь со структурой баз данных и СУБД в ГИС.

1. База данных в ГИС и требования к ней

База данных - совокупность цифровых данных, организованных по определённым правилам, устанавливающим общие принципы описания, хранения и манипулирования данными.

Базы данных ГИС (пространственные базы данных) содержат наборы данных о пространственных объектах.

Пользователь ГИС видит реальный мир через призму тематической базы данных. Содержащиеся в ней данные должны возможно полнее и точнее отражать описываемый предмет и его характеристики. Представление данных должно учитывать типы их возможных преобразований. В связи с пространственной формой организации базы данных в ГИС, она должна удовлетворять ряду требований.

База данных должна быть:

– согласованной по времени (хранящиеся в ней данные должны соответствовать определённому времени, быть актуальными);

– полной, достаточно подробной для предполагаемого создания ГИС или географической карты; данные должны содержать все необходимые сведения для осуществления анализа или математико-картографического моделирования исследуемого объекта или явления;

– позиционно точной, совместимой с другими данными, которые могут добавляться в неё;

– достоверной, правильно отражающей характер явлений (это требует тщательно отбирать включаемые в неё атрибуты явлений);

– легко обновляемой;

– доступной для любых пользователей.

2. Структуры баз данных в ГИС

Эффективное решение задач пространственного анализа существенно зависит от структуры организации пространственных данных. Это еще один формализм, который означает способ представления множества пространственных данных в памяти компьютера.

Компьютеры не мыслят, как мы, не оперируют непосредственно с визуальными или графическими объектами, данные в компьютерной среде имеют цифровой вид. Эта совокупность цифровых данных о пространственных объектах образует множество пространственных данных и составляет содержание баз географических данных.

Процесс проектирования базы данных (БД) состоит из трёх уровней: концептуального, логического и физического.

На концептуальном уровне разрабатывается концептуальная модель БД, которая включает: описание и определение рассматриваемых объектов; выбор способов представления географических объектов в БД; выбор базовых типов пространственных объектов (точки, линии, полигоны, ячейки растра) и др.

На этом уровне определяется и содержание БД, зависящее от сути рассматриваемого явления, характера его пространственного распространения и задач, для которых создаётся БД.

Задачами могут быть: создание карты, серии карт, карт для многократного или многоцелевого использования. Концептуальный уровень не связан с определёнными программными средствами.

Логический уровень определяется существующими программными средствами и практически не зависит от технического обеспечения. На этом уровне происходит выбор логической структуры элементов БД в соответствии с системой управления базами данных (СУБД), используемой в программном обеспечении.

Наиболее распространёнными логическими структурами (моделями) баз данных являются иерархическая, сетевая и реляционная.

Иерархическая. **Во многих случаях существует взаимосвязь между данными, называемая отношением «один ко многим».**

Это отношение подразумевает, что каждый элемент данных имеет прямую связь с некоторым числом т.н. «потомков» и, конечно, каждый такой потомок, в свою очередь, может иметь связь со своими потомками и т.д. такого рода связь делает доступ к данным простым и эффективным.

Такая модель базы данных хорошо иллюстрируется иерархической системой классификации растений и животных, называемой таксономией. Например, животные делятся на позвоночных и беспозвоночных. В свою очередь, позвоночные имеют подмножество, называемое млекопитающими.

Млекопитающие могут быть разделены на подгруппы. Структура похожа на генеалогическое дерево.

В иерархической модели записи данных образуют древовидную структуру, при этом каждая запись связана только с одной записью, находящейся на более высоком уровне.

Доступ к любой записи осуществляется строго по определенным «веткам» и узлам такого дерева. Ветвление основано на формальных ключевых признаках, которые определяют продвижение по этой структуре от одной ветви к другой.

Иерархические модели хорошо подходят для задач с явно выраженной иерархически соподчиненной структурой информации. Но они обладают низким быстродействием, трудно модифицируемы. Например, если информации о ключевом признаке недостаточно, то вы не сможете продвигаться по дереву.

Главным преимуществом такой системы является то, что в ней легко искать, она хорошо определена, эффективна с точки зрения организации машинной памяти. Такая БД может легко расширяться добавлением новых ветвей и формулированием новых правил ветвления.

Для создания иерархической структуры необходимо знание все-возможных вопросов, которые могут задаваться, поскольку эти вопросы используются как основа для разработки правил ветвления и ключей. Иерархическая модель базы данных показана на рис. 1.

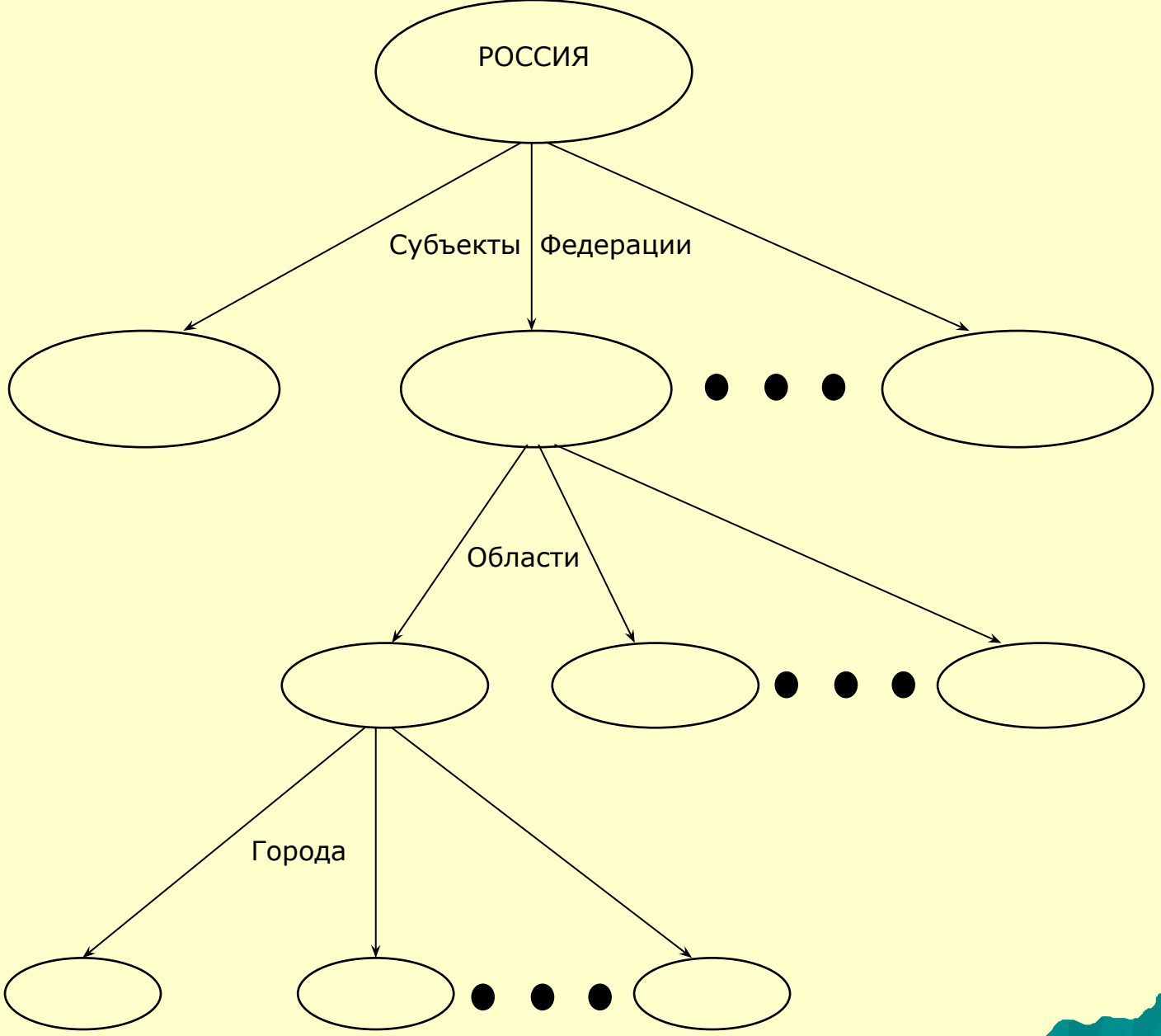


Рис. 1. Иерархическая модель базы данных

Сетевая структура. **Сетевые БД используют отношение «многие к многим», при котором один элемент может иметь многие атрибуты, при этом каждый атрибут связан со многими элементами.**

Например, исследуемый участок земной поверхности может иметь много квадратов, в каждым из которых могут быть связаны несколько животных и растительных видов, причём каждый вид может присутствовать в нескольких квадратах.

Для реализации таких отношений вместе с каждым элементом данных может быть связана специальная переменная, называемая «указателем» (pointer), которая направляет нас ко всем другим элементам данных, связанным с этим.

То есть каждый отдельный элемент данных может быть прямо связан с любым местом базы данных, без введения отношения «предок-потомок».

Указатели определяют местоположение других записей, связанных с ними. Сетевая модель базы данных показана на рис. 2.

Областной центр

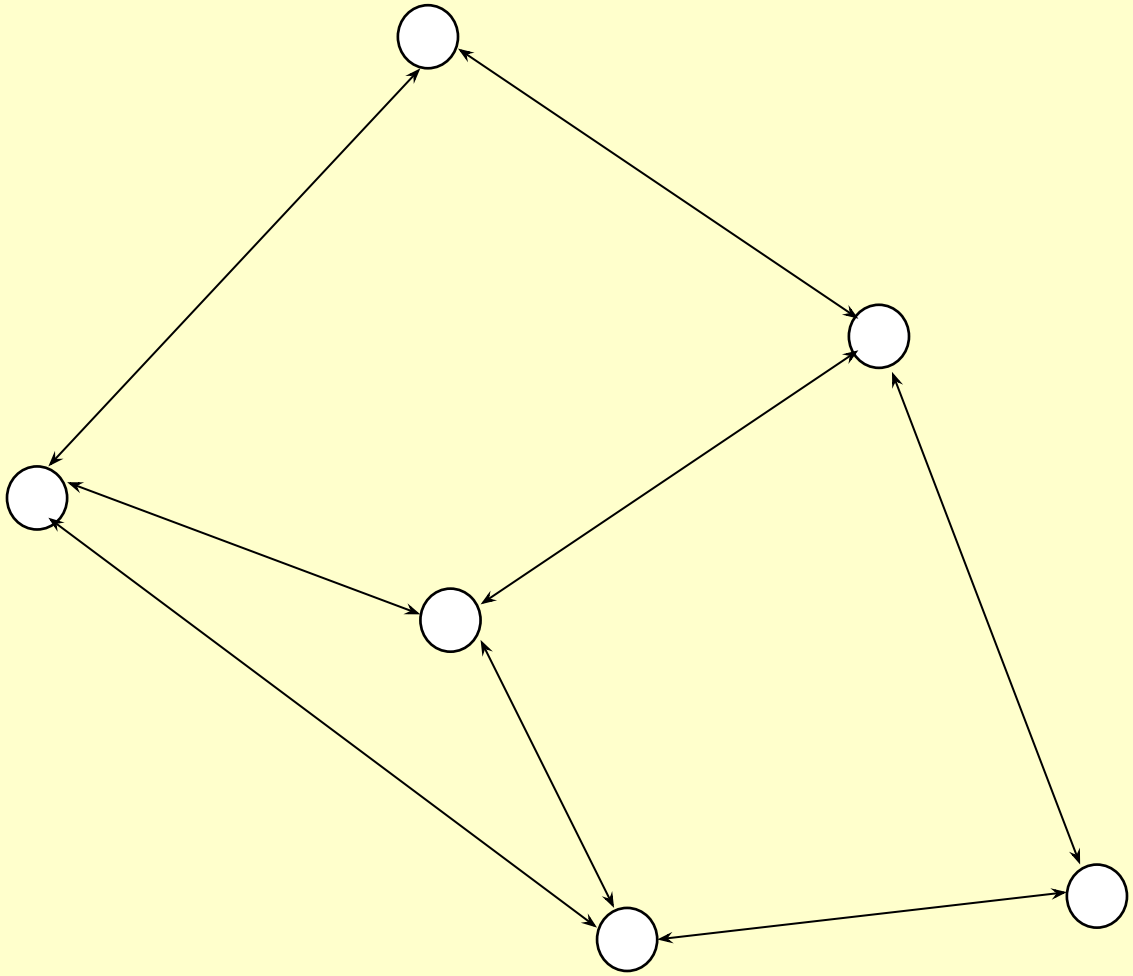


Рис. 2. Сетевая модель базы данных

Сетевые структуры обычно рассматриваются как усовершенствование иерархических структур, т.к. они менее жесткие и могут представлять отношение «многие ко многим». Поэтому они допускают гораздо большую гибкость поиска, нежели иерархические структуры.

Их главным недостатком является то, что в крупных БД ГИС количество указателей может стать очень большим, требуя значительной затраты памяти.

Такие модели БД трудно редактировать, например, удалять записи, так как вместе с данными можно удалить и указатели. Подобные модели хорошо работают в случае решения сетевых, коммуникационных задач.

В иерархической и сетевой моделях БД для поиска конкретной записи сначала необходимо определить путь доступа к этой записи, а затем просмотреть все записи, находящиеся на этом пути.

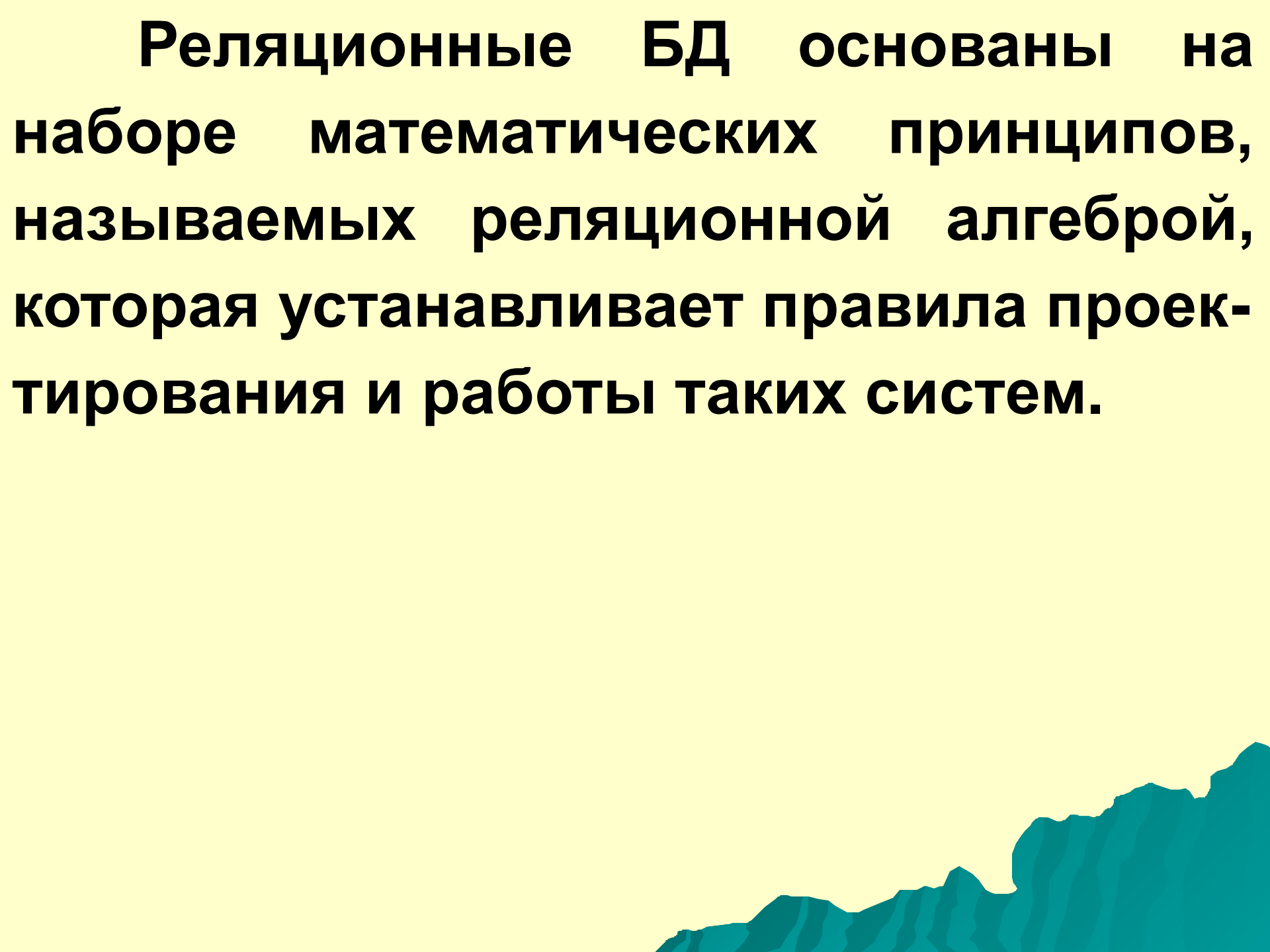
Реляционые БД наиболее популярны. Они имеют табличную структуру: строки соответствуют одной записи об объекте, а столбцы содержат однотипные характеристики всех объектов.

Разнообразные способы индексации данных существенно сокращают время поиска и запроса к данным. Атрибуты объектов группируются в виде т.н. «отношений». Пример реляционной базы данных приведён в табл. 1.

Таблица 1. Пример реляционной базы данных

Город	Численность населения, млн. чел	Координата X	Координата Y	Страна
Амстердам	1,86	4,894833	52,373040	Голландия
Вашингтон	3,22	-76,953830	38,890910	США
Вена	1,88	16,320990	48,202120	Австрия
Лондон	11,1	-0,177998	51,487910	Великобритания
Москва	13,1	37,938250	55,764230	Россия

Реляционные БД основаны на наборе математических принципов, называемых реляционной алгеброй, которая устанавливает правила проектирования и работы таких систем.



Поскольку реляционная алгебра основывается на теории множеств, каждая таблица отношений функционирует как множество, и первое правило гласит, что таблица не может иметь строку, которая полностью совпадает с какой-либо другой строкой.

Поскольку каждая из строк уникальна, одно или несколько полей могут использоваться для определения критерия поиска.

Так, примером использования одного поля может быть выбор уникального личного номера ИНН, номера телефона, дом. адреса и др., которые имеются в других колонках той же таблицы при выборе определенного имени.

Такой критерий поиска называется «первичным ключом» для поиска значений в других полях. Всякая строка таблицы должна иметь уникальное значение в колонке первичного ключа.

Реляционные системы ценны тем, что позволяют собирать данные в простые таблицы, при этом задачи организации данных также просты.

Чтобы мы могли установить реляционные соединения каждая таблица должна иметь хотя бы одну общую колонку с другой таблицей, с которой мы желаем установить такое соединение. Реляционные БД поддерживаются следующими СУБД: dBase, INFO, ORACLE, INFOR-MIX

Физический уровень **связан с аппаратными и программными средствами. Определяется объем хранимой информации, необходимые объёмы оперативной и долговременной памяти компьютера, рассматриваются вопросы структурирования файлов на носителе информации для обеспечения программного доступа к данным, тип представления данных.**

3. Системы управления базами данных в ГИС

Базы данных представляют собой совокупности специальным образом организованных наборов данных, хранящихся на внешних или внутренних носителях.

Сложность работы с множеством файлов, в которых хранятся данные, требует особой организации управления, которое реализуется СУБД.

СУБД – комплекс программ и языковых средств, предназначенных для создания, ведения и использования баз данных.

СУБД выполняют различные операции с данными: ввод, хранение, манипулирование, обработку запросов, поиск, выборку, сортировку, обновление, сохранение целостности и защиту от потери или несанкционированного доступа.

Повышенные требования к СУБД в ГИС по сравнению с традиционной формой их использования связаны с различием в представлении позиционной и атрибутивной составляющей информации.

Функции СУБД:

- управление данными во внешней памяти;

- управление буферами оперативной памяти;

- операции над БД;

- обеспечение надежности хранения данных в БД;

- поддержка языка управления БД.

Функция управления данными во внешней памяти **обеспечивает организацию структуры внешней памяти для хранения данных и ускорения доступа к ним. Обычно в СУБД создаётся собственная система именования объектов БД**

Управление буферами оперативной памяти. **БД ГИС имеют объем, значительно превышающий объём оперативной памяти. Для того, чтобы СУБД не зависела от скорости работы устройств внешней памяти организуется система буферов оперативной памяти с определёнными правилами замены и обновления буферов.**

Последовательность операций над БД, рассматриваемых СУБД как единое целое, называется транзакцией. **При выполнении транзакции СУБД либо фиксирует произведённые ею изменения в БД во внешней памяти, либо не производит никаких изменений. Каждая транзакция начинается при целостном состоянии БД и оставляет это состояние целостным после своего завершения.**

***СУБД должна выполнять функцию обеспечения надежности хранения данных в БД*, т.е. обеспечивать восстановление последнего согласованного состояния БД после любого программного или аппаратного сбоя.**

Для восстановления БД нужна некоторая избыточная информация. Для поддержания такой информации ведётся журнал изменений БД. Журнал – недоступная для пользователя часть БД, в которую поступают записи обо всех изменениях в базе данных.

Т.к. для работы с базами данных используются специальные языки, то СУБД должна обеспечить поддержку этих языков. В настоящее время наиболее распространены реляционные СУБД, стандартным языком которых является язык SQL (Structured Query Language).

Этот язык позволяет манипулировать данными реляционной БД. Компилятор языка SQL преобразует имена объектов в идентификаторы на основании специально поддерживаемых служебных таблиц-каталогов.

Специальные операторы этого языка позволяют определять запросы к столбцам БД. Язык SQL содержит все средства, необходимые для работы с БД, и обеспечивает базовый пользовательский интерфейс с базами данных.

Структура СУБД содержит три компоненты: командный язык для выполнения операций с данными (ввод, вывод, манипуляции), интерпретирующую систему (или компилятор) для обработки команд и перевода их на язык машины, интерфейс пользователя для формирования запросов к БД.

З А К Л Ю Ч Е Н И Е

На занятии вы

- познакомились с понятием базы данных ГИС и предъявляемыми к ним требованиями;

- узнали об уровнях процесса проектирования баз данных;

- познакомились с наиболее распространёнными логическими структурами (моделями) баз данных;

- получили представление о функциях и структуре систем управления базами данных.