

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ПГУ)

---

---

А.С. БОЖДАЙ, Ю.И. ЕВСЕЕВА, А.А. ГУДКОВ

ОСНОВЫ УПРАВЛЕНИЯ ГЕОИНФОРМАЦИОННЫМИ ПРОЕКТАМИ

Учебно-методическое пособие

ПЕНЗА 2017

**Бождай, А.С., Евсеева Ю.И., Гудков А.А.**

Основы управления геоинформационными проектами. Учебно-методическое пособие / А.С.Бождай, Ю.И.Евсеева, А.А.Гудков – Пенза : Изд-во Пенз. гос. ун-та, 2017. – 49 с.: ил. – Библиогр.: с. 49

Учебно-методическое пособие рассчитано на читателей, начинающих изучение геоинформатики, геоинформационных технологий и систем. Подробно рассматриваются теоретические и практические вопросы связанные с созданием цифровой картографической основы методом векторизации в программе Easy Trace, управлением типовым геоинформационным проектом в среде ArcView GIS, программированием функций пространственного анализа с использованием языка скриптов Avenue.

Учебное пособие разработано на кафедре «Системы автоматизированного проектирования» и предназначено для студентов бакалавриата по направлениям 09.03.01 «Информатика и вычислительная техника», 02.03.03 «Математическое обеспечение и администрирование информационных систем», магистратуры по направлениям 09.04.01 «Информатика и вычислительная техника», 01.04.02 «Прикладная математика и информатика» и аспирантуры по специальности 05.13.10 «Управление в социальных и экономических системах (технические науки)»

## **Введение**

Информационные процессы имеют решающую важность в ходе любой осмысленной деятельности человека. Наибольшей ценностью обладает та информация, которая представлена в динамике своего развития и изменения. За последние десятилетия было предпринято множество попыток математической формализации и моделирования процессов сбора, хранения, представления и обработки информационного хаоса. Полученные модели были положены в основу научных, технологических и программно-аппаратных решений, но долгое время технологии существенно отставали от стремительно возрастающих объемов информации.

Важно понимать, что очень большой объем доступной информации без соответствующей автоматизированной оперативной обработки и представления может зачастую не помочь, а помешать решению проблемы. Приблизиться к цели представилось возможным лишь с появлением мощных вычислительных систем, средств телекоммуникации и соответствующих периферийных устройств. Скорость обработки информации и доступные объемы памяти стали соизмеримы с объемами и скоростями ее внутренней динамики. После этого, принципиальной задачей стал поиск такой оптимальной компьютерной модели, которая с приемлемой эффективностью решала бы конфликт между противоречивыми критериями скорости и качества, объема и наглядности обработки данных. Ярким примером такой новейшей технологии будущего является геоинформатика и геоинформационные системы (ГИС).

Основная идея, лежащая в основе геоинформационных технологий (ГИТ) заключается в том, что любой процесс происходит в пространстве и вполне логично рассматривать и анализировать его с этой точки зрения. Информация привязывается к реальным или условным пространственно-географическим координатам. Это крайне важно при изучении сложных явлений на стыке множества предметных областей за счет возможности

совместного анализа больших групп параметров в их взаимной пространственной связи [2].

Термин геоинформатика состоит из трех корневых основ, отражающих его суть: география, информатика и автоматика (в англ.: geoinformatics). Имеет смысл привести сразу несколько определений, подчеркивающих тот факт, что геоинформатика является не только теоретической научной дисциплиной, но и глубоко затрагивает вопросы технологического и производственного характера.

**Геоинформатика** – научная дисциплина, изучающая геосистемы (их структуры, связи, динамику, функционирование в пространстве – времени) посредством компьютерного моделирования. (Научно-познавательный аспект).

**Геоинформатика** – технология, сбора, хранения, преобразования, отображения и распространения пространственно-координированной информации, имеющая целью обеспечить решение задач инвентаризации, оптимизации, управления геосистемами. (Технологический аспект).

**Геоинформатика** – производство, имеющее целью изготовление аппаратных средств и программных продуктов, включая создание баз данных (БД), систем управления, ГИС разного целевого назначения и проблемной ориентации. (Производственный аспект) [1 - 3].

Таким образом, геоинформатика предстает в виде системы, охватывающей науку, технику и производство.

**Географическая информационная система (ГИС)** – автоматизированная программно-аппаратная система, осуществляющая сбор, хранение, отображение и распространение пространственно-координированной информации. ГИС предназначена для решения научных и прикладных задач инвентаризации, анализа, оценки, прогноза и управления окружающей средой и территориальной организацией общества [4].

Первая автоматизированная картографическая система была создана в Великобритании в 1964г., а первая геоинформационная – в Канаде

(университет Манитоба) в начале 70-ых гг. Толчком к этому послужили военные технологии, использующие ЭВМ, топографические карты и космоснимки для расчета траекторий летательных аппаратов, ракет, а также в разведке. Только к началу 90-ых гг. ГИТ стала всецело массовой и гражданской благодаря стремительному развитию вычислительной техники, телекоммуникаций и систем дистанционного зондирования.

Современные ГИС классифицируются по ряду признаков [4, 5]:

1. По проблемной ориентации и тематике:

- инженерные;
- имущественные;
- тематического картографирования;
- управления природными ресурсами;
- административные;
- обработки космических изображений.

2. По целям и функциональному составу:

- инструментальные;
- ГИС – браузеры;
- векторизаторы;
- обработчики данных дистанционного зондирования;
- справочно-картографические системы.

3. По территориальному охвату:

- общенациональные;
- региональные;
- глобальные.

4. По программной архитектуре:

- закрытые;
- специализированные (ограниченная возможность модификации);
- открытые.

Данное пособие включает в себя три раздела, посвященные описанию практических аспектов работы с геоинформационными системами. В первом разделе рассмотрена технология создания цифровых карт методом векторизации с использованием программного средства Easy Trace. Второй раздел посвящен вопросам оформления и использования готовых цифровых карт с помощью ГИС ArcView GIS. Третий раздел освещает вопросы создания прикладного программного обеспечения для ArcView GIS с использованием встроенного объектно-ориентированного языка скриптов Avenue. Каждый из разделов содержит теоретическую часть, практические указания и список контрольных вопросов для самопроверки.

# **1. Автоматизированное проектирование цифровой картографической основы методом векторизации**

**Цель работы:** изучение технологии векторизации растровой карты с помощью инструментария Easy Trace.

## **Теоретическая часть**

В наши дни огромное количество информации, связанной с различными сферами человеческой деятельности, имеет территориальную привязку. Информационные системы больших городов, обеспечивающие работу городских властей, медицинских служб, образовательных учреждений и других элементов социальной инфраструктуры, включают в свой состав цифровые карты. Наличие карт позволяет пользователю системы оперативно решать задачи, связанные с пространственным расположением интересующих его объектов. Например, при вызове скорой помощи необходимо как можно быстрее получить информацию о месте вызова, а при поиске школы для ребенка - о расположении школы и связанных маршрутах общественного транспорта.

Информационная система, использующая, помимо баз данных, цифровые карты, носит название геоинформационной системы (ГИС). Современные ГИС являются многослойными. Многослойность означает наличие нескольких слоев цифровых карт, связанных друг с другом. К каждому слою может быть подключено несколько таблиц баз данных, и наоборот, каждая таблица может быть подключена к нескольким слоям.

Для создания цифровой карты и ведения геоинформационного проекта необходимо организовать структуры для хранения и использования данных. В ГИС такие структуры имеют свою специфику и значительно отличаются от структур данных систем автоматизированного проектирования (САПР) и систем автоматизированного картографирования (АМ – системы, от англ. Automated Mapping). Эта специфика заключается в объединении

графической, пространственной, картографической и описательной информации на в единой модели данных.

С точки зрения конечного пользователя, в ГИС четко выделяется два вида информации - пространственная (цифровая карта) и атрибутивная (базы данных, содержащие описательные сведения для объектов, составляющих цифровую карту).

Рассмотрим несколько важных определений [1 – 5]. **Объектом ГИС** называется объект реальной геосистемы, описываемый одним или несколькими геометрическими векторными примитивами, а также набором атрибутов. К геометрическим векторным примитивам относятся точки, полилинии и полигоны. Физически такие примитивы описываются последовательностью пар координат своих узлов.

Атрибутами являются числовые или символьные характеристики объекта или векторного примитива, хранящиеся в специальной БД. В качестве атрибутов также могут быть использованы растровые изображения, аудио и видеоматериалы, а также поведенческие атрибуты в виде функций и программных модулей, выполняющихся при определенных условиях.

Совокупность примитивов и атрибутов образует **простой объект ГИС**. Совокупность простых объектов образует **составной объект ГИС**. Иерархическое представление объектов обладает рядом несомненных достоинств: наглядность, избежание дублирования информации, поддержка принципа наследования. Таким образом, ГИС является ярко-выраженной объектно-ориентированной системой. По сути, каждый объект ГИС инкапсулирует в себе все четыре информационных составляющих, указанных выше:

- графическая составляющая: тип геометрического векторного примитива с координатами узловых точек;
- пространственная составляющая: привязка условных графических координат векторных примитивов к реальным пространственно-географическим координатам с учетом геодезических проекций и правил;



- картографическая составляющая: условные графические обозначения, при помощи которых каждый векторный примитив будет визуализироваться на цифровой карте;
- описательная (семантическая) составляющая: текстовые, числовые, мультимедийные и поведенческие характеристики каждого объекта, хранящиеся в атрибутивной базе данных.

Каждому объекту ГИС присваивается уникальный идентификатор (ID), при помощи которого осуществляется связь между векторными примитивами и соответствующими записями в атрибутивных базах данных.

**Слоем** называется совокупность объектов, сгруппированных по какому-либо тематическому, типовому или логическому признаку. Каждому слою также могут быть приписаны описательные атрибуты, а также цвет, флаг видимости, условные графические обозначения при визуализации. Любой слой может быть включен или выключен (в зависимости от состояния флага визуализации) из образа общей цифровой карты. Таким образом происходит удобное регулирование информационной насыщенности карты и абстрагирование в зависимости от рассматриваемой проблемы. Описание всех слоев и их атрибутов цифровой карты называется **легендой**.

При проектировании системы слоев, разработчик цифровой карты сталкивается с проблемой **классификации**. Под классификацией понимается система распределений объектов, явлений, процессов и понятий по классам в соответствии с определенными признаками. Классификация применяется в связи с необходимостью устранения второстепенных деталей. Определение общих свойств объектов позволяет найти общие ключевые абстракции и механизмы, что в свою очередь приводит к значительному упрощению решаемых задач. Задача классификации чрезвычайно сложна и трудноформализуема, поскольку, во-первых, нет определения «совершенная» классификация, во-вторых, каждая новая точка зрения на проблему порождает массу новых классификаций. Очевидно лишь то, что процедуру классификации должен определить последовательный итеративный подход с

привлечением экспертов в соответствующих предметных областях. Каждая новая итерация может выявить какие-либо новые сведения о решаемой проблеме, а следовательно, может внести новые, более важные критерии классификации.

Традиционно, при проектировании цифровых карт, используются модели данных на основе классификации объектов по географическо-тематическому признаку. Ярким примером такой модели является **расширенная геореяционная модель данных**, разработанная компанией ESRI (США) [6]. В ее основе лежит понятие **тематического класса** или **темы** - самодостаточного логического набора данных, определенного пользователем. Тематический объект инкапсулирует информацию в форме свойств и методов о том, где располагаются данные и как они должны отображаться.

**Цифровая картографическая основа (ЦКО, цифровая карта)** - цифровая модель, сформированная в принятой для карт проекции, системе координат, условных знаков и способов изображения в соответствии с правилами картографической генерализации. Пространственная обработка данных в практически всех ГИС происходит на основе ЦКО.

Все ЦКО делятся на векторные и растровые. Векторные ЦКО состоят из слоев на основе объектов, описываемых цифровыми геометрическими примитивами. Растровые ГИС представляют собой модель участка территории на основе целостного изображения, состоящего из множества цветных точек (пикселей), каждая из которых помимо цвета, может нести и другую информационную нагрузку.

Ряд специфических черт выгодно отличает цифровую карту от бумажного аналога:

- цифровая точность представления координат;
- ЦКО обладают «плавающим» (произвольно задаваемым) масштабом;
- ЦКО имеет гибкую математическую базу и с легкостью меняет проекции представления и системы координат;

- автоматическая проверка топологической корректности, что позволяет избежать при дальнейшей работе с картой множества ошибок;
- ЦКО имеет «плавающую» информационную насыщенность за счет слоеной структуры, при этом информационная насыщенность карты может легко регулироваться пользователем.

При использовании ГИС как информационной системы, оперирующей данными об индивидуализированных объектах, особенно техногенного происхождения, выгоднее использовать векторную модель. Когда же речь идет о пространственных явлениях, описываемых непрерывными полями (атмосферные явления, разливы рек, лесные пожары, прогноз погоды и т.п.), используется растровая модель. На практике очень часто обе модели образуют гибридный вариант. Векторные данные могут преобразовываться в растровые (или ссылаться на них) для проведения моделирования или аналитических операций, а по завершению, конвертируются обратно. Растровое изображение зачастую применяют в качестве подложки для векторного в целях повышения наглядности представления информации. Тем не менее, в таких случаях растровая графика играет, как правило, вспомогательную роль.

Процесс создания цифровой векторной карты состоит из нескольких этапов, наиболее важными из которых являются:

1. Выбор источника цифровой карты. Это может быть бумажная карта, спутниковый снимок или данные дистанционного зондирования (съемка с атмосферных или космических летательных аппаратов в каком-либо диапазоне электромагнитного спектра).
2. Преобразование выбранного источника в растровую форму. Для бумажной карты это может быть процесс сканирования, а для данных дистанционного зондирования – обработка при помощи специального ПО для преобразования результатов съемки в видимую часть спектра.
3. Векторизация растрового изображения и создание атрибутивных баз данных для каждого слоя цифровой карты.

Таким образом, **векторизация** это процесс преобразования растрового изображения в векторную форму с учетом выбранной системы слоев и заданных структур атрибутивных баз данных. **Векторизатор** – это специализированное программное средство для выполнения процесса векторизации.

Векторизация имеет три основных разновидности:

- **Ручная векторизация по подложке.** Отсканированное изображение (подложка) выводится на экран и оцифровка осуществляется по этой подложке вручную при помощи «мыши». По сути, пользователю приходится вручную обвести курсором «мыши» каждый объект, отображенный на растровом изображении. Достоинства: высокая точность получаемого векторного материала, низкие требования к исходному качеству источника цифровой карты, низкие требования к уровню профессиональной подготовки пользователя. Недостатки: высокая трудоемкость и временные затраты.
- **Интерактивная векторизация по подложке.** По сути не отличается от предыдущего метода за исключением того, что часть рутинных операций производится автоматически. Например, при векторизации непрерывной линии достаточно задать начальную точку и небольшой набор вспомогательных параметров. Далее векторизатор сам отследит эту линию до тех пор, пока на его пути не встретятся неопределенные ситуации (разветвление или разрыв линии). Качество интерактивной векторизации прямо связано с качеством исходного материала и сложностью карты.
- **Автоматическая векторизация.** Предусматривает очень небольшое, по сравнению с первыми двумя способами, вмешательство пользователя в работу программы. Перед автоматической векторизацией отсканированное изображение подвергается специальной подготовке, связанной с изменением яркости/контраста, применением фильтров, устранением высокочастотных графических «шумов». После выполнения оцифровки пользователем производится окончательное редактирование на предмет

выявления ошибок программы, которая может неверно распознать объект (принять символ за группу точек, определить полигон как набор несвязанных линий и т.п.). Недостатком метода является относительно высокая трудоемкость проведения окончательного редактирования, высокие требования к исходному качеству материала, высокую стоимость лицензионных программ-векторизаторов, работающих в автоматическом режиме.

Таким образом, применение программ-векторизаторов позволяет автоматизировать решение следующих базовых задач векторизации цифровой карты:

1. Восстановления геометрических характеристик объектов. Как правило, объекты на цифровых картах подчиняются некоторым требованиям. Например, контуры зданий должны быть выровнены вдоль осевых линий улиц.
2. Привязка объектов к определенным слоям цифровой карты.
3. Обеспечение топологической связности объектов цифровой карты, то есть корректных общих границ, вершин, узлов и т.д.
4. Создание производных объектов, таких как границы административных районов, зоны населенных агломераций различной плотности, буферные зоны природных явлений (зоны затопления при разливах рек, температурные зоны и др.).
5. Устранение «шумов» и растровых погрешностей исходного источника цифровой карты.

В данной лабораторной работе для векторизации растровой карты используется программное решение Easy Trace. Несомненным достоинством программы является ее "самодостаточность": Easy Trace позволяет проводить процедуры сканирования, коррекции растра, цветоделения, векторизации и экспорта без использования каких-либо сторонних программных средств и расширений. Для просмотра результатов векторизации будем использовать

программный пакет ArcView - мощный и легкий в использовании инструмент для обеспечения доступа к географической информации.

### Порядок выполнения работы и практические указания

1. Запустите систему Easy Trace.
2. Выберите в главном меню пункт **Файл=>Новый проект**. Откроется окно с настройками создаваемого проекта (рис. 1.1). Дайте проекту имя "Town". Поскольку карта, с которой мы будем работать, не предназначена для какой-либо конкретной ГИС, в поле **Прототип** оставьте значение **Без прототипа**. Проект-прототип содержит данные, необходимые для корректной и удобной работы с картой в целевой ГИС: настройки инструментов трассировки, утилит векторизации, автоматической коррекции и контроля данных.
3. Нажмите кнопку **Далее**. Переходим на второй этап настройки параметров проекта (рис. 1.2). Выберите пункт **Создать проект на основе растрового файла**, а затем файл M1.PCX (находится на локальном диске в папке с программой Easy Trace). Выбранный файл содержит растровую карту, векторный эквивалент которой будет создан в процессе выполнения лабораторной работы.

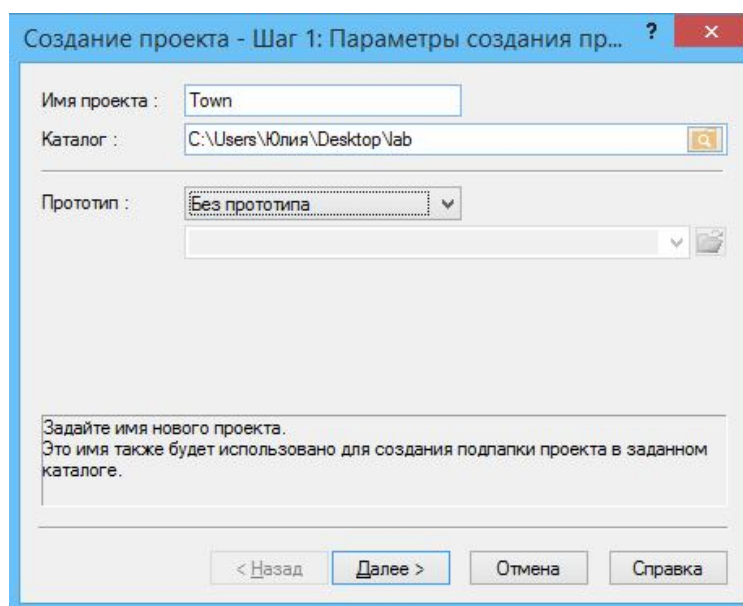


Рисунок 1.1. – Параметры создания проекта

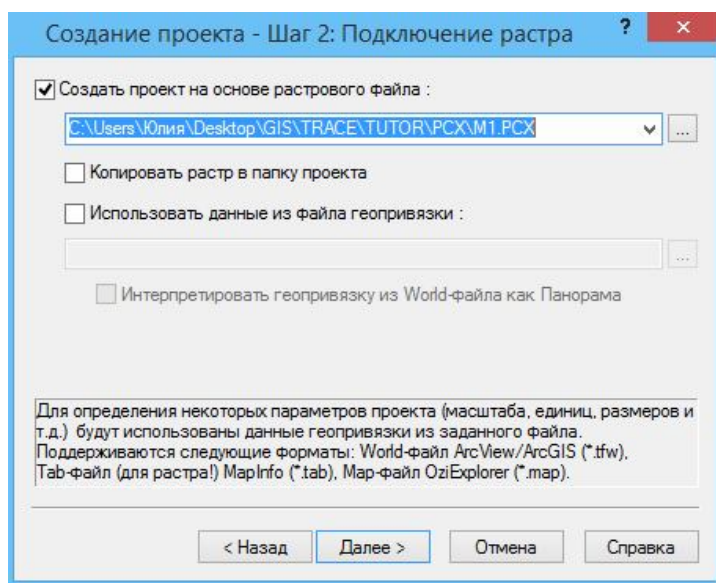


Рисунок 1.2 - Подключениерастровой подложки

4. Нажмите кнопку **Далее**. Переходим к третьему этапу настройки проекта (рис. 1.3). В открывшемся диалоговом окне можно изменить значения следующих параметров:

- 1) Направление осей координат.
- 2) Масштаб будущей карты. Чем детальнее масштаб, тем чаще EasyTrace будет ставить узлы при векторизации линейных объектов в автоматическом режиме.
- 3) «Единицы измерения». В этом окне устанавливаются единицы измерения в будущей карте.
- 4) «DPI виртуального растра». В этом окне устанавливается разрешение растрового изображения, которое будет векторизоваться.
- 5) В соответствующих полях указываются значения X и Y координат левого нижнего и правого верхнего углов векторного поля проекта, которые могут не совпадать с координатами углов растрового поля.

В нашем случае можно оставить настройки по умолчанию.

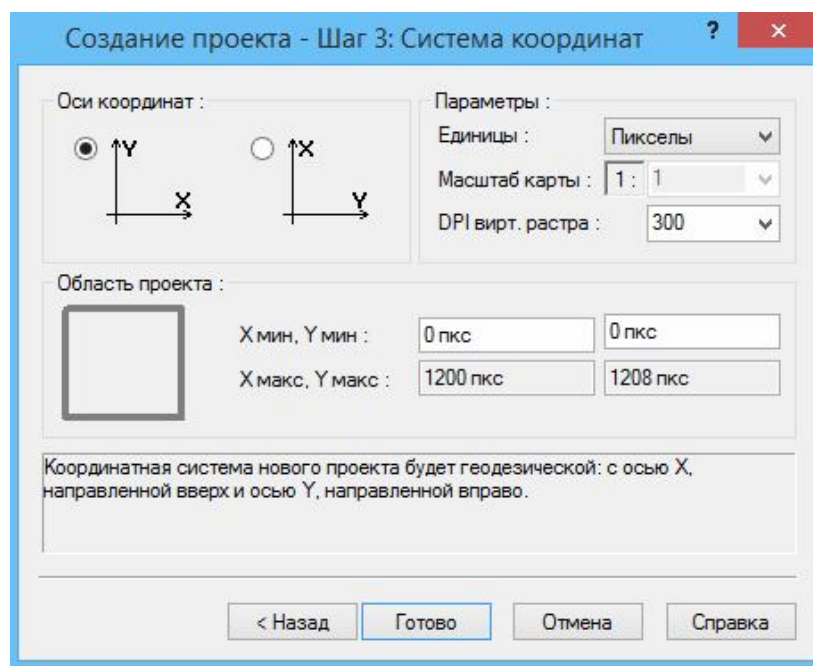


Рисунок 1.3 - Система координат проекта

5. Нажмите кнопку **Далее**. В основном окне Easy Trace будет отображена растровая карта из файла M1.PCX. Заметим, что загруженная карта имеет черно-белый формат. Это связано с тем, что изображение предварительно было подвергнуто процедуре бинаризации - выделению черно-белых растров из цветных. Хотя программа Easy Trace способна векторизовать как черно-белые растры, так и цветные, векторизация по черно-белой подложке выполняется значительно быстрее и удобнее. Easy Trace также включает в свой состав инструмент-бинаризатор, однако его изучение выходит за рамки данной лабораторной работы.

6. Для удобства будем работать не с целой картой, а с ее фрагментом. Для этого выберите в главном меню пункт **Растры=>Задать границы...** и выделите необходимый фрагмент карты (рис. 1.4). Щелкните правой кнопкой мыши по выделенному фрагменту и на появившейся панели нажмите кнопку **Задать границы**.



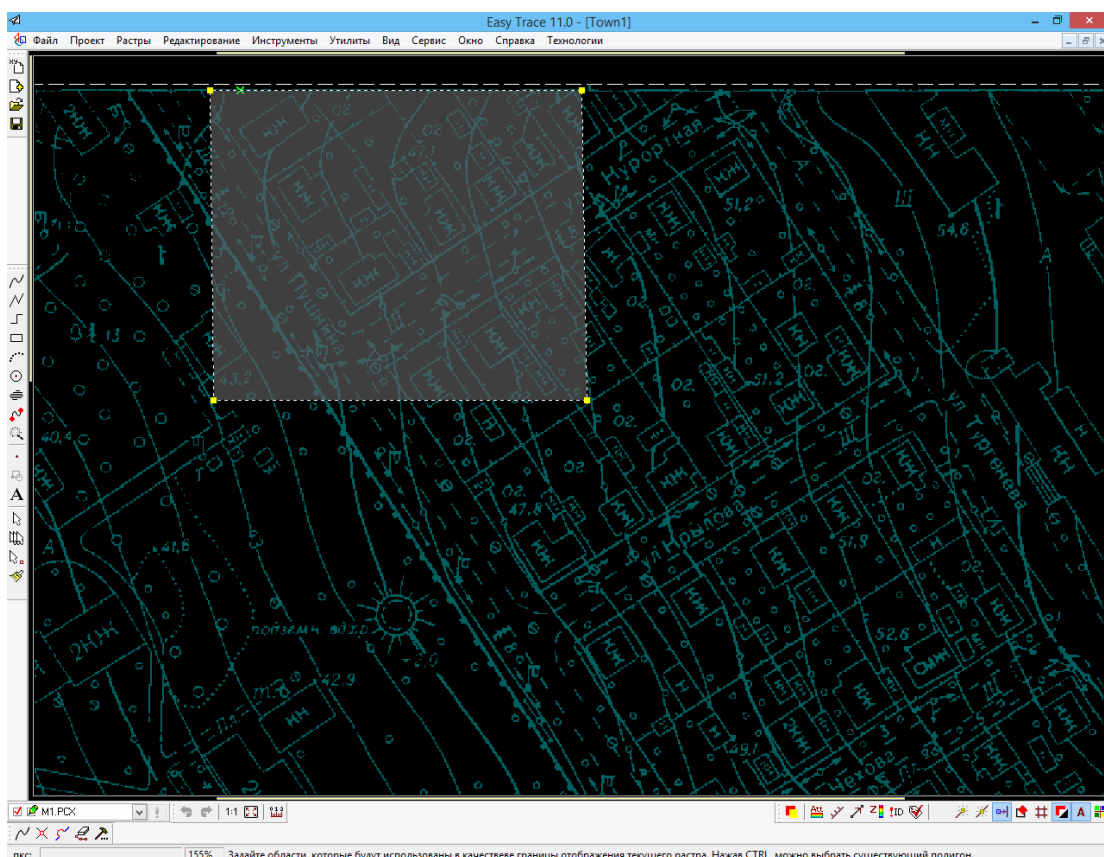


Рисунок 1.4 - Выделение фрагмента карты

7. Откройте редактор слоев проекта. Для этого выберите в главном меню пункт **Проект=>Слои проекта**. Откроется окно менеджера слоев (рис. 1.5). Изначально проект содержит один растровый слой (соответствует загруженному изображению карты) и один векторный (является пустым). Векторный слой предназначен для хранения "векторизованных" копий объектов растрового слоя (например, зданий). Однако реальные цифровые карты, как правило, не ограничиваются одним слоем. Количество слоев в них обычно равно количеству типов выделяемых на карте объектов. Каждый слой хранит объекты определенного типа. Как видно из рисунка 1.5, в данном проекте мы выделили 4 тематических векторных слоя: здания, дороги, опоры освещения, рельеф(изолинии). Объекты каждого тематического слоя выделены собственным цветом (например, здания - красным, а дороги - зеленым). Добавьте в созданный вами проект векторные тематические слои в соответствии с рис. 1.5. Для этого щелкните правой

кнопкой мыши по полю **Векторные слои** и во всплывающем меню выберите пункт **Создать слой**. Введите имя слоя и задайте цвет его объектов.

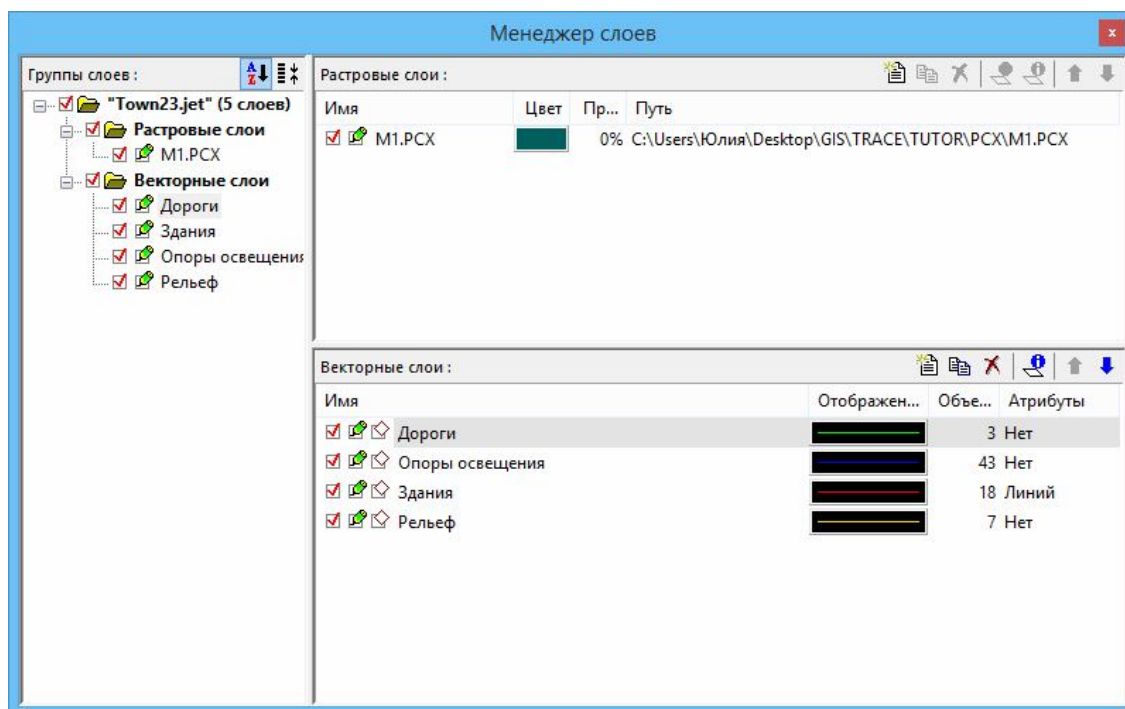


Рисунок 1.5 - Менеджер слоев проекта

8. Базы данных в ГИС предназначены для хранения связанной с географическими объектами информации (например, ширины и названий рек, имен владельцев домов и т.д.). Easy Trace позволяет установить связь с базой данных 2 типам объектов: точкам и полилиниям. Атрибутивные таблицы, являющиеся основным элементов баз данных в ГИС, определяются независимо для линий и точек каждого векторного слоя. Таким образом, к любому слою проекта может быть подключено 2 таблицы. Для того, чтобы создать атрибутивную таблицу для некоторого векторного слоя, необходимо в менеджере слоев щелкнуть левой кнопкой мыши по столбцу **Атрибуты** напротив имени интересующего слоя. Будет вызвано окно конструктора базы данных (рис. 1.6). Создайте для каждого векторного слоя собственную атрибутивную таблицу.

9. Заполните векторные слои объектами. Для этого в поле со списком, расположенном в левом нижнем углу основного окна Easy Trace, отметьте галочкой базовый растровый слой и интересующий векторный слой

(остальные слои должны остаться невыделенными). Затем с помощью одного из инструментов трассировки (панель инструментов трассировки расположена по левому краю основного окна Easy Trace) обведите на исходном растровом изображении интересующие объекты. На рис. 1.7 показан процесс оцифровки зданий с помощью трассировщика прямоугольников.

10. Повторите действия, описанные в пункте 9, для каждого векторного слоя. На рисунке 1.8 изображен результат выделения объектов всех 4-х слоев проекта.

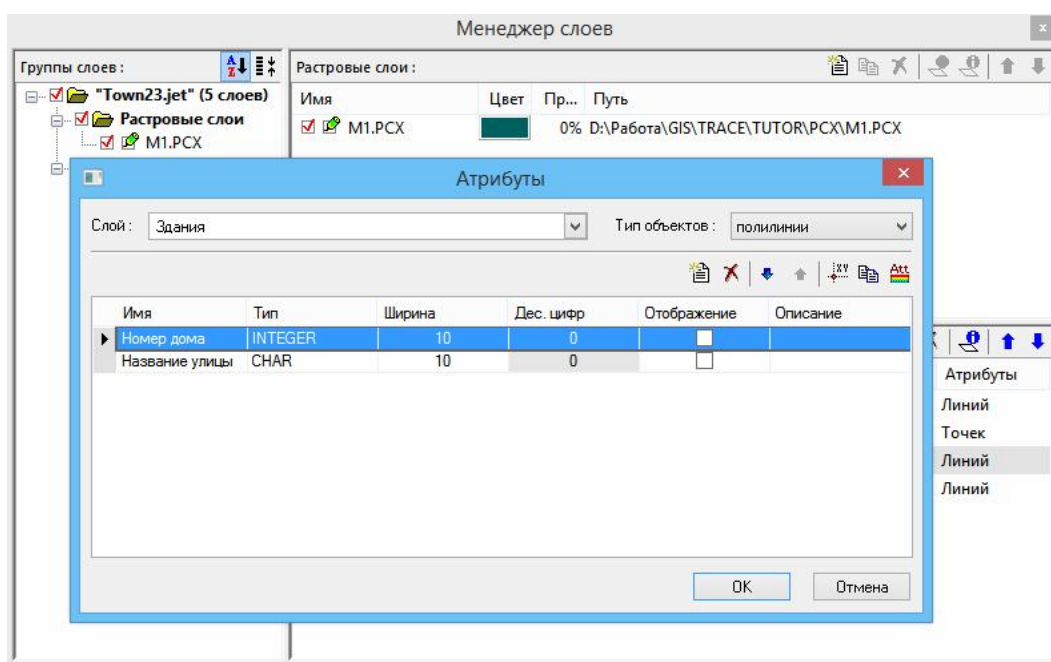


Рисунок 1.6 - Редактор атрибутивной базы данных

11. Для просмотра результатов оцифровки экспортируйте проект во внутренний формат ГИС ArcView. Для этого выберите в главном меню пункт **Файл=>Экспорт**. В поле **Формат** выберите пункт SHP (ArcView). Окно экспорта представлено на рис. 1.9 Нажмите кнопку **Далее** и в следующем окне выделите векторные слои для экспорта. Нажмите кнопку **Далее** еще несколько раз, оставляя предоставляемые настройки без изменений.

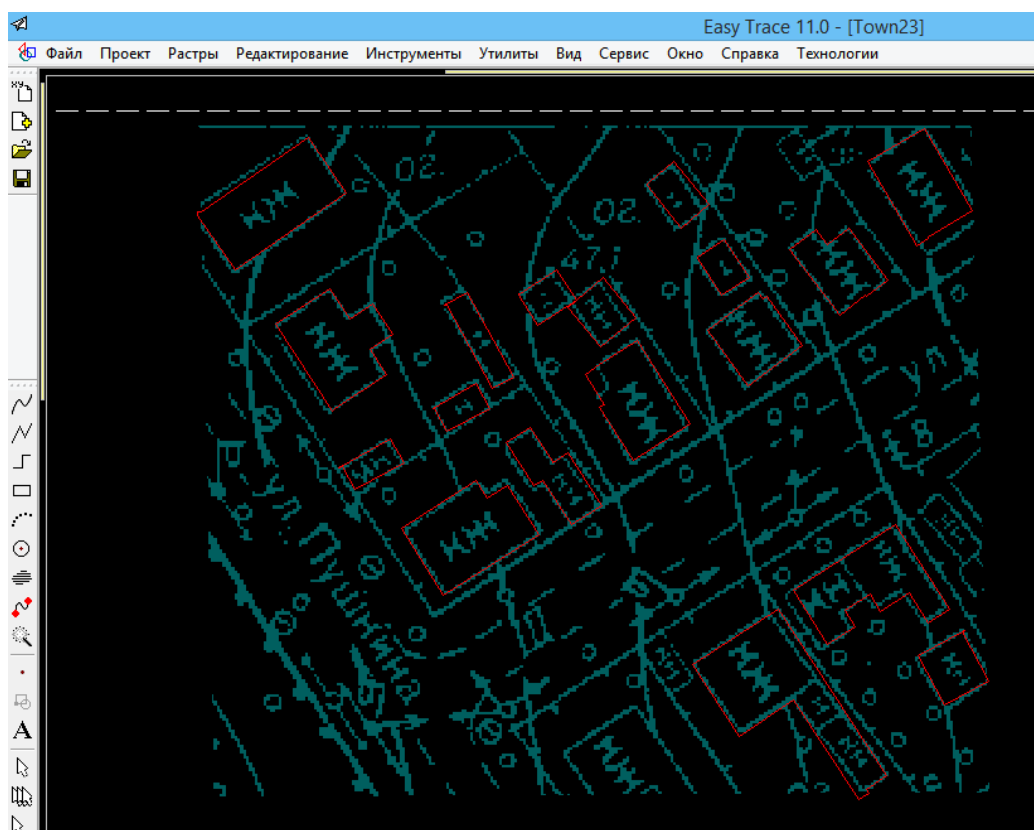


Рисунок 1.7 - Заполнение векторного слоя объектами

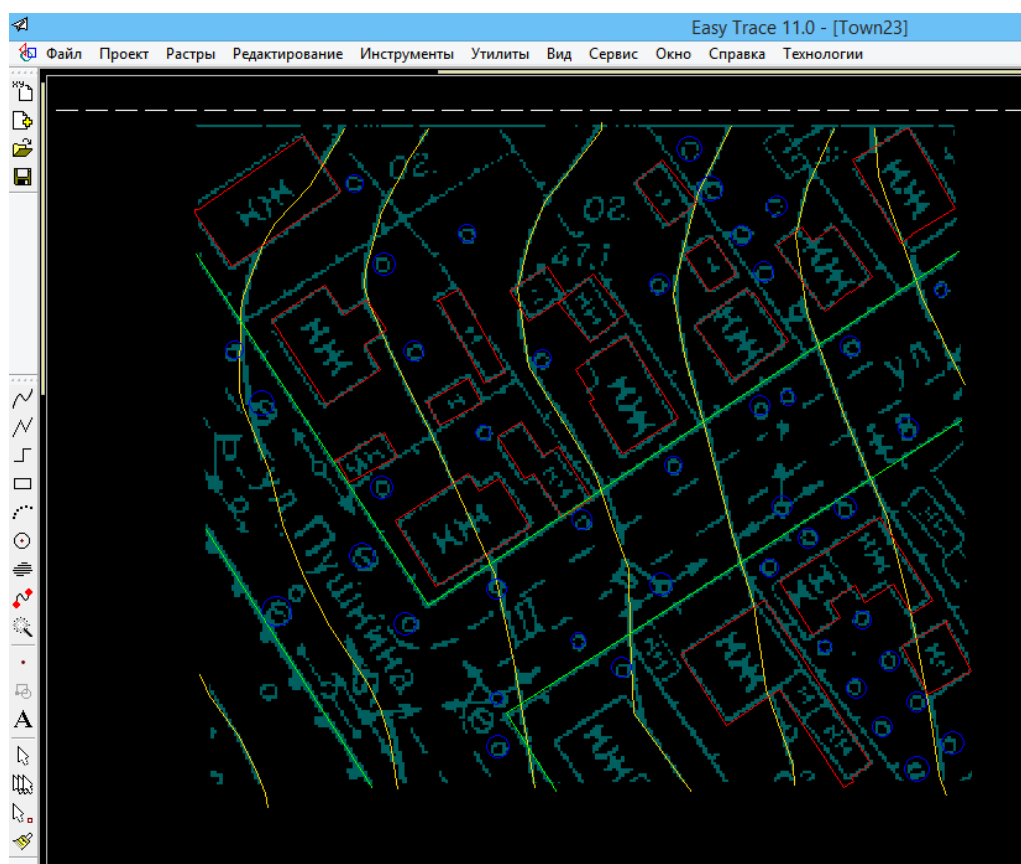


Рисунок 1.8 - Векторные слои

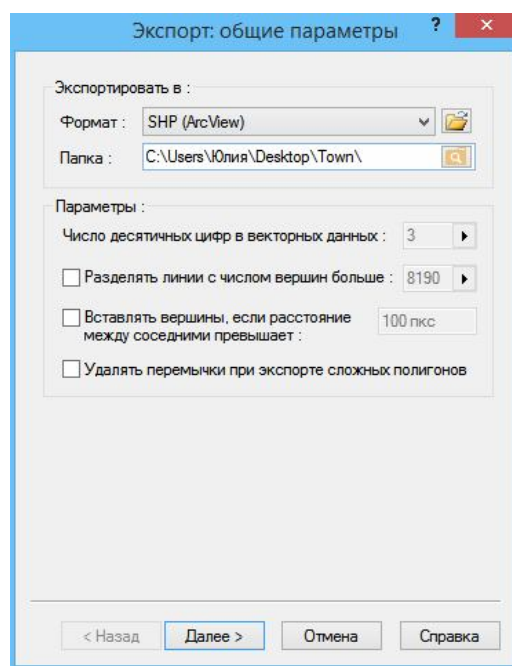


Рисунок 1.9 - Экспорт проекта в формат ArcView

12. Запустите ArcView, создайте пустой проект. В рабочей области основного окна программы отобразится окно компонентов проекта, содержащее 5 основных типов компонентов: **Views (Виды)**, **Tables (Таблицы)**, **Charts (Диаграммы)**, **Layouts (Компоновки)**, **Scripts (Скрипты)**. Щелкните левой кнопкой мыши по пункту **Views**. Автоматически будет создано новое представление. Нажмите на кнопку **Add theme (Добавить тему)** на появившейся в верхней части окна панели инструментов. В открывшемся диалоговом окне выберите файлы формата shp, которые были получены в процессе экспорта проекта карты из Easy Trase. Файлы будут загружены в проект в качестве векторных слоев новой карты. Чтобы отобразить слои, в окне представления поставьте галочки напротив их имен. Результаты импорта представлены на рис. 1.10.

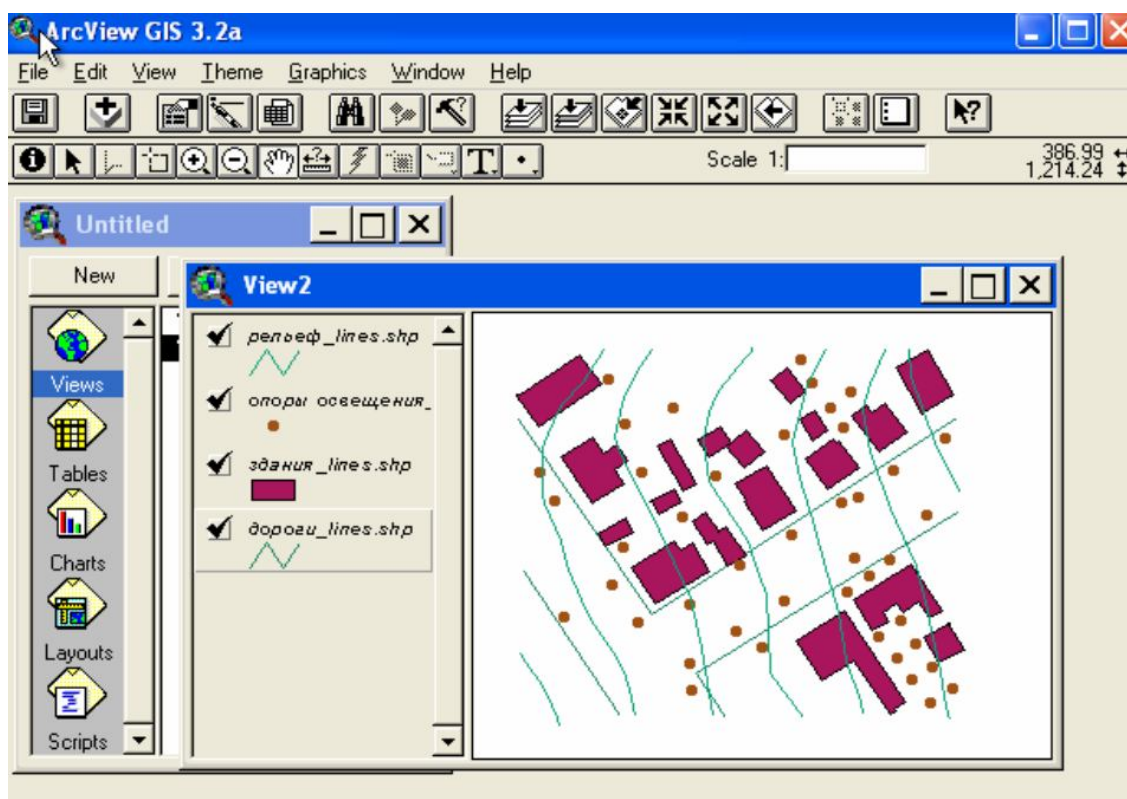


Рисунок 1.10 - Импорт векторных слоев в ArcView

13. Добавьте таблицы атрибутивной базы данных. Выберите пункт **Tables**, нажмите кнопку **Add**. В открывшемся диалоговом окне выберите файлы формата dbf, которые были получены в процессе экспорта проекта карты из Easy Trace. Результаты импорта представлены на рис. 1.11.

### Задание для самостоятельной работы

1. Получить у преподавателя фрагмент карты для выполнения лабораторной работы.
2. Провести оцифровку и экспорт фрагмента карты в ArcView в соответствии с рассмотренным примером.
3. Оформить отчет.



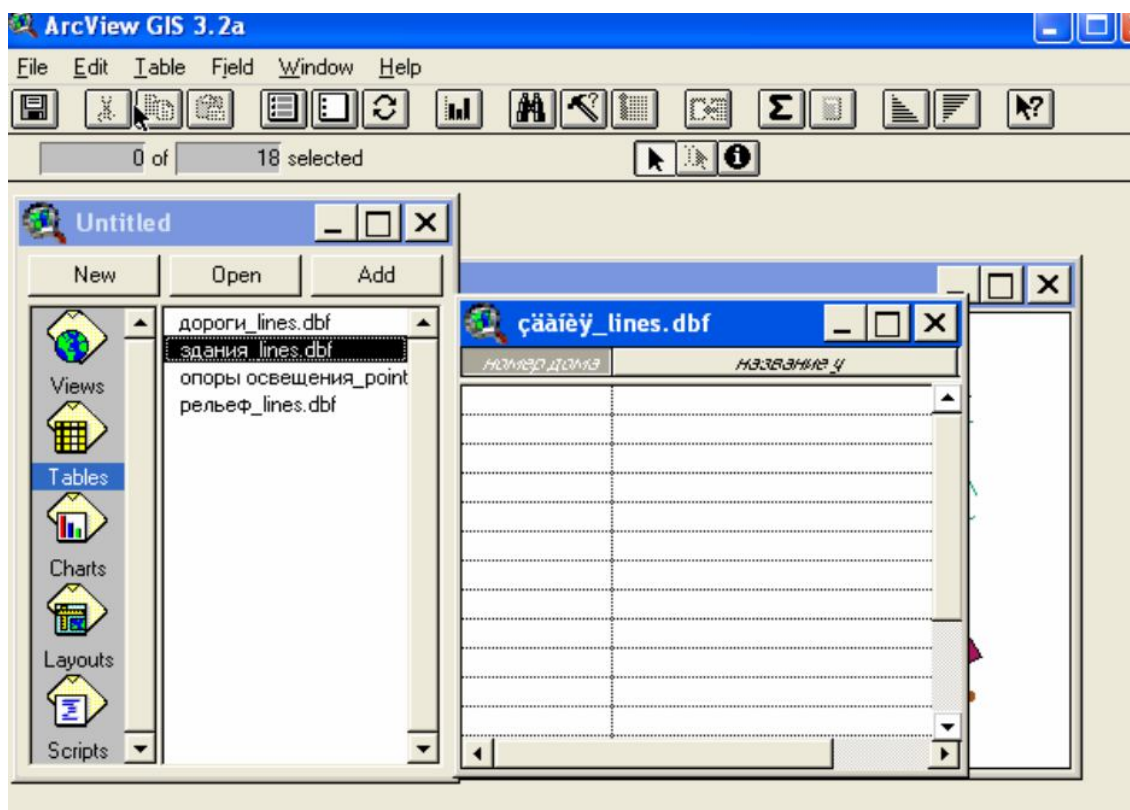


Рисунок 1.11 - Импорт таблиц атрибутивной базы данных в ArcView

### Контрольные вопросы

1. Что такое геоинформатика и геоинформационная система?
2. Что такое объект ГИС?
3. В чем отличие простых и составных объектов ГИС?
4. Из чего состоит объект ГИС?
5. Что понимается под слоем цифровой карты?
6. По какому принципу происходит распределение объектов ГИС по слоям?
- В чем заключается проблема классификации?
7. Что такое цифровая картографическая основа?
8. Что понимается под термином «легенда» цифровой карты?
9. В чем суть расширенной геореляционной модели данных?
10. Назовите основные типы цифровых карт.
11. Перечислите основные преимущества использования цифровых карт.
12. Для каких целей и задач удобнее применять векторные модели ЦКО?
13. Назовите основные этапы создания цифровой карты.

14. Что такое векторизация и векторизатор? Охарактеризуйте основные способы векторизации.
15. Назовите основные этапы векторизации цифровой карты по векторной подложке.
16. В чем назначение и основные функции программы Easy Trace?
17. Для каких целей необходимо создание атрибутивной БД?
18. Каким образом осуществляется связь между объектами ГИС и записями атрибутивной БД?



## 2. Ведение геоинформационных проектов в среде ArcView GIS

**Цель работы:** изучение основ работы с геоинформационной системой ArcView и оформление фрагмента цифровой картографической основы по правилам цифровой картографии.

### Теоретическая часть

Геоинформационная система (ГИС) ArcView разработана в Институте исследования систем окружающей среды (ESRI, США) и предназначена для сбора, хранения, обработки и визуализации пространственно привязанной информации.

ГИС ArcView имеет модульное строение (рис. 2.1) - то есть имеет ядро, к которому могут подключаться различные функциональные модули.

Ядро универсальной ГИС ArcView содержит следующие компоненты (рис. 2.2):

- системы ввода и вывода, включая систему подготовки макетов карт к печати;
- система управления графическими и пространственными данными;
- система управления атрибутивными базами данных;
- система визуализации;
- система обработки и анализа, включая модуль построения диаграмм и графиков;
- встроенный скриптовый язык программирования Avenue.

Система управления пространственными данными позволяет создавать, редактировать, визуализировать и представлять в различных картографических проекциях пространственные данные, выполнять пространственные запросы. ArcView может работать с несколькими наборами географических данных в интерактивных картах, называемых «Видами».

Система управления базами данных позволяет создавать и редактировать атрибутивные данные, устанавливать реляционные отношения между различными таблицами, связываться с внешними базами данных, а также выполнять различные запросы по атрибутивным данным.

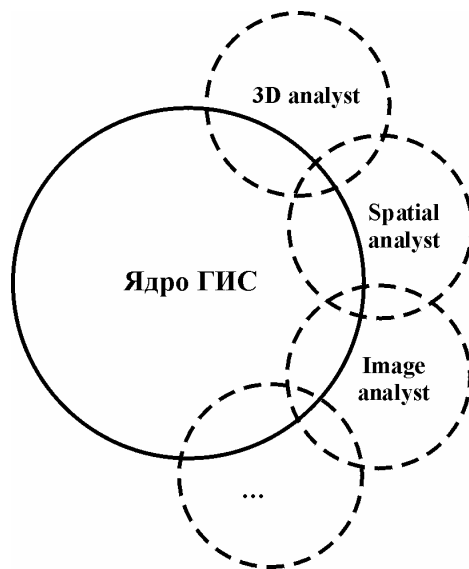


Рисунок 2.1 - Модульное строение универсальной ГИС ArcView

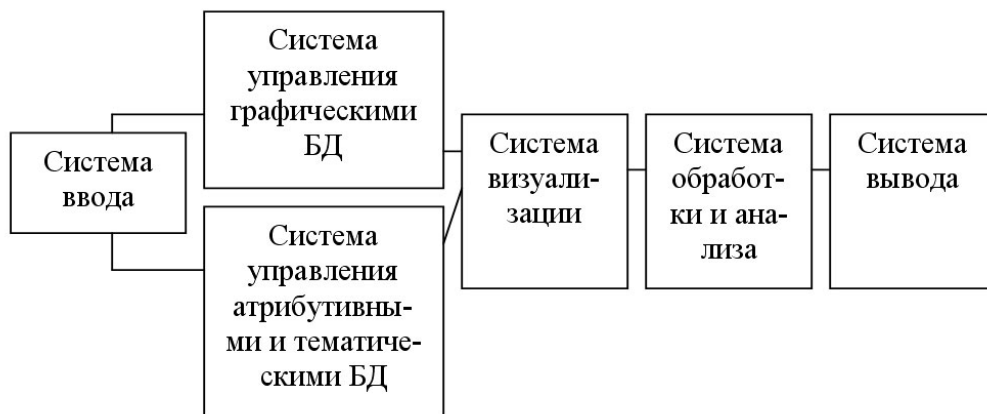


Рисунок 2.2 – Состав ядра типовой ГИС

Система построения диаграмм и графиков позволяет проводить построение различных диаграмм и графиков по атрибутивным данным.

Система подготовки макетов карт к печати позволяет проводить сборку картографической, атрибутивной и графической информации на один лист и выводить полученный макет на печать.

Встроенный объектно-ориентированный язык программирования Avenue позволяет пользователю создать свой собственный интерфейс, программно представлять имеющиеся наборы данных, разрабатывать собственные алгоритмы обработки пространственной и атрибутивной информации.

Основными компонентами проекта ArcView являются [6]:

**Модуль «Виды» (Views)** - это подсистема просмотра и интерактивной обработки цифровой картографической основы, которая позволяет отображать, исследовать, делать запросы и анализировать пространственные данные в ArcView. Модуль «Виды» определяет набор пространственных данных, которые будут использоваться, и способ их отображения, однако он не содержит файлы пространственных данных в явном виде. Вместо этого, модуль ссылается на заранее сформированные исходные файлы данных. Процесс формирования таких файлов детально рассмотрен в разделе 1.

Таким образом, модуль «Вид» работает с исходными файлами динамическим образом, отражая текущее состояние содержащихся в них данных. Если исходные данные изменяются, модуль автоматически отразит изменение при очередной визуализации цифровой карты. Эти же самые данные одновременно могут использоваться в нескольких цифровых картах, открытых в модуле. Модель цифровых карт, используемая в ArcView, соответствует расширенной геореляционной модели данных, упоминаемой в разделе 1.

**Модуль «Таблицы» (Tables)** выполняет роль подсистемы управления атрибутивными базами данных в ArcView и позволяет работать с реляционными (табличными) базами данных, описывающих семантику объектов цифровой карты. Информацию из таблиц можно добавлять на карты, символизировать при помощи условных графических обозначений, делать запросы и анализировать эти данные с точки зрения их положения в пространстве (то есть географически). Таблица в ArcView включает ссылку на представляемый ею источник табличных данных, но не содержит сами

табличные данные. Это означает, что таблицы являются динамическими, то есть, если исходные данные изменяются, таблица, основанная на этих данных, автоматически отразит изменения при следующем открытии проекта, содержащего эту таблицу. Кроме того, ArcView предоставляет возможности для создания новых таблиц и их полнофункционального редактирования.

**Модуль «Диаграммы» (Charts)** – содержит функции визуального представления данных из атрибутивных БД, в особенности атрибутов географических (распределенных в пространстве) объектов, удобный для представления информации в оптимальном для восприятия и осмысления виде. Диаграммы в ArcView особенно удобны, так как они динамически связаны с картами. Например, если выбрать какие-либо объекты цифровой карты в модуле «Вид», то автоматически будут выделены соответствующие записи в базе данных модуля «Таблицы» и элементы диаграмм и графиков в модулей «Диаграммы».

**Модуль «Компоновка» (Layouts)** – отвечает за предварительную подготовку и макетирование цифровой карты перед выводом ее на печать. Модуль «Компоновка» определяет какие данные будут использоваться для вывода и как они будут отображаться. На одном листе с картографическим изображением могут быть размещены легенда карты, фрагменты таблиц, графики, диаграммы, пояснительные надписи. При компоновке все данные также имеют динамическую связь с модулями «Виды», «Таблицы», «Диаграммы».

**Модуль «Скрипты» (Scripts)** - являются компонентом ArcView, содержащим компилятор и редактор скриптов на языке Avenue. Также как макросы, процедуры или скрипты в других скриптовых языках программирования, скрипты Avenue объединяют средства для выполнения трех общих целей: задачи автоматизации функций геоинформационного проекта, добавления новых возможностей в ArcView, создания и внедрение в проект законченных прикладных программ (приложений).

## Порядок выполнения работы

1. Запустите программу ArcView.
2. Создайте новый проект. Для этого выберите пункт меню **File=>New project(Файл=>Новый проект)**. Откроется окно пустого проекта, содержащее 5 основных типов компонентов: **Views (Виды)**, **Tables (Таблицы)**, **Charts (Диаграммы)**, **Layouts (Компоновки)**, **Scripts (Скрипты)** (рис.2.3).

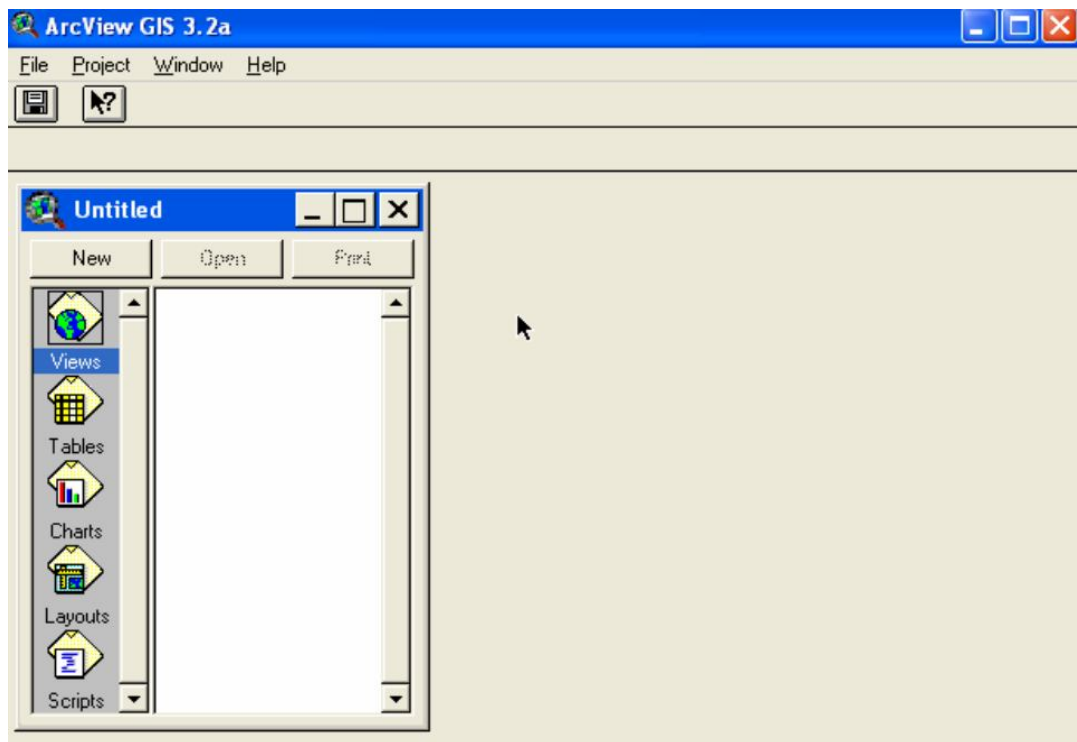


Рисунок 2.3 - Пустой проект ArcView

3. Для дальнейшей работы необходимо подключить к созданному проекту тему или ранее созданный проект. Чтобы подключить готовый проект, необходимо: 1) выбрать вкладку **Views (Виды)** 2) Выбрать пункт меню **Project=>Import(Проект=>Импорт...)** 3) В открывшемся диалоговом окне выбрать проект **world.apr**. На рис. 2.4 показано диалоговое окно импорта проекта, на рис. 2.5 – внешний вид проекта в модуле «Виды».
4. Загруженный проект содержит карту мира. Для дальнейшей демонстрации функций ArcView выберем отдельную страну – например, Испанию.

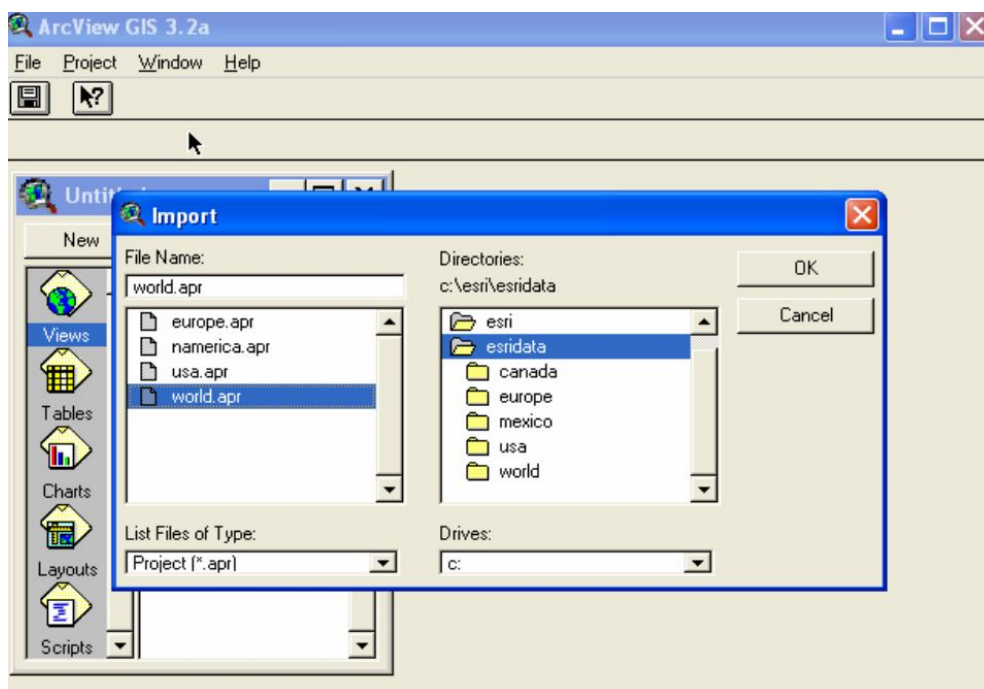


Рисунок 2.4 - Диалоговое окно импорта проекта

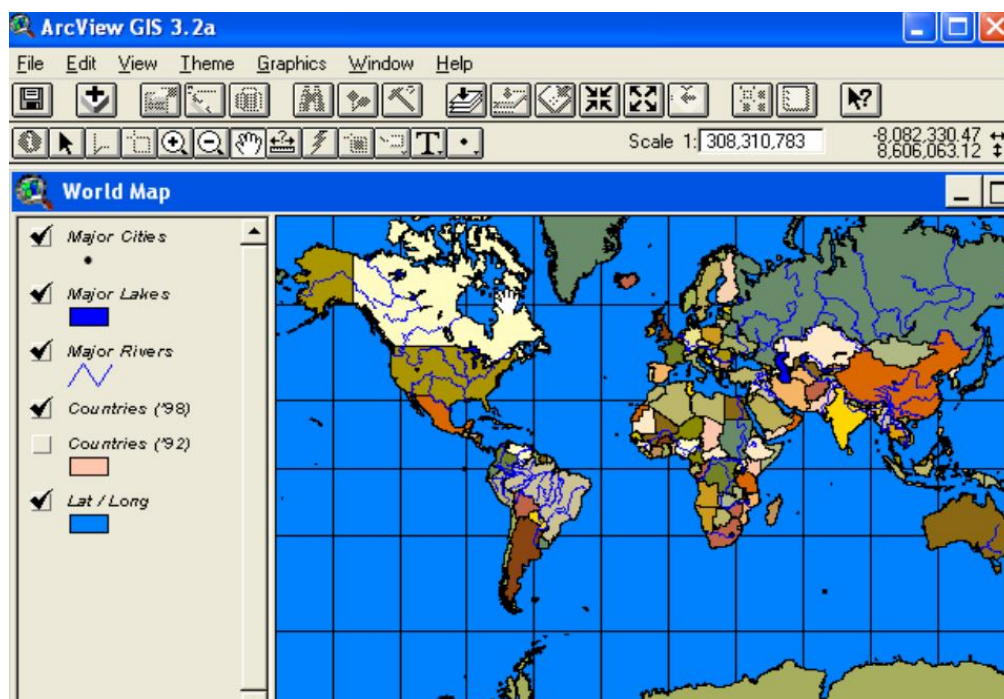


Рисунок 2.5 - Внешний вид проекта в модуле «Виды»

5. Можно заметить, что на загруженной карте отсутствуют названия городов. Данный недочет можно исправить с помощью инструмента **Auto-Label** (Автоподпись). Чтобы применить его, выделим в перечне слоев (панель слева от карты) слой **Major Cities**. Затем выберем пункт меню

**Theme=>Auto-Label (Тема=>Автоподпись...).** Отобразится окно настройки автоподписи (рис.2.6). В поле со списком **Label field (Поле подписи)** оставим значение **Name (Название города)** и нажмем **ОК**. На карте появятся подписи основных (наиболее крупных) городов.

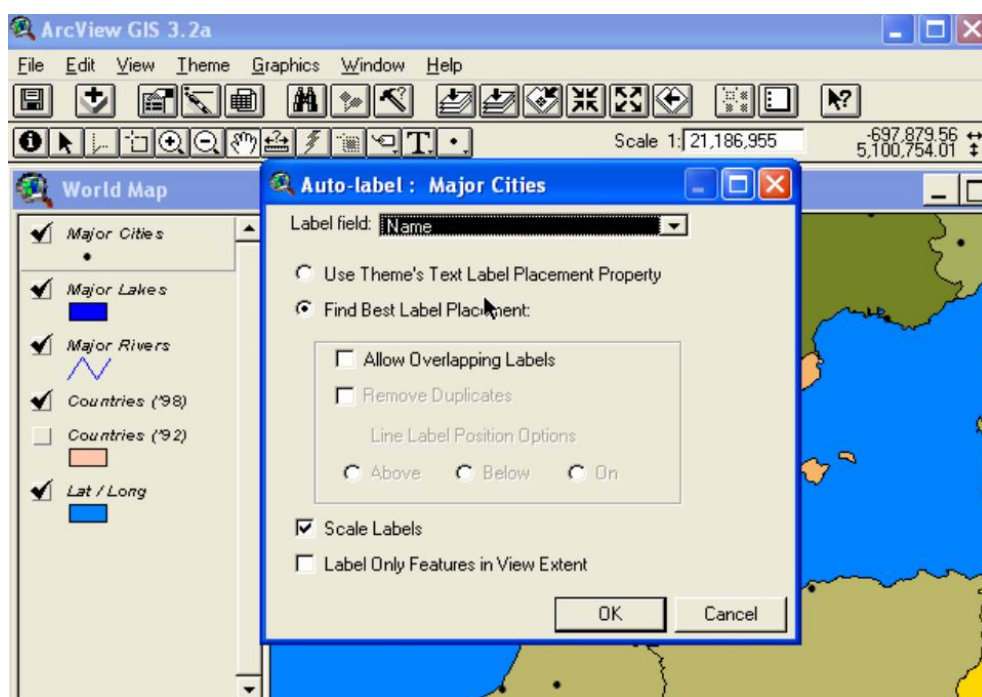


Рисунок 2.6 - Настройка инструмента автоподписи

6. Инструмент **Автоподпись** добавил названия городов на английском языке. Чтобы заменить названия на русскоязычные, выберите инструмент **Text(Текст)** на верхней панели. Выделите с помощью инструмента название города, откроется окно редактирования подписи (рис. 2.7). Исправьте название. Повторите действие для всех городов страны.

7. Легенда представляет собой поясняющий текст (либо условные графические обозначения) на географической карте. В ArcView можно автоматически создавать легенды, отображающие параметры ключевых объектов. В качестве примера создадим легенду, ставящую в соответствие каждому городу графический символ. Размер символа будет определяться численностью населения города (чем больше население - тем крупнее символ).

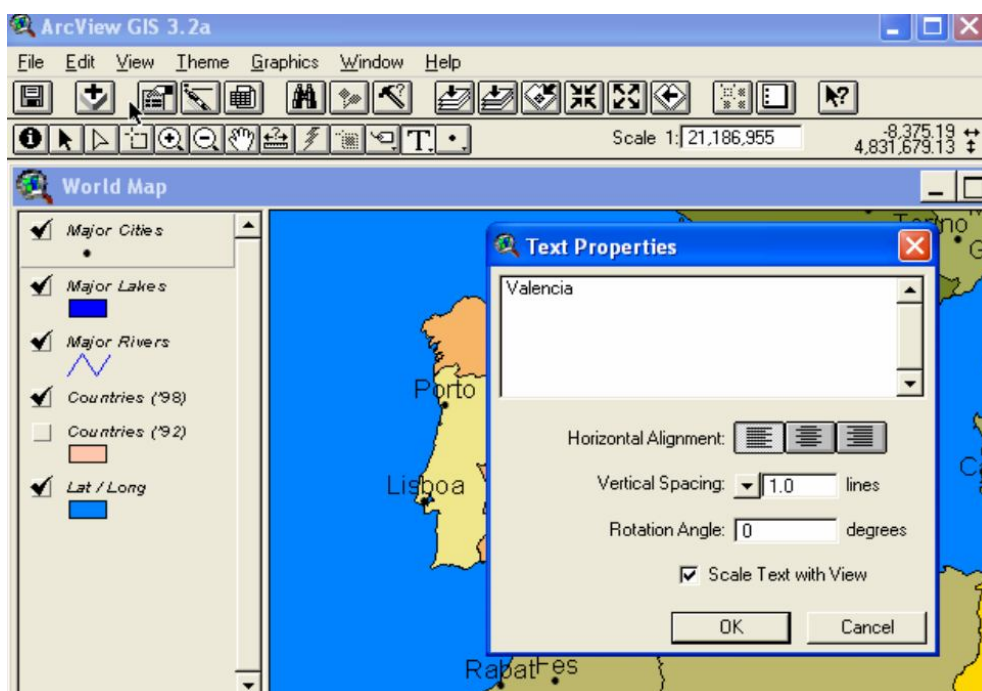


Рисунок 2.7 – Инструмент редактирования подписи

Выделим в окне вида слой **Major Cities**, затем выберем инструмент **Edit Legend (Редактировать легенду)** на верхней панели (также можно открыть этот инструмент двойным щелчком мыши по слою). Откроется окно редактирования легенды (рис. 2.8). В поле **Legend Type (Тип легенды)** выберем **Graduated symbol (Масштабируемый символ)**. Отобразится таблица из 3-х столбцов: **Symbol (Символ)**, **Value (Значение)**, **Label (Подпись)**. Столбец **Value** содержит диапазоны значений параметра из поля со списком **Classification field (Классифицировать по полю)**, которые соответствуют значениям масштабируемого символа. В данном случае в **Classification field** можно указать только одно значение – **Population (Популяция, численность населения)**. Рекомендуется также изменить численное значение диапазонов в столбце **Value**, чтобы результаты были более наглядны. После завершения настройки легенды нажмите кнопку **Apply (Применить)**. Отредактированная легенда будет отображена слева от карты (рис. 2.9).



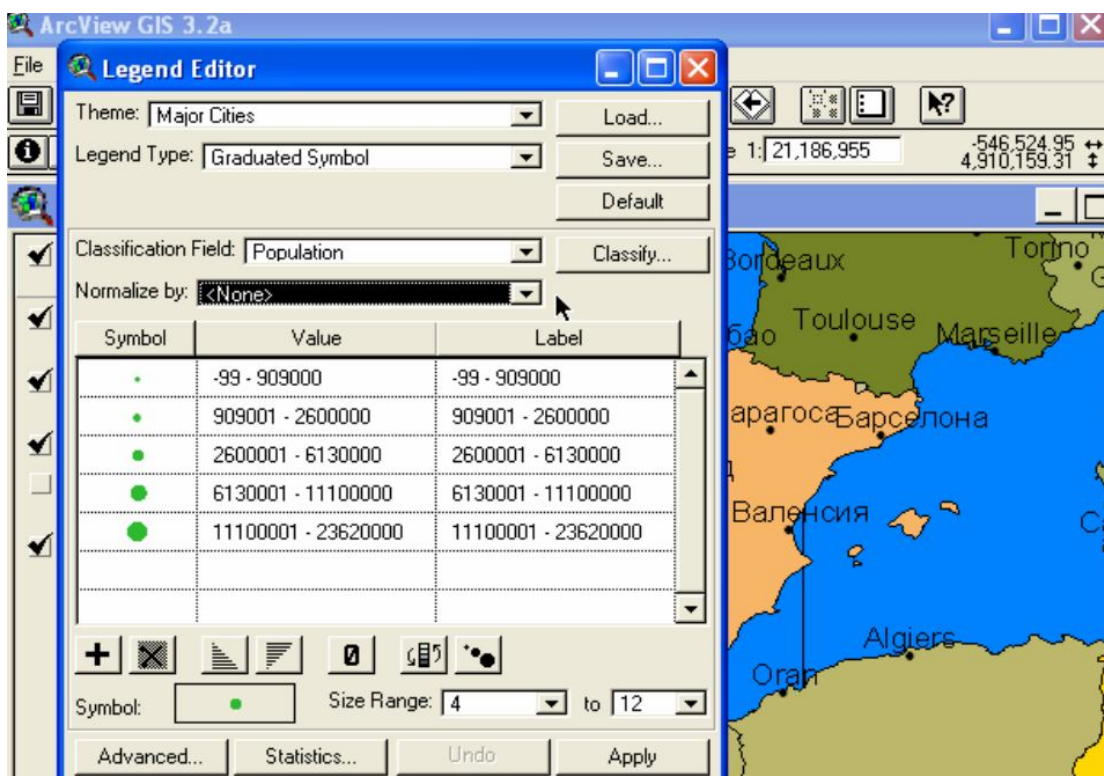


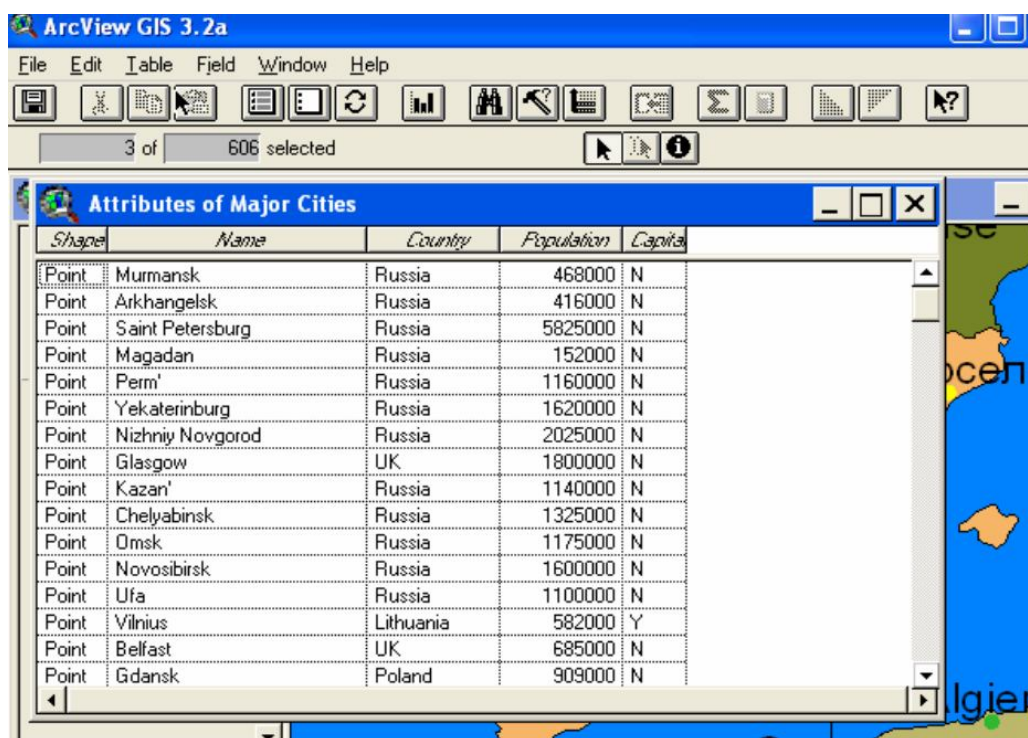
Рисунок 2.8 – Инструмент редактирования легенды



Рисунок 2.9 - Отображение легенды цифровой карты

8. Важнейшим компонентом любой ГИС являются атрибутивные базы данных. Для того, чтобы в ArcView просмотреть атрибутивную таблицу слоя,

нужно выбрать кнопкой мыши интересующий слой и нажать кнопку **Open Theme Table (Открыть таблицу темы)** на верхней панели. Отобразится база данных слоя (рис. 2.10).



Shape	Name	Country	Population	Capital
Point	Murmansk	Russia	468000	N
Point	Arkhangelsk	Russia	416000	N
Point	Saint Petersburg	Russia	5825000	N
Point	Magadan	Russia	152000	N
Point	Perm'	Russia	1160000	N
Point	Yekaterinburg	Russia	1620000	N
Point	Nizhniy Novgorod	Russia	2025000	N
Point	Glasgow	UK	1800000	N
Point	Kazan'	Russia	1140000	N
Point	Chelyabinsk	Russia	1325000	N
Point	Omsk	Russia	1175000	N
Point	Novosibirsk	Russia	1600000	N
Point	Ufa	Russia	1100000	N
Point	Vilnius	Lithuania	582000	Y
Point	Belfast	UK	685000	N
Point	Gdansk	Poland	909000	N

Рисунок 2.10 - Атрибутивная таблица слоя Major Cities

Работа с атрибутивной базой данных была бы неполноценной без возможности написания запросов. Для того, чтобы создать запрос, необходимо выбрать пункт меню **Table=>Query...(Таблица=>Запрос...)**. Откроется конструктор запросов. На рис. 2.11 показан пример составного запроса к таблице слоя **Major Cities**. Согласно данному запросу, из таблицы должны быть выбраны испанские города с численностью населения более 1005000 человек.

9. В ArcView также можно создавать диаграммы для визуализации и изучения данных. Для создания диаграммы должна быть открыта одна из атрибутивных таблиц. Создадим диаграмму для атрибутивной таблицы слоя **Major Cities**. Поскольку атрибутивная таблица данного слоя слишком велика, отобразим на диаграмме только испанские города. Откроем базу данных слоя и создадим соответствующий запрос на выборку

("([Country]="Spain)"). Затем нажмем кнопку **Create Chart (Создать диаграмму)** на верхней панели. При этом откроется окно с настройками диаграммы (рис. 2.12).

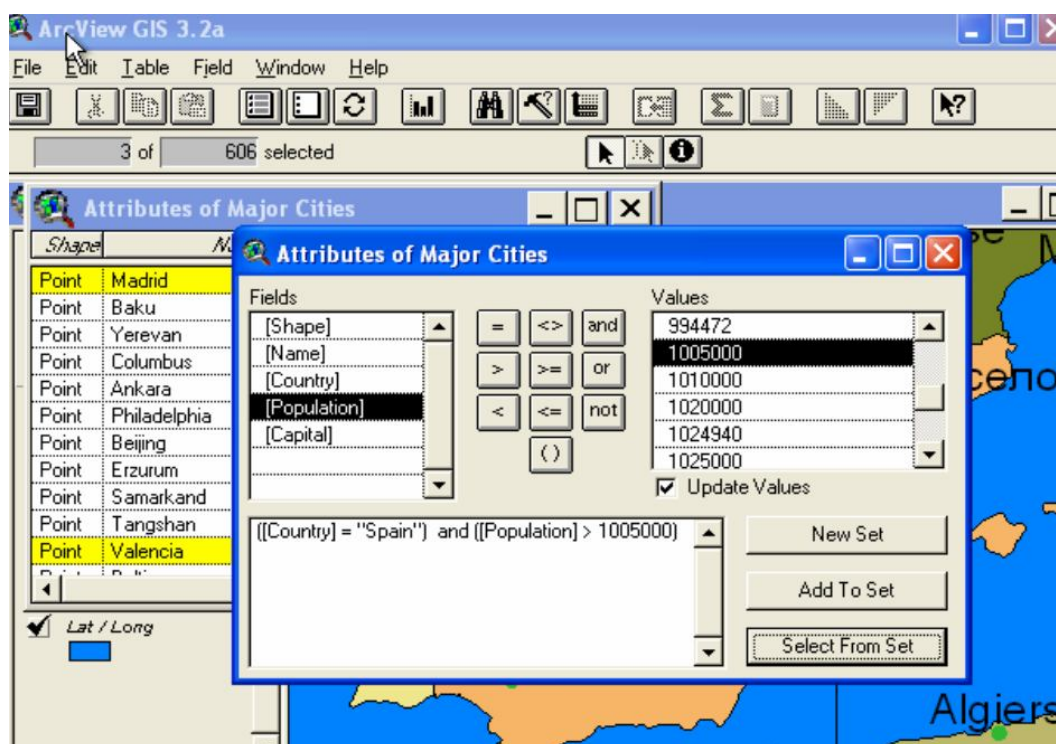


Рисунок 2.11 – Пример запроса на выборку в БД

В поле **Fields(Поля)** можно выбрать атрибуты городов, по которым будет строиться диаграмма. В данном случае можно выбрать только один атрибут – **Population (Численность населения)**. Нажмем кнопку **Add (Добавить)**, чтобы выбрать его. В поле **Label series using (Легенда)** укажите значение для легенды диаграммы. В данном случае это будет **Name (Название города)**. Нажмите **ОК**, появится окно с диаграммой (рис.2.13).

10. С помощью инструмента **Layouts (Компоновки)** в ArcView можно подготовить карту к печати. Для этого выберем пункт меню **View=>Layout...(Вид=>Компоновка...)**. Откроется окно менеджера шаблонов (рис. 2.14). В нем выберем один из шаблонов печати (в данном примере был выбран шаблон Landscape) и нажмем **ОК**. На рис. 2.15 изображена карта, подготовленная к печати.

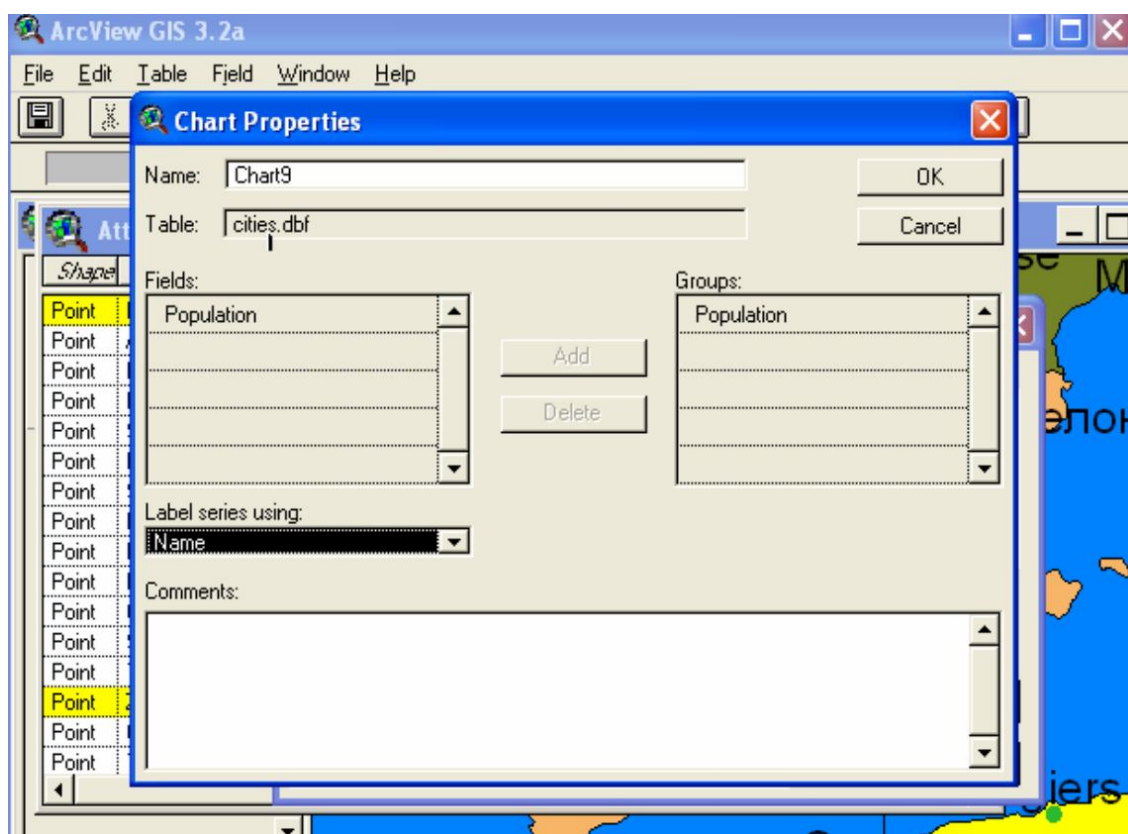


Рисунок 2.12 - Настройки параметров диаграммы

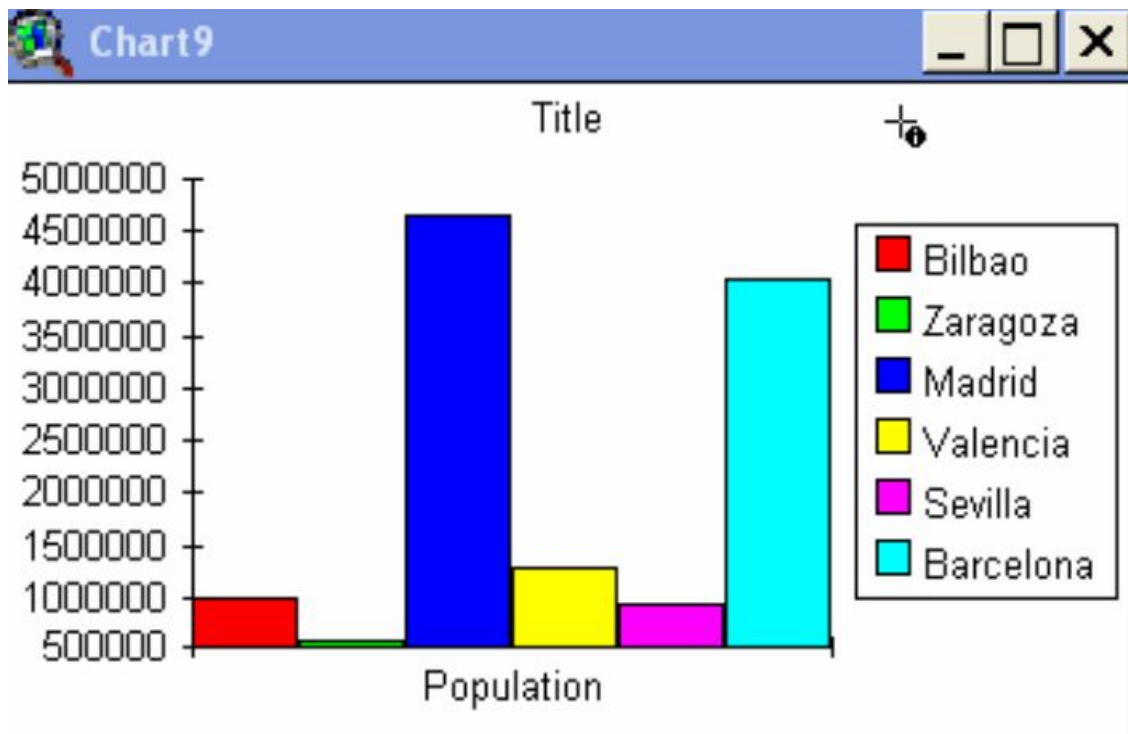


Рисунок 2.13 - Диаграмма, отражающая численность населения основных городов Испании



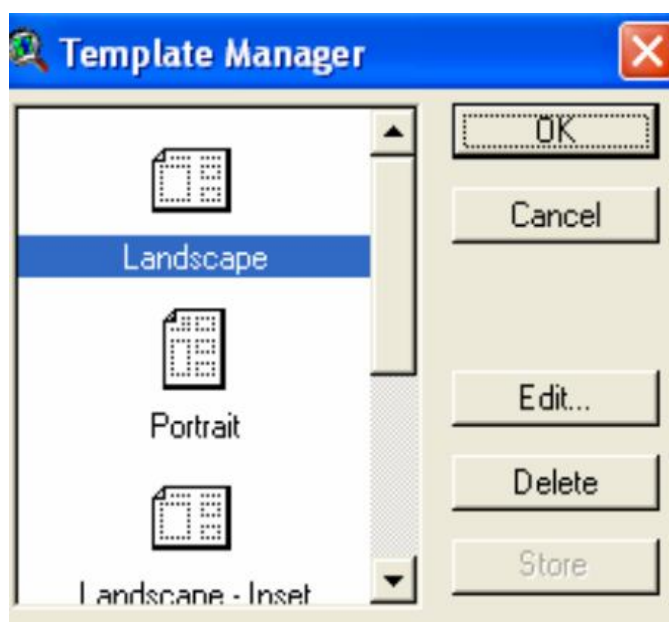


Рисунок 2.14 - Менеджер шаблонов модуля «Компоновки»

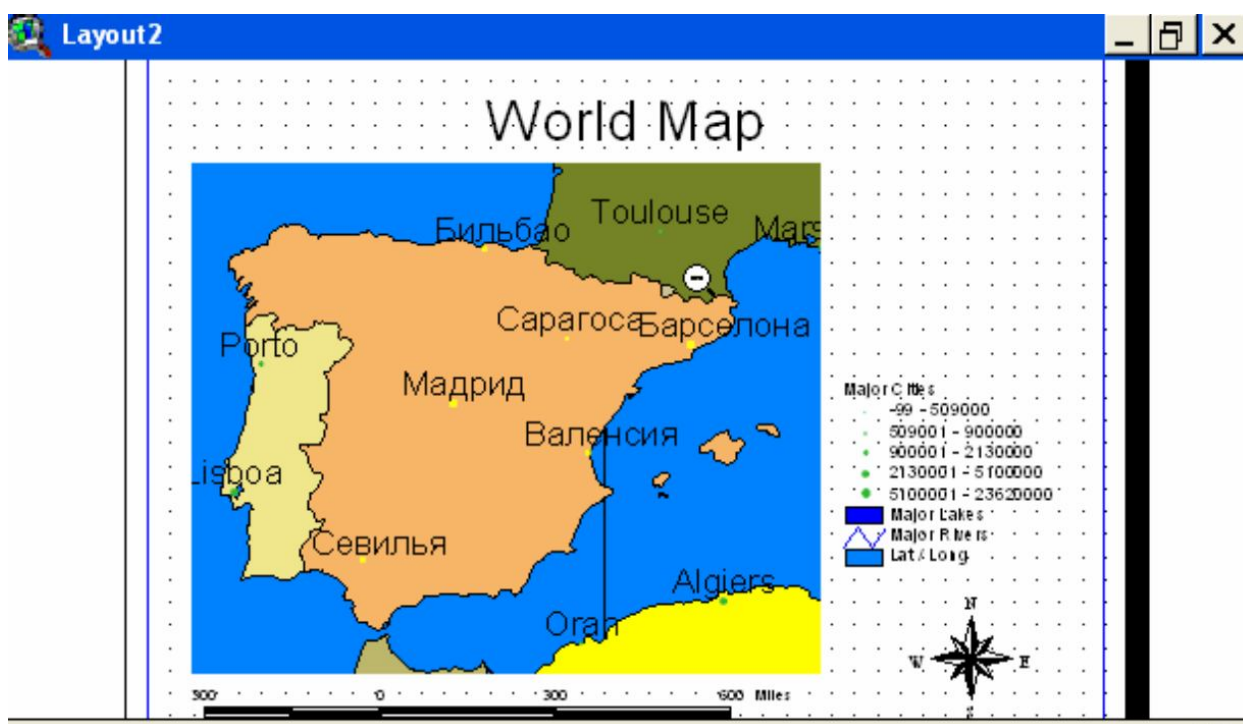


Рисунок 2.15 - Карта, подготовленная к печати

### Задание для самостоятельной работы

1. Получить у преподавателя название страны (фрагмент карты мира) для выполнения лабораторной работы.

2. Провести редактирование и оформление полученного фрагмента в соответствии с рассмотренным примером.
3. Реализовать различные виды запросов к БД (по заданию преподавателя), построить соответствующие диаграммы.
4. Оформить отчет с использованием распечатанной карты, подготовленной в модуле «Компоновки».

### **Контрольные вопросы**

1. Охарактеризуйте приложение ArcView с точки зрения его структуры и функционального назначения.
2. Что входит в ядро типовой геоинформационной системы?
3. Перечислите основные модули системы ArcView?
4. Дайте краткую характеристику назначения и функционала для каждого из модулей ArcView.
5. Перечислите основные функции ArcView для оформления и редактирования внешнего вида цифровой карты. Как происходит редактирование легенды карты?
6. Какой модуль ArcView отвечает за управление атрибутивными базами данных?
7. Опишите инструменты ArcView для построения запросов к атрибутивным базам данных.
8. Что означает динамическая связь данных между модулями ArcView?
9. Какой модуль ArcView отвечает за подготовку цифровых карт к печати?
10. Перечислите основные действия, требуемые для подготовки цифровой карты к печати.

### 3. Разработка функций пространственного анализа и язык скриптов Avenue.

**Цель работы:** изучить основы работы со встроенным языком программирования Avenue, разработать и внедрить скрипты в меню проекта ArcView.

#### Теоретическая часть

Avenue представляет собой встроенный объектно-ориентированный язык программирования системы ArcView. Особенность Avenue, (как и всех объектно-ориентированных языков) - идентификация объектов а, затем, посылка к ним запросов. В отличие от процедурного программирования, в котором упор сделан на функции (процедуры), в Avenue вместо непосредственного вызова функций с аргументами посылается запрос к объекту. Когда объект получает этот запрос, он выполняет какое-то действие. Объекты ArcView являются элементами иерархии классов, которые организованы по функциональным категориям, связанным со всеми аспектами приложения [7].

Объект - это элемент, который представляет что-либо, с чем Вы работаете в ArcView. К объектам относятся:

- элементы интерфейса (элементы управления и окна документов);
- компоненты проекта (виды);
- базовые элементы Avenue, (числа, даты, логические выражения, строки, или графические примитивы типа точек, линий и т.д.).

Для каждого класса в иерархии классов ArcView задаются характеристики, общие для всех его объектов – экземпляров. Например, класс *Doc* имеет 6 подклассов: *Chart*, *Layout*, *Project*, *Script*, *Table* и *View*. Каждый из этих подклассов наследует все характеристики класса *Doc* и дополнительные атрибуты и запросы, присущие данному подклассу.

Запросы позволяют создавать, управлять или получать информацию об объектах. Для каждого класса Avenue имеются запросы, действующие с данным классом или элементами этого класса. Запросы инициируют методы, присущие классу.

Запрос к объекту в Avenue возвращает другой объект. В некоторых случаях эти объекты представляют собой информацию о других объектах. Например, оператор:

$$theName = theView.GetName$$

возвращает объект строки, являющийся именем объекта "*theView*". Переменная *theName* получает этот объект. Аналогично, выражение:

$$myNumber = 2 + 5$$

возвращает числовой объект, который является результатом запроса на сложение двух числовых объектов. ArcView приписывает этот числовой объект переменной *myNumber*. Некоторые запросы приводят к возврату объекта *NIL*.

Запросы сгруппированы в две категории:

- созданные непосредственно для класса;
- созданные для объектов или экземпляров этого класса.

Запросы к классу не нацелены на отдельный объект, но обеспечивают механизм для более общих функциональных возможностей. В запросах к классу фактическое имя класса (например, *View*, *LineFile*, *List*) является целью ("target") запроса. Запросы к классу обычно создают экземпляры класса или возвращают информацию о воздействии на класс. Пример:

$$FileName.Make("myfile.txt")$$

возвращает новый экземпляр *FileName*.

Запросы к экземпляру позволяют выполнять действие с объектом. Например, можно максимизировать окно показа Вида с именем *View1*, можно сделать тему под названием "Города" в модуле «Виды» активной, можно изменить цвет графики при выводе на экран. Запросы к экземпляру обычно выполняют некоторое действие или изменяют состояние объекта.



Различные классы поддерживают различные запросы. Например, можно послать запрос к модулю «Виды» на зуммирование фрагмента цифровой карты, однако, при этом, соответствующий объект должен «понимать» и уметь обрабатывать запрос зуммирования.

В некоторых случаях, можно послать одинаковый запрос (такой как, *Save*(сохранить), *Open*(открыть), или *Make*(создать)) к объектам, которые принадлежат разным классам. Эти запросы приводятся в действие по-разному, но приводят к одинаковому общему результату. Это так называемый полиморфизм. Например, запрос *Make*, сделанный для класса *List* выполнит иную функцию, чем запрос *Make*, сделанный для *TextFile*, но оба запроса *Make* сгенерируют новый экземпляр целевого класса.

Существуют следующие формы написания запросов [7]:

- **Постфиксная форма (Postfix)**, где запрос используется после объекта.

- Синтаксис : *Object.Request* (объект.запрос)

- Пример: *av.GetProject*

Постфиксные запросы также могут быть объединены в виде цепочки:

- Синтаксис : *Object.Request.Request...Request*

Каждая пара “запрос-объект” (*object.request*) возвращает другой объект, а несколько пар “запрос-объект” выполняются по порядку, слева направо.

- Пример: запрос *theView.FindTheme("Roads")* возвращает объект темы.

Запрос *SetVisible* делает объект темы невидимым.

- *theView.FindTheme( "Roads" ).SetVisible( false )*

- **Инфиксная форма (Infix)** для арифметических операторов.

- Синтаксис: *Object Request Object*

- Пример: *2 + 3*

Выражения выполняются слева направо. Avenue не поддерживает обычного приоритета при порядке выполнения операторов.

- **Префиксная форма (Prefix)** - запрос стоит перед объектом.

- Синтаксис: *Request Object*

- Пример: *not True*

Рассмотрим пример простейшей программы на языке Avenue.

```
theView = av.FindDoc("View1")  
if (theView<>nil) then  
else  
    MsgBox.Info( "No themes", "")  
end
```

Здесь *av* - зарезервированное слово, которое представляет в ArcView объект прикладной программы. Запуская ArcView, создается объект прикладной программы, который остается, пока работает ArcView. При записи сценария *av* - отправная точка. Например, *av.GetProject* возвращает текущий проект или *av.GetActiveDoc* возвращает активный вид. Можно минимизировать или максимизировать окно прикладной программы, обращаясь к *av.Minimize* или *av.Maximize*. У этой этого объекта есть метод *FindDoc*. Этот метод ищет вид. Возвратит нуль, если не нашел.

### Порядок выполнения работы

1. Будем работать с проектом, созданным в предыдущей лабораторной работе. Загрузим проект и выберем вкладку **Scripts (Скрипты)**. Создадим новый скрипт на основе скрипта-примера из библиотеки примеров ArcView. Дважды щелкнем по вкладке **Scripts (Скрипты)**. Будет автоматически создан новый скрипт и открыто окно его редактирования. Нажмем **Load Text File (Загрузить текстовый файл)** на верхней панели инструментов. Откроется окно загрузки файла скрипта. Выберем уже созданный разработчиками скрипт, как это показано на рисунке 3.1 (файл *tblinput.ave*).

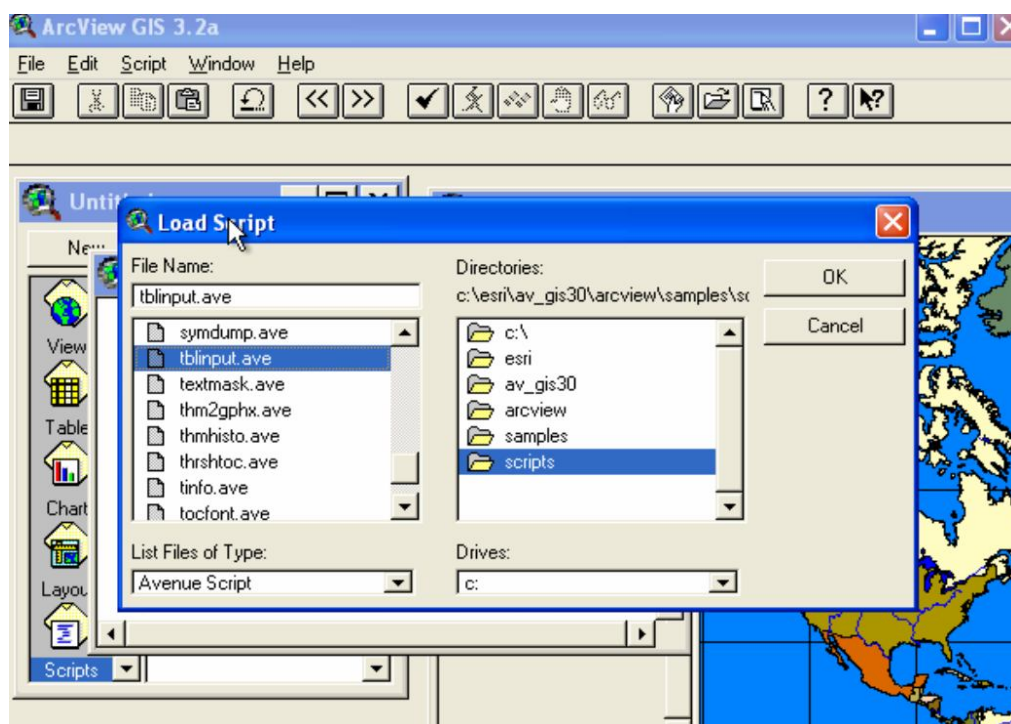


Рисунок 3.1 - Добавление скрипта в проект

2. Загруженный скрипт (рис. 3.2) создает таблицу с данными, которые вводит пользователь. Создается пустая таблица, и используется внутренний объект ArcView *MsgBox.MultiInput*, чтобы запросить у пользователя информацию, которая будет добавлена. Форма ввода будет отображаться до тех пор, пока пользователь не нажмет кнопку **Cancel (Отмена)**. После нажатия на кнопку новая таблица будет добавлена в проект и откроется.

3. Откомпилируем скрипт, нажав **Compile (Компиляция)**. Теперь необходимо создать на панели инструментов кнопку, при нажатии на которую скрипт будет запущен. Кнопка – не единственный элемент интерфейса, к которому может быть привязан скрипт. Такая привязка может быть осуществлена к любому типу элементу (например, пункту основного меню). Дважды щелкните левой кнопкой мыши по панели инструментов, откроется окно настройки панели (рис. 3.3). В поле **Type (Тип)** укажем тип документа, в котором будет работать скрипт. В данном случае это **Tables(Таблицы)**. Во вкладке **Category (Категория)** выберем тип создаваемого элемента управления.

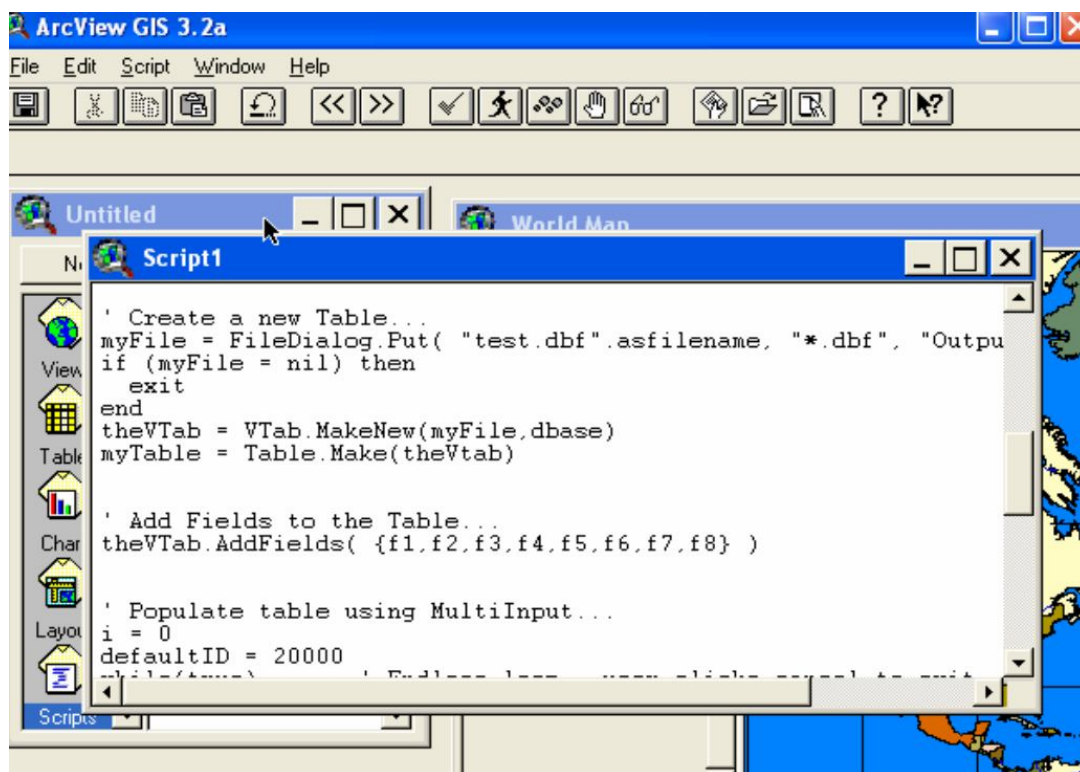


Рисунок 3.2 – Текст загруженного скрипта

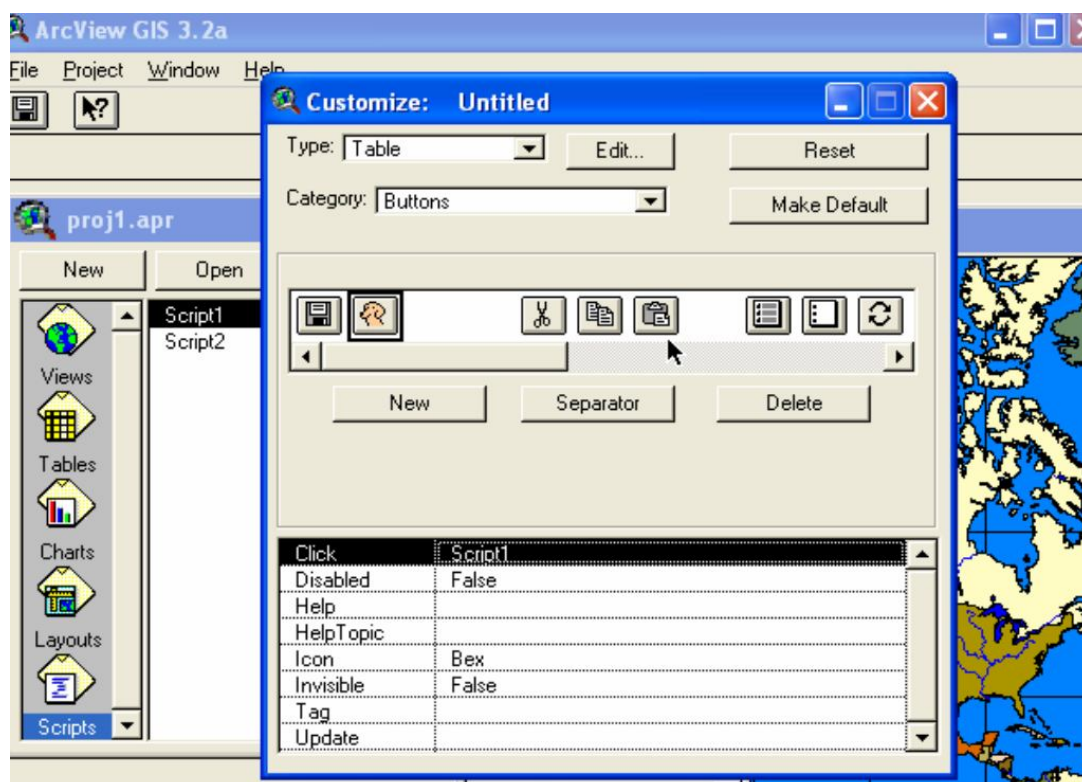


Рисунок 3.3 - Создание кнопки скрипта

В этой вкладке содержится перечень всех элементов интерфейса, доступных для привязки скриптов. В данном случае это **Buttons (Кнопки)**. Определим место, где будет находиться кнопка. Для этого переместим черную рамку, которая сначала оконтуривает первую кнопку, туда, где хотите вставить новую кнопку, и нажмем **New (Новая)**. Появится новая кнопка. Затем дважды щелкнем на поле **Click (Запустить по щелчку)** и выберем из появившегося окна менеджера скриптов (Script manager) нужный скрипт. В данном случае это Script1.

4. Запустим созданный скрипт нажав на созданную кнопку. Для этого дважды щелкнем по вкладке **Tables (Таблицы)**. Откроется новая таблица и появится панель инструментов для ее редактирования. На панели инструментов отобразится созданная кнопка. Нажмем на нее. Ход работы вызванного скрипта показан на рис. 3.4.

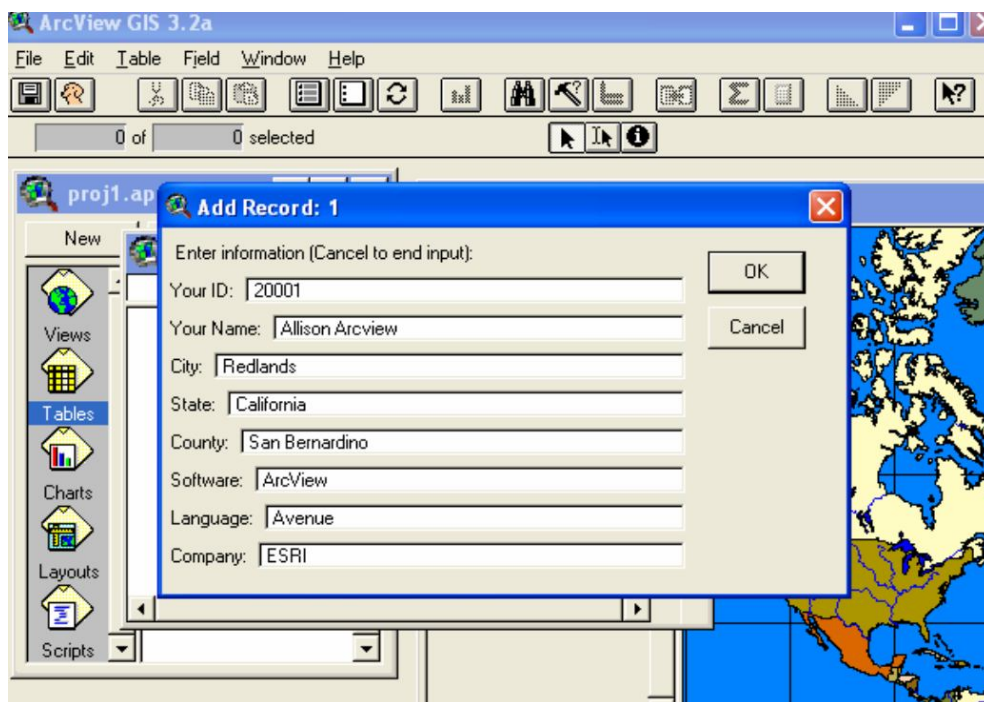


Рис. 3.4. – Результат запуска созданного скрипта

4. Отредактируем файл Script1. Оставим в таблице только 3 поля и назовем их "Name", "Patronymic", "Surname". Для корректной работы также необходимо исправить некоторые другие фрагменты скрипта: изменить формат поля f1 с `#FIELD_SHORT` на `#FIELD_CHAR`, убрать проверку на

соответствие типов данных и т.д. Текст исправленного скрипта будет выглядеть следующим образом:

*' Создать поля, которые будут добавлены в таблицу...*

*f1 = Field.Make( "Name",#FIELD\_CHAR, 25, 0 )*

*f2 = Field.Make( "Patronymic",#FIELD\_CHAR, 25, 0 )*

*f3 = Field.Make( "Surname",#FIELD\_CHAR, 25, 0 )*

*' Создать новую таблицу...*

*myFile = FileDialog.Put( "test.dbf".asfilename, "\*.dbf", "Output File")*

*if (myFile = nil) then*

*exit*

*end*

*theVTab = VTab.MakeNew(myFile,dbase)*

*myTable = Table.Make(theVtab)*

*' Добавить поля в таблицу...*

*theVTab.AddFields( {f1,f2,f3} )*

*' Заполним таблицу, используя функцию MultiInput...*

*i = 0*

*defaultID = 20000*

*while(true) ' Бесконечный цикл – для выхода из него, нужно нажать cancel...*

*i = i + 1*

*defaultID = defaultID + 1*

*l = MsgBox.MultiInput("Enter information (Cancel to end input):",*

*"Add Record:"++i.AsString,*

*{ "Name:", "Patronymic:", "Surname:"},*

*{ "Vasilii", "Ivanovich", "Ivanov"} )*

*' Если пользователь нажал cancel, то завершить ввод...*

*if (l.count < 1) then*

*break*

*end*

*' Добавить записи в таблицу...*

*rec = theVTab.AddRecord*

*theVTab.SetValue(f1, rec, l.Get( 0 ) )*

*theVTab.SetValue(f2, rec, l.Get( 1 ) )*

*theVTab.SetValue(f3, rec, l.Get( 2 ) )*

*end*

*myTable.GetWin.Open*

Ход работы скрипта представлен на рис. 3.5.

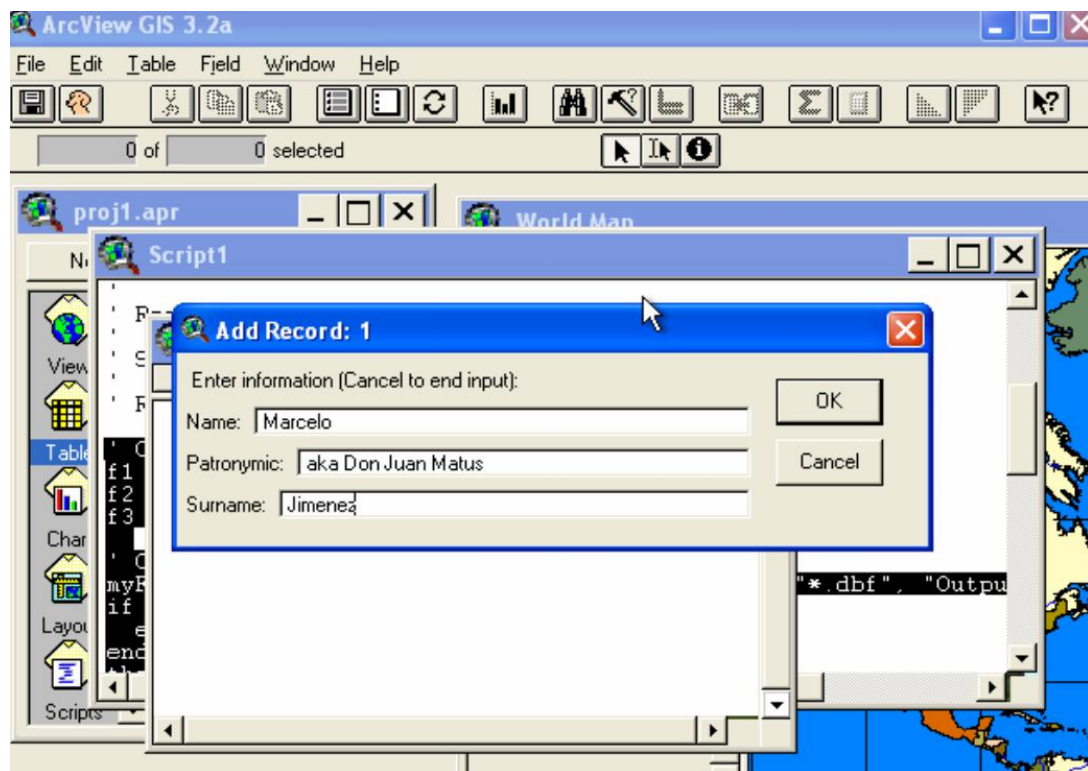


Рисунок 3.5 - Запущенный модифицированный скрипт

### **Задание для самостоятельной работы**

1. Получить у преподавателя задания для выполнения самостоятельной работы.
2. Связать разработанные скрипты с элементами управления (кнопками), откомпилировать, отладить и запустить скрипты.
3. Внести в скрипты собственные модификации, откомпилировать, отладить и запустить модифицированный скрипт.
4. Оформить отчет.

### **Контрольные вопросы**

1. Назовите основные свойства языка Avenue.
2. В чем отличия объектно-ориентированных скриптов от процедурно-ориентированных?
3. Что понимается под объектом геоинформационного проекта ArcView? Приведите примеры.
4. Как производится управление объектами в проекте ArcView?
5. Назовите две основных категории запросов в скриптах Avenue?
6. Что в языке Avenue понимается под полиморфизмом?
7. Перечислите основные формы написания запросов в скриптах Avenue?
8. Опишите порядок действий, необходимый для создания скрипта.
9. Опишите порядок действий, необходимый для привязки скрипта к элементу интерфейса.
10. К каким элементам интерфейса может быть привязан скрипт Avenue?



## Список литературы

1. Коновалова Н.В., Капралов Е.Г. Введение в ГИС. Учебное пособие. – Изд-во, ПетрГУ, Петрозаводск, 2013. - 148с.
2. Бершадский А.М., Бождай А.С. Геоинформационные технологии и системы. Учебное пособие – Пенза, изд-во ПензГУ, 2001. – 128 с.
3. Капралов Е.Г., Кошкарев А.В., Заварзин А.А. Основы геоинформатики. Книга 1. – М.: Academia, 2004. – 352 с.
4. Геоинформатика. В 2 книгах. Книга 1 / Под ред. В.С.Тикунова – М.: Academia, 2010, 416 с.
5. Берлянт А.М. Геоиконика. - М.: «Астреля», 1996. - 208с.
6. Введение в ArcView. Пер. с англ. СП «ДАТА+» М.: 1995. - 114с.
7. Введение в Avenue. Пер. с англ. СП «ДАТА+» М.:1995 - 126с.