

**Д. С. ПАРЫГИН, А. В. ИГНАТЬЕВ  
Н. П. САДОВНИКОВА, А. С. ГУРТЯКОВ**

# **ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ**

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ВОЛГОГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Д. С. Парыгин, А. В. Игнатьев  
Н. П. Садовникова, А. С. Гуртяков

# ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ

*Учебно-методическое пособие*



Волгоград  
2023

Рецензенты:

кафедра «Информационные системы и математическое моделирование»  
Волгоградского института управления – филиала Российской академии  
народного хозяйства и государственной службы при Президенте РФ,  
зав. кафедрой канд. техн. наук, доцент *О. А. Астафурова*;  
профессор кафедры «Методика преподавания математики и физики, ИКТ»  
Волгоградского государственного социально-педагогического  
университета д-р пед. наук *Т. М. Петрова*

Печатается по решению редакционно-издательского совета  
Волгоградского государственного технического университета

**Парыгин, Д. С.**

Геоинформационные системы : учеб.-метод. пособие / Д. С. Парыгин, А. В. Игнатьев, Н. П. Садовникова, А. С. Гуртяков ; ВолгоГТУ. – Волгоград, 2023. – 128 с.

ISBN 978-5-9948-4602-5

Материалы пособия посвящены вопросам сбора и подготовки географических данных, организации данных в геоинформационных системах, основам геопространственного анализа. Приведены примеры задач пространственного анализа и инструменты для их решения. Описаны возможности современных геоинформационных систем.

Пособие предназначено для студентов очной, очно-заочной и заочной форм обучения по направлениям 09.03.01 «Информатика и вычислительная техника» (в том числе программа «Инженерия систем искусственного интеллекта»), 09.03.02 «Информационные системы и технологии», 09.04.01 «Информатика и вычислительная техника», 09.04.02 «Информационные системы и технологии» (в том числе профиль «Искусственный интеллект в проектировании городской среды»), 08.04.01 «Строительство» (профиль «Организация информационного моделирования в строительстве»).

Ил. 77. Табл. 11. Библиогр.: 26 назв.

ISBN 978-5-9948-4602-5

© Волгоградский государственный  
технический университет, 2023

© Д. С. Парыгин, А. В. Игнатьев,  
Н. П. Садовникова, А. С. Гуртяков, 2023

## ОГЛАВЛЕНИЕ

|                                                                                                       |    |
|-------------------------------------------------------------------------------------------------------|----|
| Введение.....                                                                                         | 5  |
| 1. Курс лекций по дисциплине.....                                                                     | 7  |
| 1.1. Основы геоинформатики.....                                                                       | 7  |
| 1.2. Функции и структура ГИС.....                                                                     | 8  |
| 1.3. Геопространственные данные.....                                                                  | 10 |
| 1.3.1. Растровые модели данных.....                                                                   | 16 |
| 1.3.2. Векторные модели данных.....                                                                   | 18 |
| 1.4. Задачи анализа пространственных данных.....                                                      | 25 |
| 1.5. Картографические сервисы и пространственные данные.....                                          | 29 |
| 1.6. Управление данными в ГИС (на примере OpenStreetMap).....                                         | 35 |
| 1.7. Технологические решения на основе пространственных данных.....                                   | 38 |
| Вопросы.....                                                                                          | 43 |
| 2. Задания для практических занятий.....                                                              | 44 |
| 2.1. Модели пространственных данных.....                                                              | 44 |
| 2.2. Пространственные базы данных.....                                                                | 45 |
| 2.3. Методы и средства визуализации данных.....                                                       | 45 |
| 2.4. Основы ведения территориальных кадастров.....                                                    | 46 |
| 2.5. Прикладные аспекты ГИС.....                                                                      | 46 |
| 2.6. Применение ГИС для решения прикладных задач.....                                                 | 47 |
| 3. Методические указания к лабораторным работам.....                                                  | 48 |
| Лабораторная работа № 1 «Основы работы с ГИС».....                                                    | 48 |
| Задания.....                                                                                          | 56 |
| Вопросы к отчету.....                                                                                 | 57 |
| Лабораторная работа № 2 «Отображение пространственных данных».....                                    | 58 |
| Задания.....                                                                                          | 69 |
| Вопросы к отчету.....                                                                                 | 69 |
| Лабораторная работа № 3 «Платформа OpenStreetMap:<br>основные функции, инструменты и библиотеки»..... | 70 |
| Задания.....                                                                                          | 74 |
| Вопросы к отчету.....                                                                                 | 76 |
| Лабораторная работа № 4 «Задачи пространственного анализа».....                                       | 77 |
| Задания.....                                                                                          | 89 |
| Вопросы к отчету.....                                                                                 | 90 |

|                                                                                                                    |     |
|--------------------------------------------------------------------------------------------------------------------|-----|
| Лабораторная работа № 5 «Трехмерная визуализация в ГИС».....                                                       | 91  |
| Задания.....                                                                                                       | 98  |
| Вопросы к отчету.....                                                                                              | 98  |
| Лабораторная работа № 6 «Получение данных пространственного анализа<br>на основе дистанционного зондирования»..... | 99  |
| Задания.....                                                                                                       | 103 |
| Вопросы к отчету.....                                                                                              | 104 |
| 4. Методические указания к контрольной работе.....                                                                 | 105 |
| 4.1. Настройка и запуск системы моделирования OSMLS.....                                                           | 105 |
| 4.2. Создание и запуск тестового модуля.....                                                                       | 106 |
| 4.3. Задание.....                                                                                                  | 112 |
| Список рекомендуемой по курсу литературы.....                                                                      | 123 |

## ВВЕДЕНИЕ

Первые геоинформационные системы (ГИС) были созданы в Канаде, США и Швеции для изучения природных ресурсов в середине 1960-х годов. Сегодня ГИС-технологии получили широкое распространение во многих сферах и обеспечивают поддержку работы с любыми данными, имеющими пространственно-временную привязку. С их помощью решаются задачи интеграции картографической информации, данных дистанционного зондирования и экологического мониторинга и пр. ГИС представляет собой централизованную базу данных пространственных объектов и инструмент, который предоставляет возможности хранения, анализа и обработки любой информации, связанной с тем или иным объектом. Эффективность использования ГИС, возможность реализовать требуемые функции, построить полноценную информационную систему, интегрировать ее в существующую информационную инфраструктуру существенным образом зависит от свойств и качества программного обеспечения ГИС.

Можно выделить наиболее известных поставщиков программного обеспечения для ГИС: Autodesk Inc (AutoCAD Map, AutoCAD Civil, MapGuide, ESRI (USA) (ArcGIS), MapInfo Corp (MapInfo), ЦГИ ИГ РАН (GeoGraph/ GeoDraw/GeoConstructor), GeoSpectrum International (ГИС Panorama), НИИПМК (ГИС Terra).

С каждым годом растет потребность в Web-сервисах, использующих пространственные данные и метаданные для решения самых разных прикладных задач:

- управление земельными ресурсами, ведение земельных кадастров;
- инвентаризация, учет, планирование размещения объектов распределенной производственной инфраструктуры;
- проектирование, планировка электросетей;
- управление наземным, воздушным и водным транспортом;

- экологический мониторинг;
- оперативная оценка распространения пожаров;
- прогноз полезных ископаемых, оценка и подсчет запасов месторождений в геологии;
- прогнозирование урожайности и пр.

В настоящее время ГИС является основой муниципальной информационной системы, так как интегрирует все пространственные данные по объектам городской территории и может обеспечить поддержку для анализа этих данных и решения аналитических задач.

Современные тенденции развития ГИС-технологий связаны с интеллектуальным анализом данных и искусственным интеллектом (ИИ). Большие объемы данных, интегрированные в ГИС, обеспечивают возможность создания и обучения моделей ИИ для прогнозирования, бизнес-аналитики и создания систем управления на основе данных. Умение работать с инструментами анализа пространственных данных является одной из наиболее перспективных и востребованных компетенций ближайшего времени.

Цель создания данного пособия – ознакомление студентов и других заинтересованных лиц с методами и технологиями ГИС, принципами разработки программных систем на их основе.

---

## КУРС ЛЕКЦИЙ ПО ДИСЦИПЛИНЕ

---

### 1.1. Основы геоинформатики

Геоинформатика – это наука, технология и производственная деятельность по научному обоснованию, проектированию, созданию, эксплуатации и использованию географических информационных систем, по разработке геоинформационных технологий, по приложению геоинформационных систем (ГИС) для практических и научных целей [1].

Основные части геоинформатики:

1. Общая геоинформатика.
2. Прикладная геоинформатика.
3. Специальная геоинформатика.

Общая геоинформатика – это раздел геоинформатики, занимающийся исследованием и разработкой научных основ, концепций, обобщенным анализом геоинформационных систем безотносительно к их прикладному характеру.

Прикладная геоинформатика изучает практические методы работы с геоинформационными системами и геоинформационными технологиями

Специальная геоинформатика служит основой для анализа систем и методов обработки пространственных данных.

В основе теории геоинформатики как учения о ГИС лежат несколько базовых понятий. К ним относятся понятия пространственного объекта, пространственных данных, моделей пространственных данных, функций их обработки, включая базовые функции пространственного анализа и геомоделирования как ядра ГИС.

Проблемы разработки, функционирования и использования ГИС находятся на стыке трех областей научных знаний: компьютерные науки, науки о земном пространстве, области ГИС-приложений (табл. 1.1.1) [2].

Таблица 1.1.1

**Базовые области научных знаний ГИС**

| Компьютерные науки                                                                                                                                                                                          | Науки о земном пространстве                                                                                                             | Области ГИС-приложений                                                                                                                                                                                                                                                                           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Информатика и программирование, математическое моделирование, операционные системы, текстовые и графические редакторы, электронные таблицы, базы данных, информационные сети, обработка изображений и т. д. | Геодезия, картография, география, аэрокосмическая съемка, дистанционное зондирование Земли, глобальные системы позиционирования и т. д. | Управление территорией, градостроительство и архитектура, инженерная инфраструктура, управление недвижимостью, транспорт и логистика, экология, природные ресурсы, демографические исследования, оборона, сельское хозяйство, правонарушения, чрезвычайные ситуации и т. д. (около 80 дисциплин) |

## 1.2. Функции и структура ГИС

Геоинформационные технологии можно определить как совокупность программно-технологических средств получения новых видов информации об окружающем мире. Геоинформационные технологии предназначены для повышения эффективности: процессов управления, хранения и представления информации, обработки и поддержки принятия решений.

Основные функции ГИС [3]:

- 1) функции автоматизированного картографирования;
- 2) функции пространственного анализа;
- 3) функции управления данными.

Функции автоматизированного картографирования используют для производства высококачественных карт и изображений. Функции автома-

тизированного картографирования должны включать векторно-растровые преобразования; преобразования координатной системы, картографических проекций и масштабов; «склейки» отдельных листов; осуществление картометрических измерений (вычисления площадей, расстояний); размещение текстовых надписей и внемасштабных картографических знаков; формирование макета печати.

Функции пространственного анализа должны обеспечивать совместное использование и обработку картографических и атрибутивных данных и включать анализ географической близости, анализ сетей, топологическое наложение полигонов, интерполяцию и изолинейное картографирование полей, вычисление буферных зон.

Функции управления данными должны обеспечивать работу с атрибутивными (неграфическими) данными ГИС с целью их отбора, обновления и преобразования для производства стандартных и рабочих отчетов. Функции управления данными должны включать пользовательские запросы, генерацию пользовательских документов, статистические вычисления, логические операции, поддержание информационной безопасности, стандартных форм запросов и представления их результатов. В общем случае ГИС должна состоять из следующих четырех подсистем [4]:

- сбора, подготовки и ввода данных;
- хранения, обновления и управления данными;
- обработки, моделирования и анализа данных;
- контроля, визуализации и вывода данных.

Рабочей средой при работе с ГИС-платформой является *Проект*. Проект может включать в себя все информационные компоненты, на которых строятся ГИС-технологии. Основной структурной единицей ГИС является тематический слой, понятие которого тесно связано с более общим понятием покрытия, несущим в себе объектное содержание (например, единица административно-территориального деления).

*Покрывтие (Coverage)* – цифровая модель единицы хранения базы векторных данных ГИС, хранит в виде записей все объекты первичного (точки, дуги, полигоны) и вторичного уровня (координаты опорных точек, аннотации и т. д.) некоторого пространственного объекта и структуру отношений между ними, в том числе топологических. Пустое покрытие – покрытие, в котором отсутствуют какие-либо пространственные объекты.

*Слой (Map Layer)* – покрытие, рассматриваемое в контексте его содержательной определенности (растительность, рельеф, административное деление и т. п.) или его статуса в среде редактора (активный слой, пассивный слой).

Слой, как правило, однороден не только по тематике, но и по типам объектов (точечные, линейные, полигональные, растровые). Информационные компоненты могут быть внешними (векторные и растровые слои, таблицы, библиотеки символов) или внутренними (специальные типы слоев, запросы, макросы, карты, макеты печати и т. д.).

При создании нового проекта необходимо подключить или создать новые слои.

Слои (или покрытия) объединяются в *цифровые карты*. Карты могут не поддерживать в своей структуре покрытия, но в этом случае берут часть функций или все функции покрытий на себя. В рамках одного проекта может быть создано неограниченное количество карт. Дополнительные возможности управления дает панель управления слоями (*Легенда*) карты, на которой представлены все слои.

### **1.3. Геопространственные данные**

Основной единицей в ГИС являются данные. Данные (лат. datum – акт) – совокупность фактов и сведений, представленных в каком-либо формализованном виде для их использования в науке и других сферах че-

ловеческой деятельности. Под данными в среде ГИС понимается информация, известная об объектах реального мира; результаты наблюдений и измерений этих объектов. Элемент данных содержит две главные компоненты: географические сведения, описывающие его местоположение в пространстве относительно других объектов (пространственные данные), и атрибутивные данные, которые описывают сущность, характеристики, переменные и значения (рис. 1.3.1).

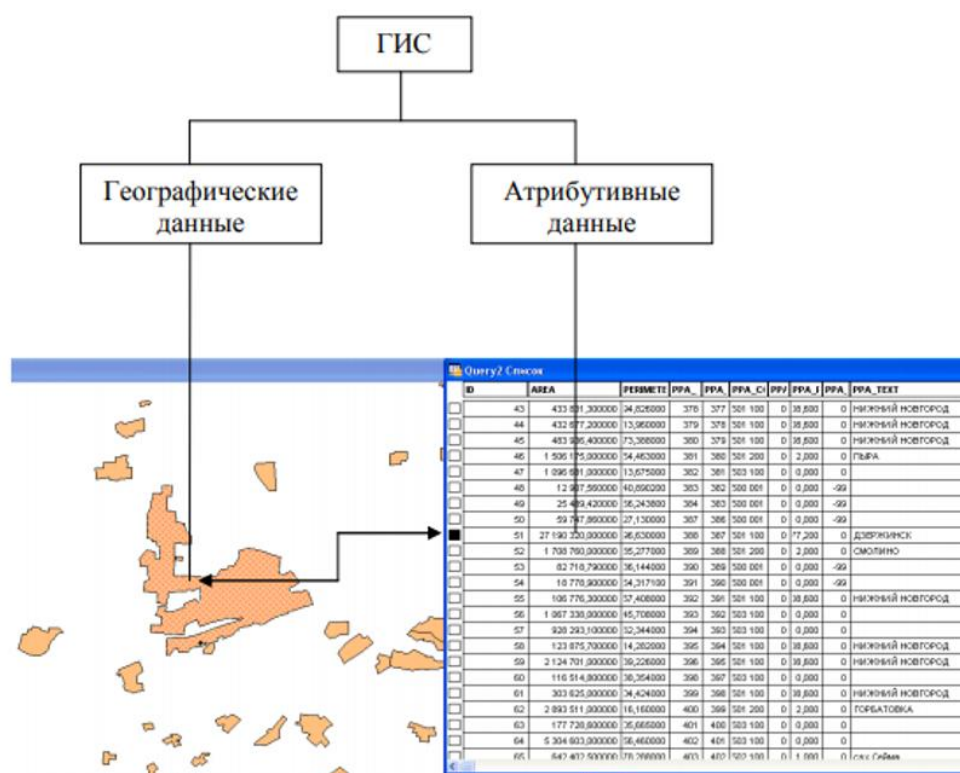


Рис. 1.3.1. Компоненты данных в ГИС

Создание географических данных – наиболее трудоемкая, дорогостоящая и ответственная часть ГИС. По мировым оценкам стоимость географических данных составляет до 80–90 % стоимости системы. Поэтому при создании ГИС исключительно важными являются: 1) задачи сохранности данных длительный период времени – до 80 лет; 2) возможность обмена данными между системами, существующими сегодня; 3) передача данных в будущие системы в условиях, когда программно-техническое обеспечение существенно меняется через 2–3 года [3].

К процедурам анализа данных в ГИС относятся: запись данных и их извлечение по запросам пользователей, формирование запросов об определенных характеристиках объектов, прогнозирование и моделирование различных процессов.

Источниками пространственных данных являются карты, аэрофотоснимки, космические снимки, результаты полевых исследований и т. д. (рис. 1.3.2) [4].

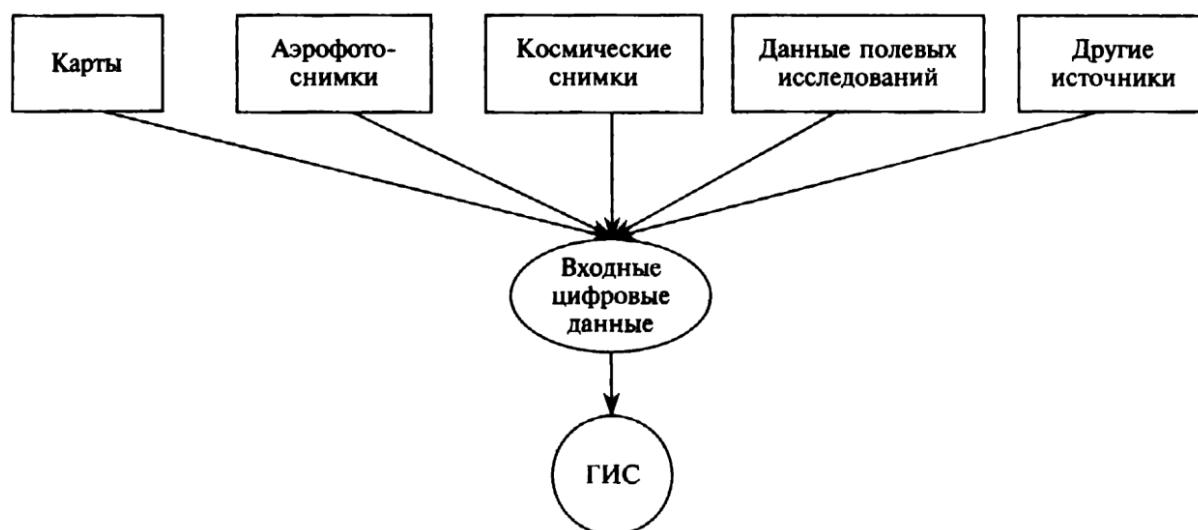


Рис. 1.3.2. Получение данных ГИС

Для качественного представления информации в ГИС необходимо отслеживать качество получаемых данных. Снижение качества может быть

связано с разными причинами. Основные источники ошибок можно отнести к одному из типов [5]:

1. Концептуальные ошибки.
2. Ошибки, связанные с источником данных.
3. Ошибки кодирования данных.
4. Ошибки, связанные с коррекцией и преобразованием данных.
5. Ошибки, возникающие на этапе обработки и анализа данных.
6. Ошибки вывода данных.

*Концептуальные ошибки.* Ошибки данного типа возникают при интерпретации и моделировании пространственных объектов. Объекты могут по-разному восприниматься разными людьми, и это будет проявляться в данных.

*Ошибки, связанные с источником данных.* В этом случае тип ошибок непосредственно зависит от источника данных. Так, при полевых исследованиях возникают, в основном, инструментальные ошибки и ошибки, связанные с человеческим фактором. В данных дистанционного зондирования и на аэрофотоснимках могут содержаться ошибки, связанные с неправильной пространственной привязкой или неверным дешифрированием. Помимо этого, источником ошибок служит изменчивость некоторых объектов во времени. В картах содержатся как позиционные, так и тематические ошибки. Как правило, они возникают из-за технических сбоев, неправильных действий оператора и несовершенства существующих методов картографии.

*Ошибки кодирования данных.* Кодированием данных называется процесс их преобразования в формат ГИС. Кодирование данных является одним из основных источников ошибок. Оцифровка карт также относится к процедурам кодирования. Ошибки могут возникать также из-за неточного совмещения листов карты или из-за недостаточной разрешающей спо-

способности растровых сканеров, которые применяются для автоматической оцифровки.

*Ошибки, связанные с редактированием и преобразованием данных.*

При использовании автоматизированных методов поиска ошибок в растровых ГИС возникают проблемы, связанные с шумом, или, иначе, ошибочной классификацией. Шум может быть регулярным и случайным.

После редактирования растровые данные могут быть преобразованы в векторные, а векторные – в растровые. При преобразовании растровых данных в векторные возникают топологические неоднозначности. При обратном преобразовании большое значение с точки зрения ошибок имеют размеры точки растра и метод растривания. Помимо топологических ошибок, преобразование векторных карт в растровые ведет к потере небольших полигонов и появлению искажений, вызванных случайным смещением и поворотом растра.

Иногда преобразование данных необходимо для их передачи на другую систему. В подобных случаях перевод базы данных ГИС из одного программного пакета в другой может привести к появлению так называемых технических ошибок.

### ***Объектная и полевая модели данных***

Способы представления данных в ГИС называются моделями. Модели данных связывают физическое описание объектов с их цифровым представлением в ГИС. Процесс создания модели состоит из следующих этапов [5]:

1. Идентификация пространственных объектов в контексте решаемой задачи и определение способа их представления в концептуальной модели.
2. Выбор растрового или векторного типа представления концептуальной модели.

3. Выбор структуры пространственных данных для хранения модели на компьютере.

Структура пространственных данных в модели – это физический способ кодирования параметров объектов, который используется для их хранения и анализа.

Объекты (например, здания, дороги, города и лесные массивы) всегда дискретны, а их местоположение точно определено. Свойства, к которым относят топографию, население, температуру и уровни выпадения осадков, обычно распределены непрерывным образом по большой площади. С учетом этих различий для описания объектов реального мира в географической базе данных используют модели двух типов – объектная и полевая [5].

*Объектная модель данных.* В объектной модели географическое пространство заполнено отдельными хорошо идентифицируемыми объектами. Каждый пространственный объект имеет четкие границы, а его свойства можно описать с помощью одного или нескольких параметров, которые называются атрибутами. Для построения объектной модели используют результаты полевых исследований, а также данные, полученные с помощью фотограмметрических методов, методов аэрофотосъемки, дистанционного зондирования и оцифровки карт.

*Полевая модель данных.* В полевой модели географическое пространство заполнено данными, которые непрерывно распределены в пространстве и не имеют четко определенных границ. Такие распределения называются полями. Для получения данных этого типа используют аэрофотосъемку, методы дистанционного зондирования, карты и полевые измерения в ряде выборочных точек (например, узлов нерегулярной триангуляционной сети). Набор полевых данных можно сформировать также косвенными методами – путем интерполяции, повторной классификации и повторной выборки исходных измерений.

### 1.3.1. Растровые модели данных

Растровая модель описывает пространственное распределение некоторой характеристики. Пространство разбивается регулярной сеткой ячеек, в каждой ячейке фиксируется значение исследуемого параметра (путем статистического осреднения, семплирования в центре ячейки и т. п.). Растровые данные могут быть как количественными (например, поле температуры), так и качественными (например, растр классифицированного снимка, каждая ячейка которого фиксирует принадлежность к тому или иному типу объекта). Таким образом, растровая модель является пространственно-ориентированной (или феномен-ориентированной) [6].

Растр представляет из себя матрицу значений. Каждой ячейке матрицы соответствует прямоугольная пространственная область фиксированного размера, которая называется пикселем. Различают растры непрерывные и категориальные (классифицированные).

В отличие от векторных данных, которые требуют указания координат для каждой вершины, регулярно-ячеистый характер растровой модели позволяет вычислять координаты пикселей на основе их индексов. Поэтому фактически растровые данные хранятся в виде линейно упорядоченного списка значений (*raster values*) и описания геометрии растра (*raster geometry*).

Геометрия растра определяет, где именно располагаются в пространстве пиксели растра и может быть описана путем указания следующих компонент (см. табл. 1.3.1).

Иногда вместо параметров *XLLCENTER/YLLCENTER* указываются *XLLCORNER/YLLCORNER*, которые кодируют координаты левого нижнего угла, а не центра левой нижней ячейки растра. Если геометрия растра характеризуется анизотропией, то вместо одного значения *CELLSIZE* могут быть указаны разные размеры ячеек по осям координат *CELLSIZE<sub>X</sub>* и *CELLSIZE<sub>Y</sub>* [6].

**Компоненты растровых данных**

| Параметр  | Назначение                                     |
|-----------|------------------------------------------------|
| NCOLS     | Количество столбцов                            |
| NROWS     | Количество строк                               |
| XLLCENTER | Координата X центра левой нижней ячейки растра |
| YLLCENTER | Координата Y центра левой нижней ячейки растра |
| CELLSIZE  | Размер ячейки                                  |

Модель растровых данных состоит из строк и столбцов пикселей одинакового размера, соединенных между собой для формирования плоской поверхности. Эти пиксели используются в качестве строительных блоков для создания точек, линий, областей, сетей и поверхностей. Большинство доступных растровых данных ГИС построено на квадратном пикселе (рис. 1.3.3). Эти квадраты обычно преобразуются в прямоугольники различных измерений, если модель данных преобразуется из одной проекции в другую (например, из координат плоскости государства в координаты UTM [Universal Transverse Mercator]).

Местоположение отдельной ячейки определяется точкой геопривязки растра, разрешением ячейки, номером колонки и номером ряда ячейки на растре. Точкой геопривязки растра может быть нижний левый или верхний левый угол растра; это условие учитывается программным продуктом.

Расстояние между двумя ячейками растра есть функция местоположения их средних точек и разрешения ячеек. В растровой модели в ГИС определение длин вертикальных или горизонтальных линий проводится путем подсчета количества ячеек, по которым проходит линия, и умножением их на размер одной ячейки. Если линия ориентирована по диагона-

лям ячеек, то необходимо выполнить произведение количества ячеек на размер ячейки и на  $\sqrt{2}$ .



Рис. 1.3.3. Обычная растровая графика, используемая в приложениях ГИС: аэрофотоснимок (слева) и ЦМР США (справа)

Площадь выбранных объектов на растре вычисляется как произведение количества ячеек на площадь отдельной ячейки.

Для хранения растровых данных используют несколько подходов, основанных на различных методах сжатия. Методы сжатия растровых данных работают внутри подсистемы хранения и редактирования ГИС, но они могут вызываться и напрямую на этапе ввода информации в ГИС.

### 1.3.2. Векторные модели данных

Векторная модель пространственных данных включает описание координатных данных пространственных объектов и, опционально, топологических отношений между ними. Векторные данные фиксируют местоположение и форму объектов в виде геометрических примитивов, таких как точки, линии, полигоны, объемные тела. Выбор модели объекта (например, представить город точкой или полигоном) зависит от масштаба анализа

и целей исследования. Векторная модель данных является объектно-ориентированной.

Векторные модели данных используют точки и связанные с ними пары координат  $X$ ,  $Y$  для представления вершин пространственных объектов, как если бы они были нарисованы на карте вручную. Атрибуты данных этих функций затем сохраняются в отдельной системе управления базами данных. Пространственная информация и информация об атрибутах для этих моделей связаны посредством простого идентификационного номера, который присваивается каждому объекту на карте.

В географических информационных системах (ГИС) существуют три основных типа вектора: точки, линии и многоугольники. Точки – это нульмерные объекты, которые содержат только одну пару координат. Точки обычно используются для моделирования отдельных объектов, таких как здания, колодцы, опоры линий электропередач, места отбора проб и т. д. Точки имеют только свойство местоположения. Другие типы точечных объектов включают в себя узел и вершину. В частности, точка – это автономный объект, а узел – это топологический узел, представляющий общую пару координат  $X$ ,  $Y$  между пересекающимися линиями и/или многоугольниками. Вершины определяются как каждый изгиб вдоль линии или многоугольника, который не является пересечением линий или многоугольников.

Линии – это одномерные элементы, состоящие из нескольких явно связанных точек. Линии используются для представления линейных объектов, таких как дороги, ручьи, разломы, границы и т. д. Линии имеют свойство длины. Линии, которые напрямую соединяют два узла, иногда называют цепями, ребрами, сегментами или дугами.

Полигоны – это двумерные объекты, созданные несколькими линиями, которые возвращаются назад для создания «закрытого» объекта. В случае многоугольников первая пара координат (точка) на первом отрез-

ке линии совпадает с последней парой координат на последнем отрезке линии. Полигоны используются для представления таких объектов, как границы города, геологические формации, озера, ассоциации почв, растительные сообщества и т. д. Полигоны имеют свойства площади и периметра. Полигоны также называют областями.

### ***Векторные данные***

Simple Features (официально Simple Features Access) – это стандарт OGC 06-103, разработанный Open Geospatial Consortium (OGC) и реализованный также в виде международного стандарта ISO 19125, который определяет общую модель хранения и доступа к векторным объектам (точка, линия, многоугольник, мультиточечные объекты, мультилинии и т. д.), в географических информационных системах [5].

Геометрическое представление пространственных объектов базируется на следующих принципах:

- все геометрии состоят из точек;
- точки являются координатами в 2-, 3- или 4-мерном пространстве;
- все точки в геометрии имеют одинаковую размерность.

В дополнение к координатам  $X$  и  $Y$  имеются два дополнительных параметра:

- координата  $Z$ , обозначающая высоту;
- координата  $M$ , обозначающая некоторую меру, связанную с точкой, а не с признаком в целом.

Всего стандарт Simple Features включает в себя 17 типов геометрий. Из них наиболее употребительными являются следующие 7 (см. табл. 1.3.2) [6].

## Типы векторной геометрии

| Тип                | Описание                                                                                                                                                                                                                                         |
|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| POINT              | Нульмерная геометрия, содержащая одну точку                                                                                                                                                                                                      |
| LINESTRING         | Последовательность точек, соединенных прямыми, несамопересекающимися отрезками; одномерная геометрия                                                                                                                                             |
| POLYGON            | Геометрия с положительной площадью (двумерная); последовательность точек, отрезки между которыми формируют замкнутое кольцо без самопересечений; первое кольцо является внешним, ноль и более остальных колец представляют дырки внутри полигона |
| MULTIPOINT         | Множество точек; геометрия типа MULTIPOINT называется простой, если ни одна пара точек в MULTIPOINT не совпадает                                                                                                                                 |
| MULTILINESTRING    | Множество линий                                                                                                                                                                                                                                  |
| MULTIPOLYGON       | Множество полигонов                                                                                                                                                                                                                              |
| GEOMETRYCOLLECTION | Множество геометрий произвольного типа за исключением GEOMETRYCOLLECTION                                                                                                                                                                         |

Примеры различных видов геометрий представлены на рис. 1.3.4.

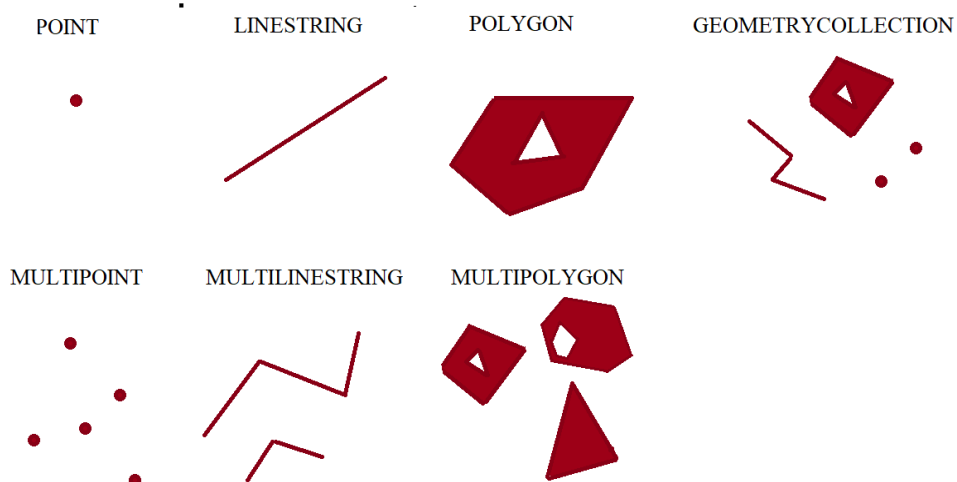


Рис. 1.3.4. Виды геометрий

Дискретные объекты, каждый из которых может занимать в любой момент времени только одну определенную точку в пространстве, представляются в виде **точек** (рис. 1.3.5). Считается, что такие объекты не имеют пространственной протяженности и ширины (внемасштабные, 0-мерные пространственные объекты) и каждый из них может быть обозначен парой координат:  $X$  и  $Y$  (широта и долгота). Примерами могут служить мысы, отдельно стоящие деревья, маяки, отметки высот и т. д. Совокупность точечных объектов образует **точечный слой** (дополнительно в описание точечного объекта может быть включена координата  $Z$ . Такие объекты часто называются **2.5D-объектами**, так как они описывают только высоту или только глубину точки, но не оба параметра одновременно).

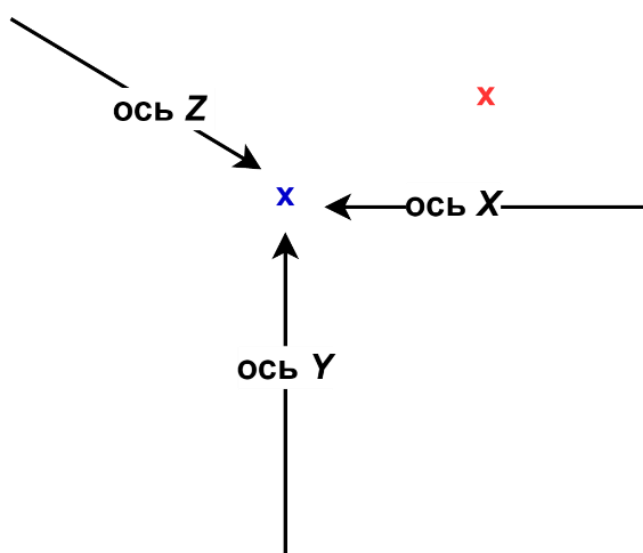


Рис. 1.3.5. Точечный объект, описываемый координатами  $X$ ,  $Y$  и (возможно)  $Z$  (атрибуты описывают каждую точку)

При выборе точек для представления объектов необходимо руководствоваться масштабом карты (как мелко показаны объекты), удобством (проще и быстрее поставить точку, нежели нарисовать полигон) и типом объектов (такие объекты, как телефонные столбы, не имеет смысла представлять в виде полигонов даже в крупном масштабе).

Объекты, немасштабные по ширине, но имеющие протяженность, отображаются в виде **линий** (1-мерный пространственный объект) (рис. 1.3.6). Это могут быть дороги, линии электропередач, реки, тектонические разломы и другие. В ГИС они представлены последовательностью пар координат. Сложные наборы линий называются **сетями**. Они содержат дополнительную информацию о пространственных взаимоотношениях этих линий. Например, линии, отражающие транспортные магистрали, могут содержать сведения об особенностях передвижения по ним (направления, скорость и т. д.), при этом информация присваивается каждому отрезку до изменения атрибутов. Точки, в которых отрезки связываются, называются **узлами**, также кодируются; каждый узел свидетельствует о смене ситуации. Все эти атрибуты определяются по всей сети без пропусков, что необходимо для обеспечения связанности и пространственных отношений. Совокупность линий образует **линейный слой**.

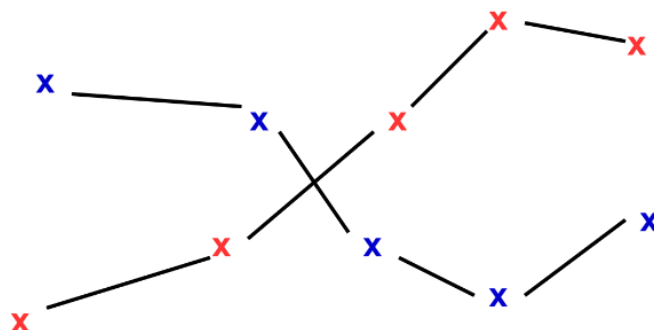


Рис. 1.3.6. Полилиния

Если точечный объект состоит из одной вершины, то **полилиния имеет две и более вершины**. Полилинии используются для хранения геометрии линейных объектов, таких как дороги, реки, изолинии, маршруты и др. Иногда в дополнение к основной геометрии для полилиний устанавливаются специальные правила. Например, горизонтали высот могут касаться друг друга (в случае отвесного склона), но никогда не должны пере-

секаться, а полилинии, используемые для хранения данных о дорожной сети, должны быть связаны в местах перекрестков. В некоторых ГИС-приложениях вы можете устанавливать набор подобных правил для определенных типов объектов (т. е. дорог), и программа будет проверять полилинии на соответствие этим правилам.

**Полигоны** используются для тех объектов, которые на карте сохраняют свои очертания (имеют длину и ширину) (2-мерный пространственный объект) (рис. 1.3.7). Отображаются в виде замкнутой последовательности пар координат, где начальная и конечная точки имеют одинаковое значение. Совокупность полигонов образует **полигональный слой**.

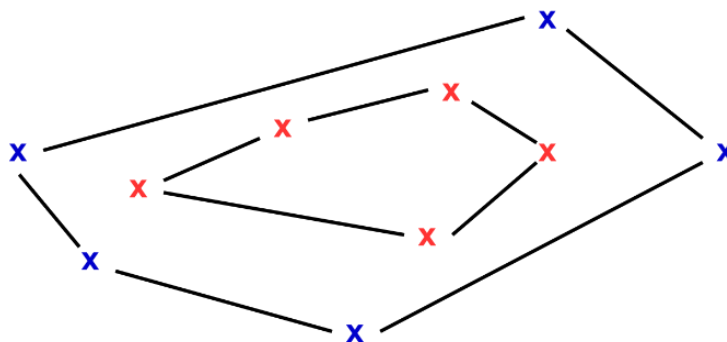


Рис. 1.3.7. Полигон

Поверхности выражены обычно высотами точек либо контурами рельефа. Модель поверхности обычно образована через последовательность точек, распределенных с различной регулярностью, причем каждая точка отражает определенное значение высоты. Через три ближайшие точки можно провести плоскость в виде треугольника, внутри которого будет зафиксирован постоянный уклон. Совокупность таких треугольников будет представлять модель поверхности, напоминающую кристалл. Создаваемая таким образом модель получила название TIN-нерегулярная триангуляционная сеть (Triangulated Irregular Network) (рис. 1.3.8).

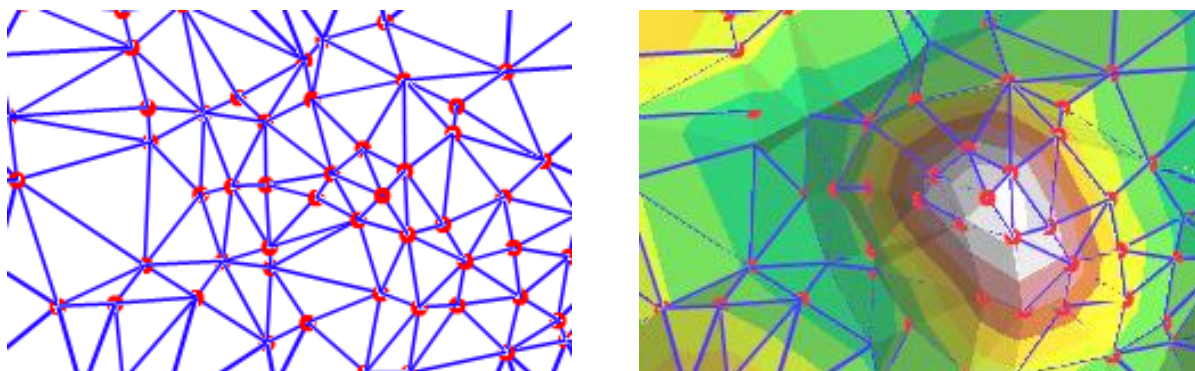


Рис. 1.3.8. Модель данных TIN

Современные ГИС поддерживают одновременно множество моделей, однако векторные получили большую популярность, поскольку обеспечивают хорошую визуализацию картографируемых объектов, возможность детально представлять реальные объекты и более оперативную обработку информации в процессе анализа и геообработки.

#### 1.4. Задачи анализа пространственных данных

Анализ пространственных данных представляет собой процесс выявления закономерностей пространственного распределения объектов и пространственных взаимоотношений между объектами исследования с использованием средств геоинформационных систем [7]. Выделяют следующие задачи анализа пространственных данных.

1. *Анализ местоположения.* Исследование пространственного распределения объектов необходимо для понимания закономерностей распределения данных и причин появления тех или иных свойств. Для этого нужно определенным образом отобразить объекты, которые исследуются, опираясь на значения их характеристик. Например, для экологического анализа важно влияние особенностей рельефа, расположения водоемов, рекреационных зон и других факторов на микроклимат и загрязнение воздуха и прочее; для развития городских территорий важно анализировать соответствие вновь создаваемых объектов с генеральным планом города,

планом существующей застройки и планом инженерной инфраструктуры; для служб оперативного реагирования важно анализировать динамику появления событий, связанных с безопасностью, и изучать связи с их местоположением.

2. *Удовлетворение пространственных условий.* Этой категории соответствуют более сложные запросы, связанные с необходимостью выполнения операций пространственного анализа. Примером таких запросов может быть выявление территорий, обладающих определенными свойствами; обоснование местоположения строительства нового объекта; определение оптимальных условий для размещения объектов инфраструктуры и прочее.

3. *Временной анализ.* Целью данной категории является выявление изменений, которые происходят в пространстве и времени, и изучение тенденций этих изменений на определенной территории. Например, каковы особенности изменения территории города, как растет или снижается посещаемость каких-либо городских пространств или объектов, как меняются транспортные потоки и прочее.

4. *Выявление структуры.* Эта категория связана с различными статистическими исследованиями, привязанными к определенной местности. Для реализации подобных исследований требуются более сложные средства пространственного анализа, интегрированные с методами статистического моделирования и машинного обучения. Применяя данные технологии, можно отвечать на вопросы, связанные с выявлением общих закономерностей развития города и отдельных его систем (определение наиболее опасных участков дорог и проблем организации дорожного движения, исследование демографических распределений, стоимости недвижимости, инфраструктурной обеспеченности и прочее).

5. *Оценка различных сценариев.* Построение прогностических моделей является наиболее эффективным подходом для поддержки принятия

решений в различных задачах пространственного развития. С помощью таких моделей прежде всего исследуется потенциал территории и выявляются наиболее критичные ситуации, которые могут привести к опасным ситуациям, связанными с реализацией техногенных или природных угроз. Например, что произойдет, если интенсивность дождя будет критической? Как изменится транспортное сообщение, если изменить схему движения общественного транспорта?

Базовые операции пространственного анализа:

1. Представление географического распределения данных, то есть отображение данных о пространственно-распределенных объектах на цифровой географической карте с учетом пространственной информации об объектах исследования.

2. Построение и выполнение запроса или получение выборки данных из базы географических данных. Запросы позволяют найти и рассмотреть определенные объекты. Существует два вида запросов в ГИС: атрибутивные и пространственные запросы. Атрибутивные запросы, называемые также непространственными, ведут поиск объектов на основании значений их атрибутов. Запросы местоположения, называемые также пространственными запросами, ведут поиск объектов по пространственным характеристикам.

3. Поиск ближайших объектов к заданному и создание буферной зоны вокруг этого объекта.

4. Наложение различных пространственных (тематических) слоев объектов. Путем наложения одного набора объектов на другой можно получить новую информацию о закономерностях пространственного размещения объектов.

При проведении пространственного анализа результаты одной операции можно использовать в качестве исходных данных для другой, тем

самым обеспечивая проведение сложного анализа, основанного на сочетании различных операций [7].

### ***Методология геопространственного анализа [2, 7]***

#### **1. Постановка проблемы**

Формулирование проблемы всегда начинается с определения цели анализа в виде вопросов, на которые необходимо получить ответы, например: сколько леса растет в пределах каждого бассейна водосбора? На этом этапе формируется соответствие между масштабом возникшей проблемы и средствами, необходимыми для ее решения, а также устанавливаются критерии, которые определяют параметры использования базы геоданных для получения ответов.

#### **2. Оценка исходных данных**

Тип данных и объектов, доступных для проведения данного исследования, в значительной мере определяет специфику метода, который будет использован, и достижимую точность результата. С другой стороны, чтобы получить качественную информацию, нужно обеспечить соответствующий уровень исходных данных.

#### **3. Выбор методов анализа**

Современные ГИС имеют большое количество аналитических средств пространственных и атрибутивных данных. Пространственный анализ определенного набора данных может включать, например, операции извлечения объектов, построения буферных зон, наложения буферных зон на другие слои, работу с объектами, попавшими в буферные зоны, и другие операции. Набор средств анализа определяется при интерпретации критериев анализа, выделенных на первом этапе. Каждое утверждение в постановке проблемы может транслироваться в ряд операций анализа.

#### 4. Обработка данных

В большинстве случаев анализа требуется набор операций с множеством слоев. При работе с векторными наборами данных они выполняются ступенчатым образом – два входных слоя используются для формирования нового слоя; этот промежуточный слой обрабатывается совместно с третьим слоем, чтобы сформировать другой промежуточный слой, и так далее до достижения желательного результирующего слоя карты. При работе с растровыми наборами данных есть возможность одновременной обработки нескольких слоев, алгоритм которой реализуется в растровом калькуляторе (Raster Calculator).

#### 5. Оценка и отображение результатов

В процессе оценки результатов выполняется их интерпретация, определяется объективность и достаточность полученной информации. Если необходимо, то принимается решение о повторении анализа с другими параметрами, или уточнение анализа, или применение другого метода. ГИС позволяет сравнительно легко и оперативно сделать необходимые изменения и получить новый результат. Можно также оперативно сравнить результаты разных анализов и увидеть, который из подходов оказался лучшим. Результаты анализа могут быть представлены в виде карты, диаграммы, значений в таблице – фактически новой информации. Необходимо решить, какую информацию выносить на карту, как группировать значения для наилучшего отображения данных.

### 1.5. Картографические сервисы и пространственные данные

*Google Maps* – картографическая веб-платформа и приложение для конечного пользователя, предлагаемое Google. Он предлагает спутниковые снимки, аэрофотосъемку, карты улиц, интерактивные панорамные виды улиц на 360° (Street View), условия движения в реальном времени и плани-

рование маршрутов для путешествий пешком, на машине, по воздуху (в бета-версии) и на общественном транспорте (рис. 1.5.1).

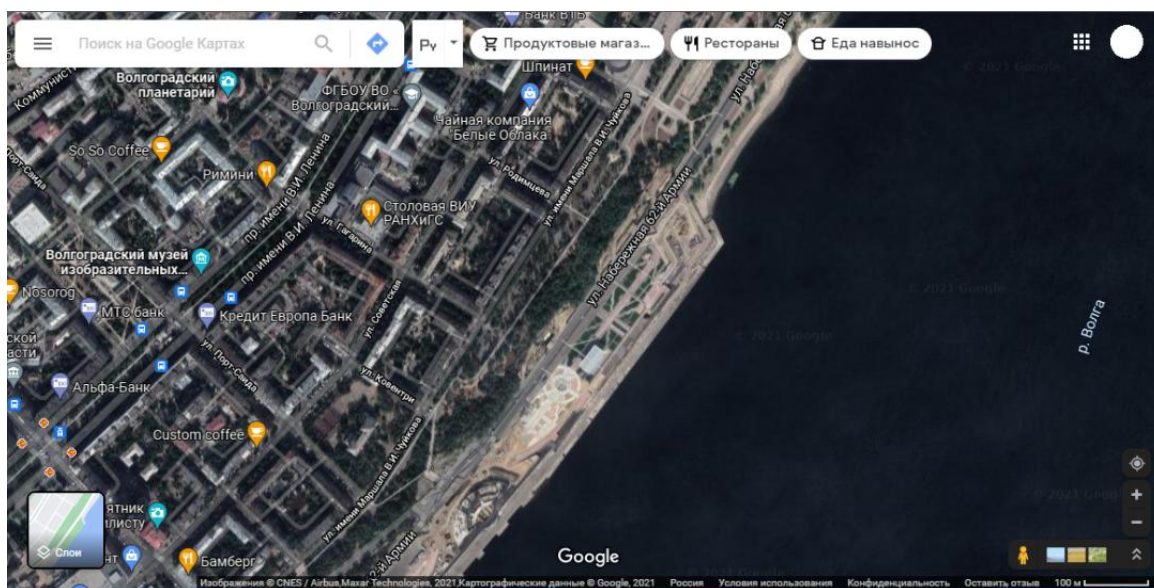


Рис. 1.5.1. Карта Google Maps

*Microsoft Bing Maps* – картографический сервис Microsoft, часть портала Bing (см. рис. 1.5.2).

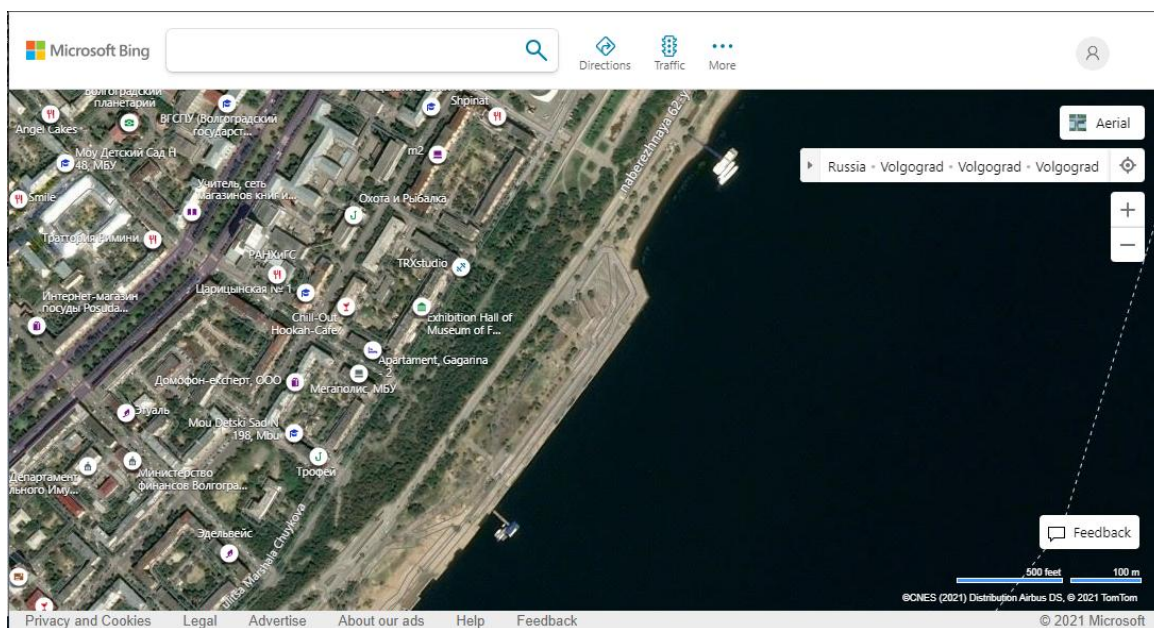


Рис. 1.5.2. Карта Microsoft Bing Maps

*Яндекс.Карты* – российский картографический веб-сервис, разработанный Яндексом. Сервис предоставляет подробные карты всего мира. Включает в себя поиск, информацию о пробках, маршрутах и панорамах улиц. В мобильные приложения также включена возможность просмотра общественного транспорта в режиме реального времени и встроенного навигатора [8] (см. рис. 1.5.3).

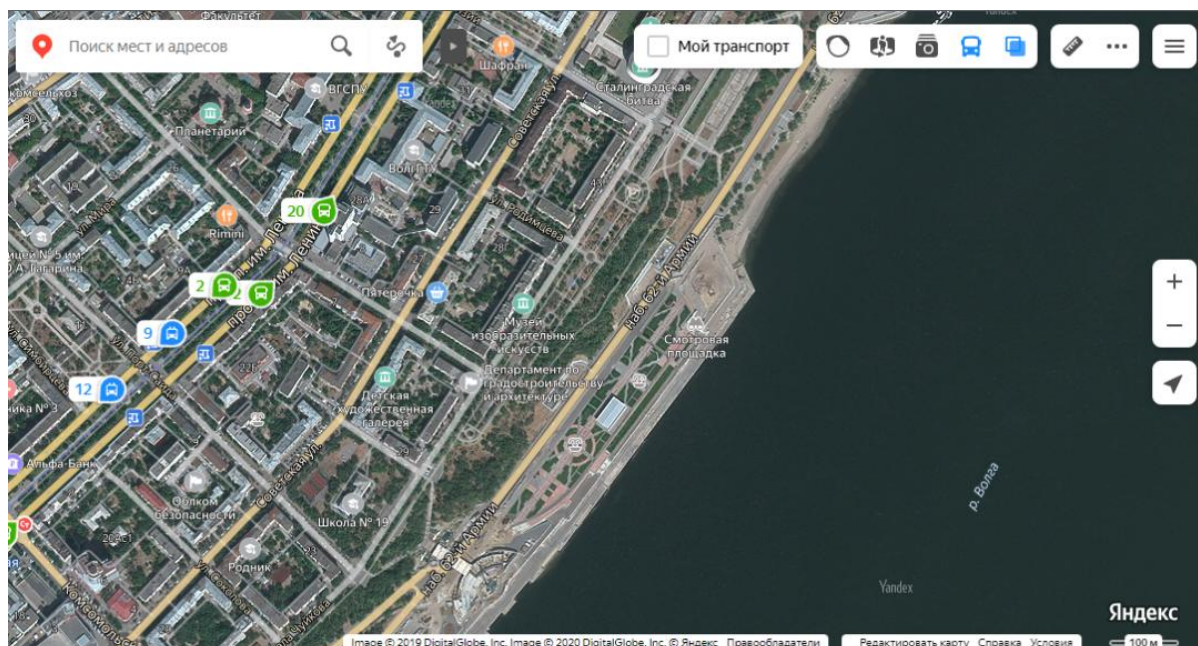


Рис. 1.5.3. Карта Яндекс.Карты

*Apple Maps* – картографический сервис от компании Apple для операционных систем iOS и macOS. Карты от Apple по точности данных различных городов проиграют всем остальным конкурентам [9] (см. рис. 1.5.4).

*2ГИС* – российская местная поисковая компания, которая разрабатывает цифровые карты и путеводители по городам России, Казахстана, Италии, Чехии, Чили, ОАЭ, Узбекистана, Кыргызстана, Кипра, Азербайджана и Украины. Карты, в первую очередь, нацелены на качественную работу при отсутствии или плохом интернет-соединении [8] (см. рис. 1.5.5).

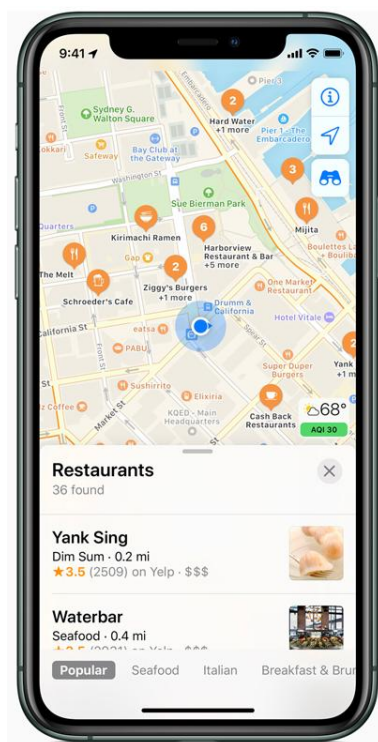


Рис. 1.5.4. Карта Apple Maps [9]

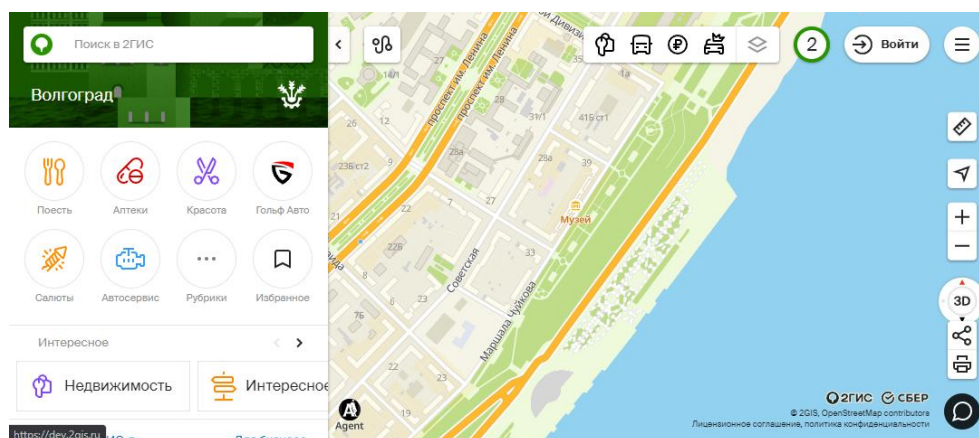


Рис. 1.5.5. Карта 2ГИС

*OpenStreetMap* – совместный веб-картографический проект по созданию бесплатной редактируемой географической базы данных мира. Карты очень детализированы и во многих регионах обходят всех своих конкурентов. Именно на базе OpenStreetMap работают различные сервисы, такие как OsmAnd, MAPS.ME и многие другие (рис. 1.5.6).

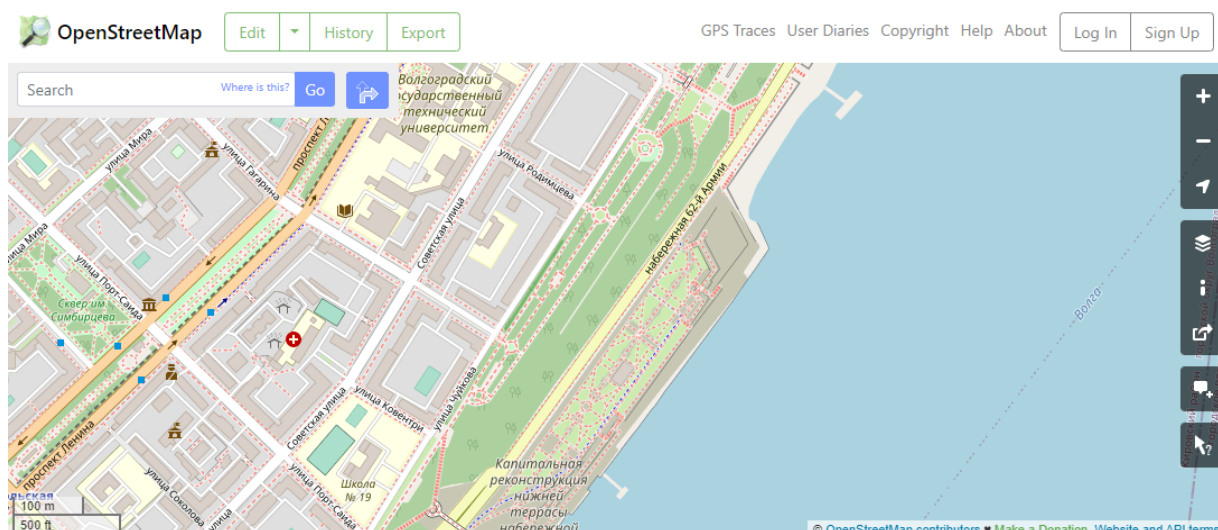


Рис. 1.5.6. Карта OpenStreetMap

*Wikimapia* – проект географической онлайн-энциклопедии. Проект реализует интерактивную веб-карту, которая использует Карты Google с географически привязанной вики-системой, с целью отметить и описать все географические объекты в мире (рис. 1.5.7).

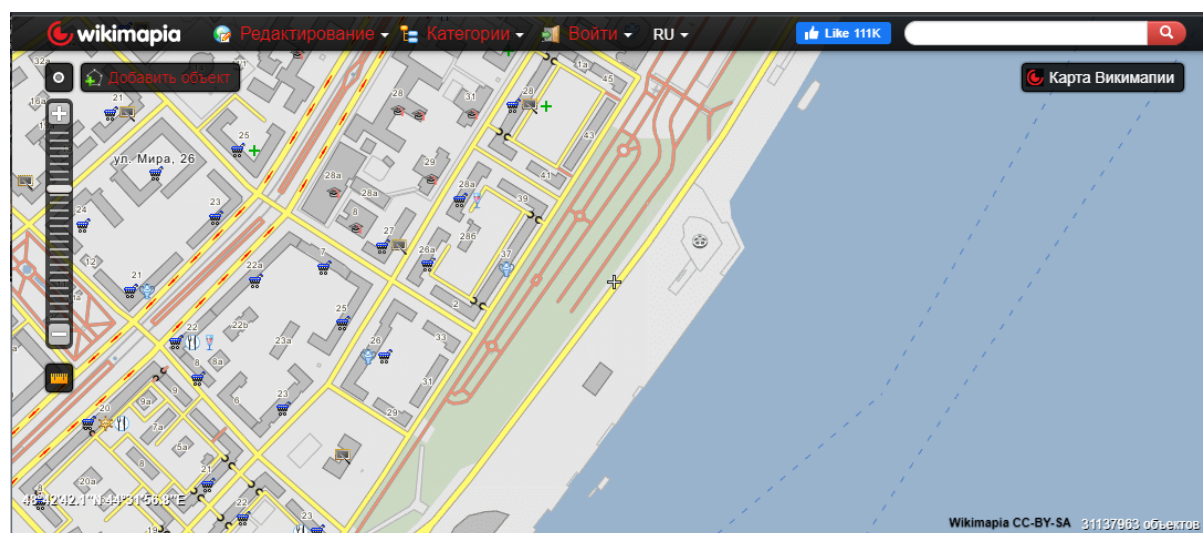


Рис. 1.5.7. Карта Wikimapia

*Qwant Maps* – картографический сервис от Qwant. Относительно недавно Qwant закрыл свой поисковик и карты для ряда регионов (рис. 1.5.8).

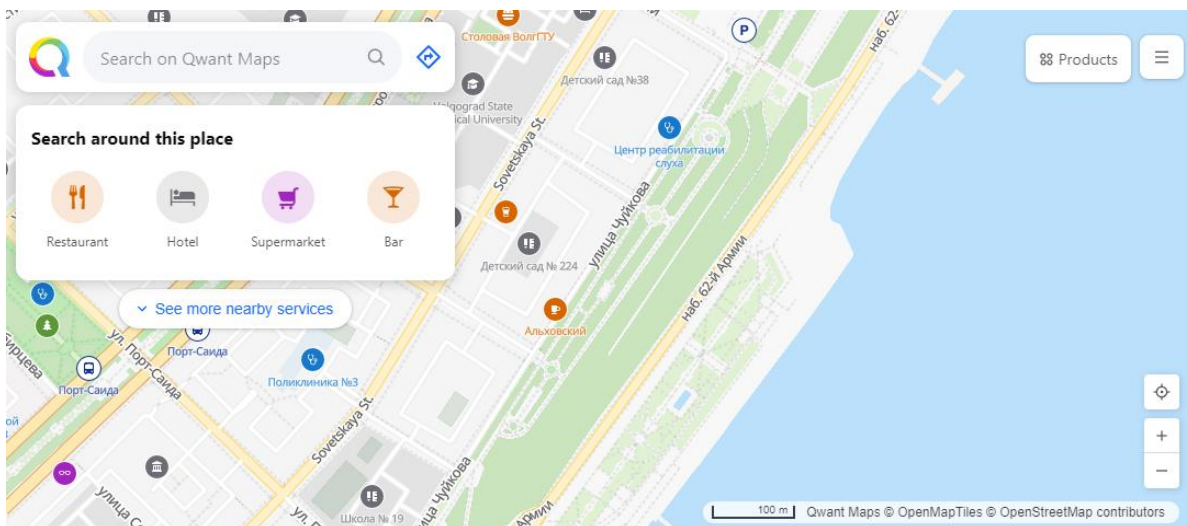


Рис. 1.5.8. Карта Qwant Maps

*HERE WeGo* – веб-служба картографии и навигации, управляемая Here Technologies. Первоначально носившие имена «smart2go», «Ovi Maps», «Nokia Maps» и «HERE Maps» разрабатывались Nokia и долгое время предустанавливались по умолчанию в устройства от Nokia. Учитывая прошлое, карты очень хорошо развиваются с уклоном в автомобильную навигацию. В ряде городов Европы детализированы даже пешеходные дорожки и дороги, которые отсутствуют на других картах [8] (см. рис. 1.5.9).

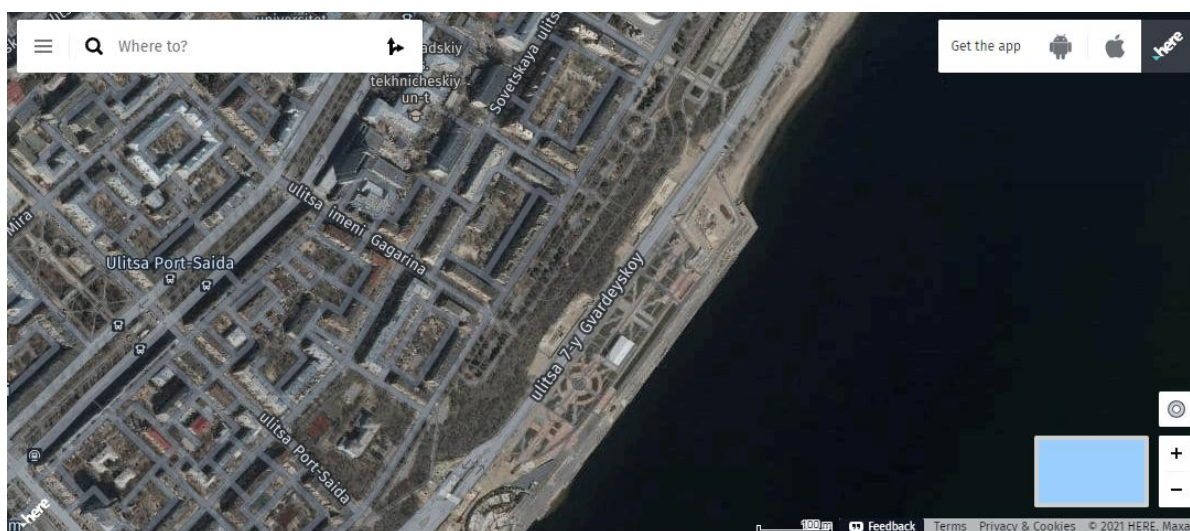


Рис. 1.5.9. Карта HERE WeGo

## 1.6. Управление данными в ГИС (на примере OpenStreetMap)

Базовым элементом структуры данных OSM является точка (node) с географическими координатами – широтой (latitude) и долготой (longitude). Точка может быть отдельным объектом или входить в состав других объектов (линий, полигонов, мультиполигонов).

Отношение – это логическое объединение точек, линий и других отношений в единый объект. Каждый объект имеет набор тегов, описывающих его. Тег (tag) определен как  $k = \text{«ключ»}$   $v = \text{«значение»}$ . Если элемент не имеет тегов, то он не является объектом, а входит в состав других объектов, как и некоторые элементы с тегами.

В целом всю структуру данных OSM можно представить схематично (рис. 1.6.1).

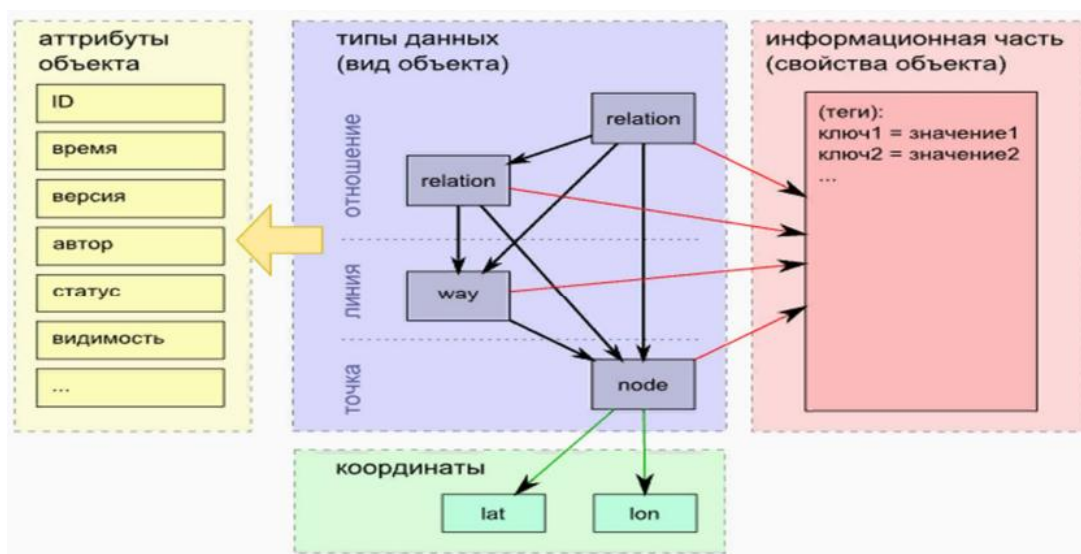


Рис. 1.6.1. Структура данных OpenStreetMap [10]

Все данные можно разбить на три группы:

1) типы данных (вид объекта) – основная группа, описывающая геометрические типы объектов на карте и их иерархию по отношению друг к другу, где основная единица точка (node) обязательно включает в себя координаты (lat, lon) и может входить во все остальные типы объектов;

2) атрибуты объектов – группа, описывающая формальные данные об авторах, времени добавления/правки, статусе, видимости; организует структуру типов объектов за счет присваивания к ним ID;

3) информационная часть (свойства объектов) – описательная группа, формирующая представление об объекте и его свойствах путем добавления определенных тегов к объекту.

Существует ряд решений для парсинга данных из osm-файла. Например, библиотека «BasicOSMParser» (доступна на GitHub в репозитории аккаунта PanierAvide/BasicOSMParser) – это набор классов Java, позволяющий анализировать необработанные XML-файлы OSM. Также библиотека позволяет экспортировать данные в виде файлов CSV. Она использует Java по умолчанию SAX parser (org.xml.sax). Тесты приложения используют платформу JUnit 4 framework.

Библиотека «osm4routing» (доступна на GitHub в репозитории аккаунта Tristramg/osm4routing) – инструмент, предоставляющий парсер данных из OpenStreetMap, чтобы превратить их в узлы-ребра. Входными данными является XML-файл OpenStreetMap. Выходными данными библиотеки могут быть CSV-файл или база данных (postgres, mysql, sqlite, postgis).

Библиотека «Libosmium» (доступна на GitHub в репозитории аккаунта osmcode/libosmium) – быстрая и гибкая библиотека C++ для работы с данными OpenStreetMap. Библиотека может прочитать OSM XML-данные или в двоичном формате (PBF) и может вызывать различные обработчики для каждого OSM-объекта. Libosmium доступен под лицензией на программное обеспечение Boost.

Информационным обеспечением основной программы и других модулей является файл формата OSM XML (экспортированные данные участка карты OpenStreetMap). В основном файле содержится список экземп-

ляров примитивных данных, таких как узлы, пути и отношения («nodes», «ways» и «relations»), а также их описание и связи.

Для удобной работы с данными файла часто обращаются к библиотеке NetTopologySuite и приводят данные из файла к структуре (точек, линий и отношений).

Для преобразования данных из объектов типа Geometry в GeoJSON используется GeoJsonWriter – средство библиотеки NetTopologySuite.

Соотношение структур полученных объектов из файла OSM XML и приведенных к структуре Geometry при помощи библиотеки NetTopologySuite представлено в табл. 1.6.1.

*Таблица 1.6.1*

**Соотношение структур объектов**

| Объект OSM XML               | Наследуемый объект<br>NetTopologySuite.Geometries |
|------------------------------|---------------------------------------------------|
| Точка (node)                 | Point                                             |
| Линия (way)                  | LineString                                        |
| Замкнутая линия (closed way) | LinearRing                                        |
| Отношение (relation)         | GeometryCollection                                |

Для каждого из типов данных необходимо хранить его теги, описывающие объект карты и его свойства. Для типа «relation» помимо тегов необходимо хранить еще и роль каждого из объектов.

Структура данных, которая должна быть получена после преобразования и использоваться в дальнейшем другими модулями и базовыми компонентами платформы, представлена на рис. 1.6.2.

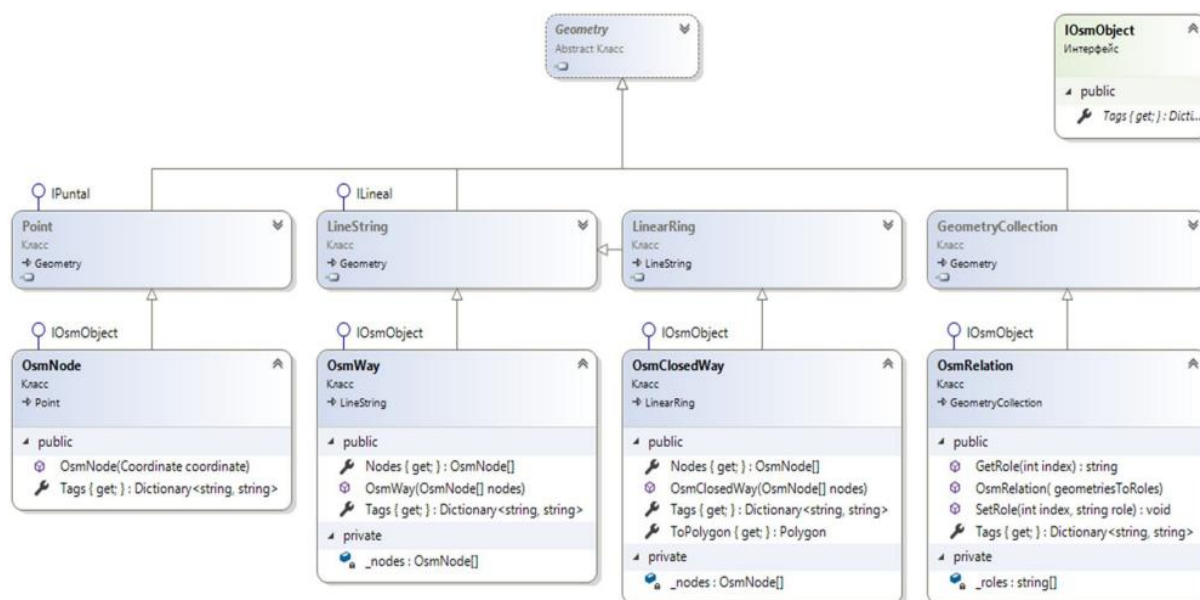


Рис. 1.6.2. Диаграмма классов OSM

## 1.7. Технологические решения на основе пространственных данных

ГИС-системы используются не только для отображения и визуализации пространственных данных, но и для различных аналитических задач, моделирования и поддержки управления.

Основными направлениями использования ГИС-технологий являются строительное, архитектурное и инженерное проектирование; географическая аналитика, госуправление и бизнес, связанный с развитием пространственно-распределенных систем.

Новые информационные технологии – искусственный интеллект и машинное обучение, трехмерное моделирование, автоматизация картографирования, облачные вычисления и прочие – расширяют возможности применения пространственных данных, приводят к росту запросов на сервисы и продукты на основе ГИС. Фактически появляется запрос на цифровизацию пространства, формирование «цифровых двойников» пространства, меняющихся во времени [11]. Внедрение сервисов на основе пространственных данных в реальные бизнес-процессы повышает эффективность

принятия решений на стратегическом и оперативном уровнях и позволяет оптимизировать ресурсы при управлении развитием территорий.

Рассмотрим примеры применения ГИС-технологий для различных прикладных задач.

В работе [12] для моделирования режимов работы и состояния водохранилища ГЭС использовались данные дистанционного зондирования Земли (ДЗЗ), а именно космические снимки, полученные с искусственных спутников Земли, прошедшие обработку и представляющие из себя растровое изображение поверхности Земли. Данные ДЗЗ доступны для любой территории и, соответственно, для любого существующего или проектируемого объекта и, что особенно важно, находятся в свободном доступе. Для решения этой задачи использовались данные о геометрических параметрах рельефа (уклон, экспозиция, кривизна и прочее) и объектов гидрографии (площадь водной поверхности и ширина створа). Эти параметры являются базовыми. Объединив их, можно получить дополнительную систему гидрологических параметров (водотоки, направления стока и точки кумулятивного стока, водосборные бассейны), которые также могут быть использованы при построении имитационной модели каскада ГЭС. Кроме этого, для исследования использовались данные о температуре воздуха (данные метеорологических постов) и температуре земной поверхности (определяется по спутниковым снимкам на основе термальных спектральных каналов), данные об интенсивности осадков на основе микроволновой радиометрии, количественные показатели снежного покрова территорий (площадь снежного покрова и его толщина).

Имитационное моделирование на основе данных ГИС-мониторинга позволяет увеличить точность расчета конструктивных параметров ГЭС и емкости водохранилища. Во время эксплуатации ГЭС применение базы данных ДЗЗ позволит увеличить точность прогнозирования естественного

притока к створу ГЭС, а также потерь, обусловленных природными факторами (рис. 1.7.1) [12].

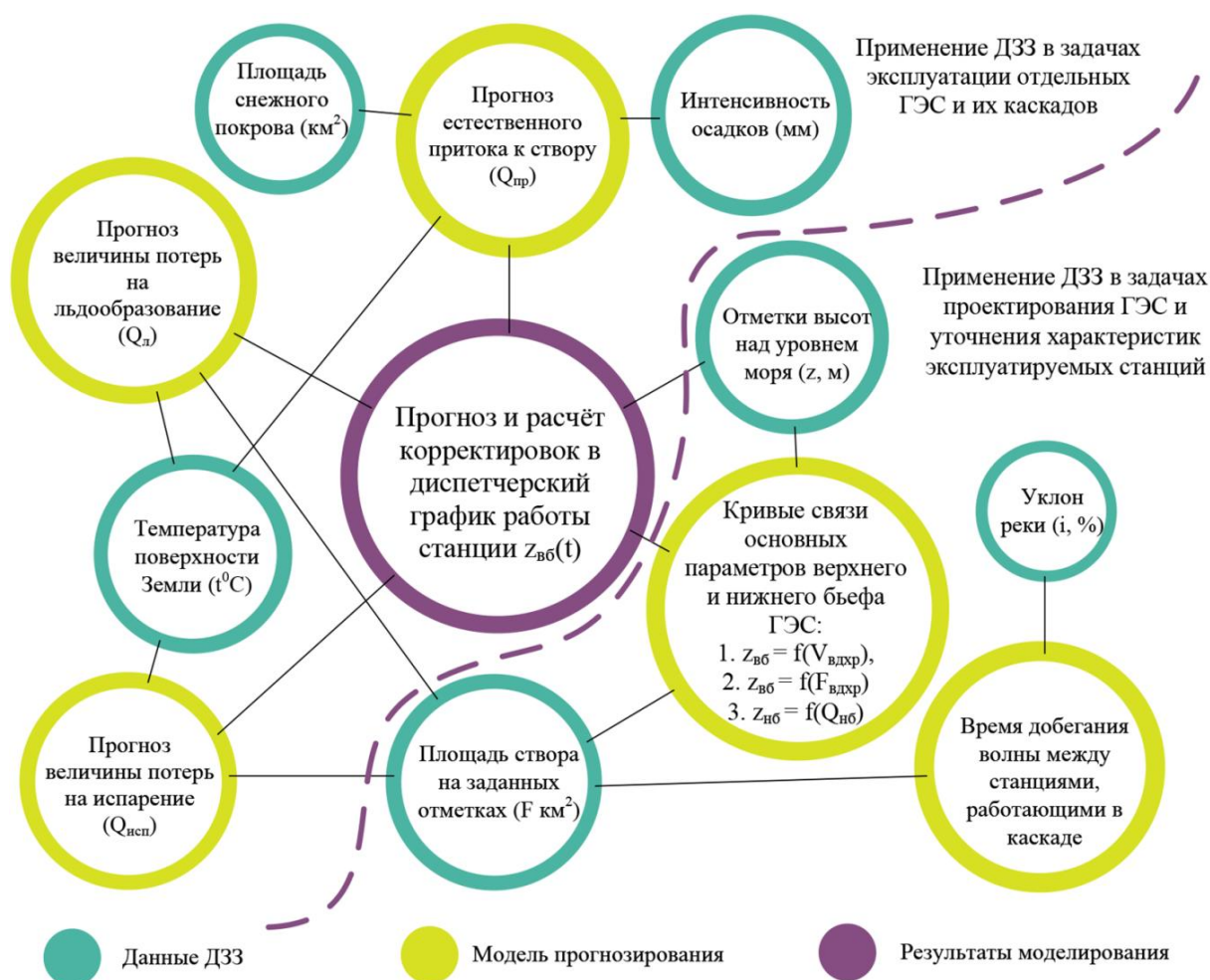


Рис. 1.7.1. Задачи анализа эксплуатации ГЭС на основе данных ДЗЗ [12]

Одним из наиболее перспективных направлений использования моделирования и пространственных данных является исследование транспортных систем и задач логистики.

Например, на основе мультиагентных моделей могут быть реализованы имитационные эксперименты, с помощью которых можно изучать изменение пассажирских потоков, планировать стоимость перевозок, оптимизировать маршрутную сеть и логистические операции [13–15].

Пространственные данные и ГИС-технологии используются для создания рекомендательных систем в сфере управления развитием городских территорий, становятся движущей силой инноваций и кардинально меняют принципы управления городом. С 2015 года Правительство Москвы потратило более 500 млн. рублей на закупку у мобильных операторов геоданных о передвижениях людей. Эти данные используются для оптимизации обеспечения процессов жизнедеятельности и улучшения городской среды [16].

Для решения задач контроля экологической безопасности, автоматизации процедур кадастровой оценки, анализа вариантов использования территорий применяются методы ДЗЗ и методы машинного обучения. Например, в проектах [17, 18] представлены решения для сбора информации о состоянии зеленых насаждений города, контроля уровня загрязнения окружающей среды. В статье [19] авторы предлагают внедрение системы поддержки принятия пространственных решений (SDSS), которая объединяет ГИС с аналитическими моделями для оценки различных типов землепользования.

Платформы, объединяющие наборы инструментов анализа данных о климате и состоянии окружающей среды, позволяют моделировать состояние воздуха на ближайшие сутки на основе данных с датчиков с применением методов машинного обучения и прогнозирования [20].

В работах [21, 22] рассматривается задача анализа обеспеченности территории города зелеными насаждениями. Для этого предлагается использовать методы классификации участков территории на основе методов распознавания актуальных спутниковых снимков с использованием нейронной сети. Метод представлен следующими этапами:

1. Получение снимков исследуемой территории.
2. Подготовка изображения к анализу:
  - а) определение зоны распознавания и привязка к масштабу карты. Снимок отображает квадратную территорию со стороной 850–1020 метров.

Один метр соответствует 2,5–3 пикселям для любых, распространяемых по ссылкам, онлайн-карт;

б) масштаб подбирается по соотношению видимой пользователем области и различимому человеческим глазом составу объектов территории с учетом разницы проекций снимков городов России, расположенных на всех широтах с севера на юг;

в) формирование изображения размером 2550\*2550 пикселей.

### 3. Обучение нейронной сети:

а) подбор размера классифицируемого фрагмента общего изображения;

б) формирование обучающей выборки, включающей группы изображений с интересующими объектами, среди которых одна или несколько групп должны включать участки территорий, имеющих зеленые насаждения разных типов;

в) обучение классификатора.

### 4. Классификация снимка исследуемой территории:

а) использование нейронной сети, обученной на распознавание нескольких классов (например, «Деревья», «Трава» и «Антропогенные объекты»);

б) создание полупрозрачной цветовой маски для фрагментов снимка, включающих найденные объекты, относящиеся к интересующему классу (например, «Деревья»);

### 5. Формирование зоны покрытия:

а) создание контура для полученной маски исследуемого класса;

б) привязка контура к координатам на цифровой карте.

## **Вопросы**

1. Понятие геоинформационной системы (ГИС).
2. Базовые возможности и структура ГИС.
3. Функции ГИС.
4. Направления научных знаний, необходимые для развития ГИС.
5. Геопространственные данные. Примеры. Определение.
6. Технологии сбора и анализа пространственных данных.
7. Источники данных для ГИС.
8. Технологии сбора пространственных данных.
9. Организация данных в ГИС.
10. Модели пространственных данных.
11. Особенности существующих картографических сервисов.
12. Картографические сервисы как источники данных для анализа.
13. Растровая модель представления пространственных данных в ГИС.
14. Векторная модель представления пространственных данных в ГИС.
15. Методы анализа пространственных данных в ГИС.
16. Методы распознавания образов в ГИС.
17. Визуализация в геоинформатике.
18. Структура данных в OpenStreetMap.
19. Примеры применения методов машинного обучения для анализа пространственных данных.

## ЗАДАНИЯ ДЛЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

### 2.1. Модели пространственных данных

1. Используя карту, изображенную на рис. 2.1.1, определите различные типы векторных данных и выделите их на карте.
2. Отметьте локацию на карте (в любом картографическом сервисе) и определите, какие векторные объекты могут быть выделены и какой тип геометрии (точка, линия и полигон) подходит для каждого из выделенных объектов.
3. Заполните табл. 2.1.1.

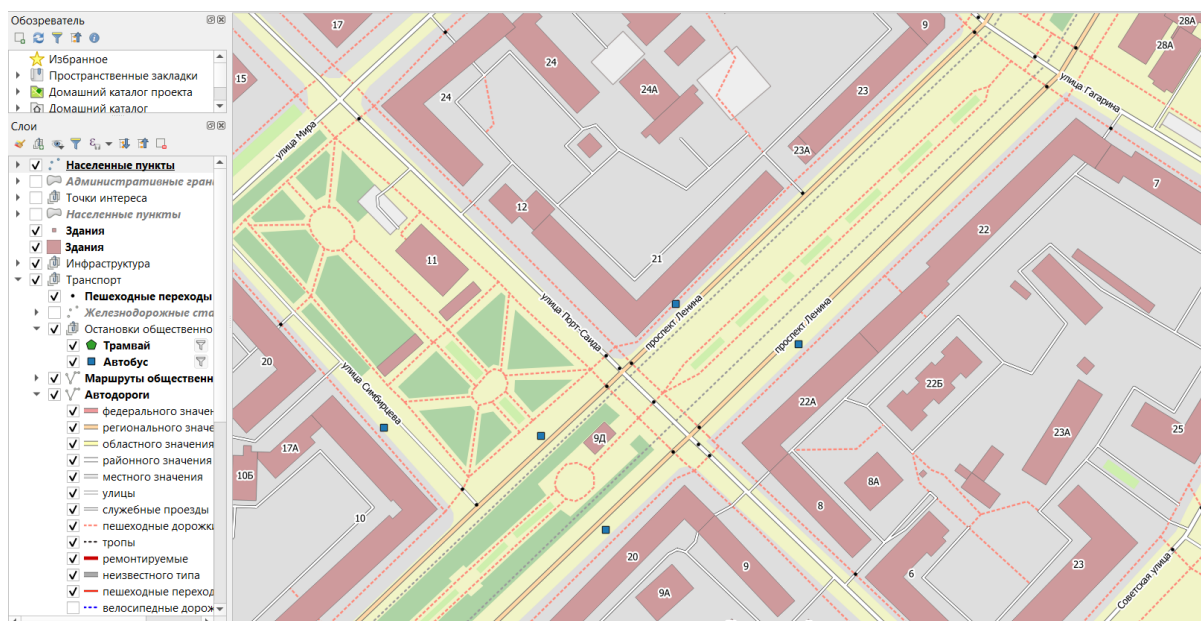


Рис. 2.1.1. Фрагмент карты для определения типов векторных данных

Таблица 2.1.1

**Структура заполнения результатов выполнения задания**

| Объект реального мира | Подходящий тип геометрии |
|-----------------------|--------------------------|
|                       |                          |
|                       |                          |
|                       |                          |

**2.2. Пространственные базы данных**

1. Опишите алгоритм отнесения объектов территории к выбранному произвольному типу на основе его пространственных признаков.

2. Выполните классификацию геоданных о произвольном типе объектов инфраструктуры культурно-бытового обслуживания. Постройте модель данных.

**2.3. Методы и средства визуализации данных**

Заполните правый столбец табл. 2.3.1, указав в нем, какой тип легенды наилучшим образом будет характеризовать информацию, указанную в левом столбце.

Таблица 2.3.1

**Структура заполнения результатов выполнения задания**

|                                                        |  |
|--------------------------------------------------------|--|
| ВВП стран                                              |  |
| Соотношение городского и сельского населения в странах |  |
| Принадлежность страны к континенту                     |  |
| Возраст и тип застройки                                |  |
| Типы растительности                                    |  |
| Тип климата                                            |  |

Приведите пример легенды по одному из вариантов.

## 2.4. Основы ведения территориальных кадастров

Найдите в Интернете пример публичного кадастра для выбранного вида объектов. Опишите данные, учитываемые в кадастре для одного из учитываемых типов объектов по форме табл. 2.4.1.

Таблица 2.4.1

Структура заполнения результатов выполнения задания

| Учитываемая характеристика                                         |                                                                  | Пример заполнения значений |                 |
|--------------------------------------------------------------------|------------------------------------------------------------------|----------------------------|-----------------|
| Пример в России<br>(или оставить пустым, если в РФ не учитывается) | Аналог другой страны<br>(при наличии)<br>или уникальные сведения | В России                   | В другой стране |
|                                                                    |                                                                  |                            |                 |

## 2.5. Прикладные аспекты ГИС

Разработать проект решения следующих задач, основанных на использовании пространственных данных:

1) оценка возможной зоны затопления в случае наводнения и его прямых последствий (затопление строений жилого и хозяйственного назначения);

2) подтверждение или опровержение гипотезы о негативном влиянии на здоровье жителей жилого массива выбросов в атмосферу отходов крупного химического предприятия;

3) проектирование оптимальной (с точки зрения затрат на строительство) трассы подводного нефтепровода, соединяющего два заранее не заданных пункта на побережье крупного внутреннего водоема, с учетом рельефа и фунтов дна, природно-охранных (распределение нерестилищ, памятников природы и заказников) и других ограничений;

4) оценка числа жителей, обеспеченных устойчивым приемом телепрограмм, транслируемых вновь построенной телевышкой, в условиях горной залесенной местности;

5) создание электронного атласа в Интернете, отражающего оперативные результаты обработки данных Всероссийской переписи населения;

6) выбор места строительства нового супермаркета с учетом конкурентного торгового окружения.

## **2.6. Применение ГИС для решения прикладных задач**

Найдите в Интернете пример применения ГИС для решения прикладных задач. Опишите функционал программных продуктов и применяемые методы по форме табл. 2.6.1.

*Таблица 2.6.1*

**Структура заполнения результатов выполнения задания**

| Функция<br>программного продукта | Методы<br>реализации | Прикладное<br>назначение |
|----------------------------------|----------------------|--------------------------|
|                                  |                      |                          |

---

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ

### Лабораторная работа № 1 «Основы работы с ГИС»

**Цель работы:** ознакомиться с основными функциями ГИС на примере QGIS Desktop.

Работа над QGIS была начата в мае 2002 года, а в 2007 году она стала проектом Open Source Geospatial Foundation (OSGeo) – международной некоммерческой организации, созданной для поддержки совместной разработки и использования геоинформационного ПО с открытым исходным кодом.

На сегодняшний день это одна из наиболее динамично развивающихся и функциональных настольных ГИС, основными преимуществами которой являются:

1) бесплатное распространение – исходя из условий лицензии GNU General Public License, использование, копирование и распространение QGIS для любых целей, в том числе коммерческих, не требует финансовых отчислений;

2) свобода – благодаря открытости исходного кода пользователи не только могут изучать особенности устройства QGIS, но и модифицировать ее в соответствии с собственными потребностями;

3) динамичное развитие – разработка QGIS ведется международной группой разработчиков, которая с 2014 года перешла на четырехмесячный цикл релизов. Таким образом, новая версия выходит три раза в год;

4) обширная документация – для рядовых пользователей доступно Руководство пользователя; разработчики пользуются Поваренной книгой разработчика PyQGIS; есть также документация для тех, кто только начинает свое знакомство с ГИС или же хочет провести полноценный обучающий курс на основе QGIS;

5) интероперабельность – гибкость во взаимодействии с различными аппаратными базами, операционными системами и программным обеспечением, способами представления геоданных и их пространственными характеристиками. Благодаря этому комплексному свойству QGIS может:

- быть установлена для Windows, Mac OS X, Linux, BSD, Android;
- поддерживать различные форматы и модели данных, а именно: более 60 форматов растровых данных (библиотека Geospatial Data Abstraction Library – GDAL); более 20 – векторных (OGR Simple Features Library), взаимодействие с базами геоданных, OGC-сервисами;
- взаимодействовать с данными в различных проекциях и системах координат (в том числе и пользовательских) через библиотеку проекций Proj.4.

QGIS изначально задумывалась как просмотрщик пространственных таблиц PostGIS, но со временем превратилась в полнофункциональную ГИС, способную решать широкий спектр задач.

Файлы установки QGIS (на момент написания методических указаний актуальной является версия QGIS 3.16 'Hannover') для различных операционных систем доступны на странице загрузок официального сайта <https://qgis.org/ru/site/forusers/download.html>.

Установочные файлы для MS Windows распространяются в вариантах установщика OSGeo4W (32 и 64 бит), а также в виде автономных установщиков 32 и 64 бит – процедура установки QGIS с помощью последних ничем не отличается от традиционной установки ПО в ОС Windows.

Программа QGIS работает с файлами, называемыми «проектом».

*Проект* – это файл, содержащий внутри себя электронную карту для работы с пространственными данными.

Файл проекта имеет расширение «qgz» (к примеру, файл «Карта\_мира.qgz» и другие, которые располагаются в папке данных). В более ранних версиях QGIS файлы проекта имели расширение «qgs», которое оставлено в текущей версии для обратной совместимости.

### ***Интерфейс Проекта***

1. «Слой» – это таблица содержания *Проекта*. Здесь отображается список содержимого Проекта, как в легенде бумажной карты.

2. Рабочее пространство *Проекта* – это электронная карта, в которой отображаются пространственные данные.

Строка состояния предоставляет общую информацию о просмотре карты и обработанных или доступных действиях, а также предлагает инструменты для управления видом карты:

1) в левой части строки состояния расположен виджет быстрого поиска, с помощью которого можно найти и запустить любую функцию или параметры в QGIS;

2) в области рядом с виджетом быстрого поиска при необходимости отображается сводка выполненных действий (например, выбор объектов на слое, удаление слоя) или подробное описание инструмента, над которым наводится курсор (доступно не для всех инструментов).

В случае длительных операций, таких как сбор статистики в растровых слоях, выполнение алгоритмов обработки или визуализация нескольких слоев в виде карты, в строке состояния отображается индикатор выполнения.

3. Параметр «**Координата**» показывает текущее положение курсора мыши, отслеживая его при перемещении по карте.

Рядом с отображением координат находится отображение поля масштаба, показывающее масштаб карты. Справа от него расположена кнопка блокировки, позволяющая заблокировать текущий масштаб и использовать уровень приближения. Уровень приближения выражается в процентах.

Справа от уровня приближения находится инструмент для задания текущего поворота карты по часовой стрелке в градусах.

Далее расположен флажок, который можно временно использовать для предотвращения рендеринга слоев в виде карты, перерисовки электронной карты Проекта. Отключать рендеринг следует в том случае, если перерисовка карты выполняется медленно.

В свойствах проекта среди прочих параметров определяется система координат карты Проекта. Системы координат могут быть двух типов:

1. Географические системы координат, в которых используются географические координаты (широта и долгота) для описания объектов. Эти системы координат не имеют проекции. В качестве единицы измерения используют, как правило, градусы.

2. Прямоугольные системы координат, в которых используются декартовы координаты (X и Y) для описания объектов. Эти системы координат используют проекции, поэтому их иногда называют системами координат проекции, проекционными системами координат, спроектированными системами координат, системами координат карты или картографическими системами координат. В этих системах координат, как правило, в качестве единицы измерения используются метры.

Современные ГИС поддерживают множество различных систем координат и преобразование между системами координат.

Некоторые ГИС в том или ином объеме поддерживают преобразование между системами координат «на лету». Это позволяет отображать данные, сохраненные в разных системах координат в одной электронной карте.

Подробнее о системах координат и о перепроецировании «на лету» можно прочитать в материале «Краткое введение в ГИС» ([https://docs.qgis.org/3.16/ru/docs/gentle\\_gis\\_introduction/](https://docs.qgis.org/3.16/ru/docs/gentle_gis_introduction/)).

Карта в Проекте, как и бумажная карта, состоит из слоев. Это слои наборов пространственных данных. В бумажной карте данные слои перечислены в легенде карты. В Проекте эти слои указываются в Таблице содержания Проекта. Окно Таблицы содержания Проекта называется «Слои».

Для того чтобы выполнить какую-либо операцию со слоем пространственных данных, его надо выделить – сделать активным.

Активный слой выделяется полосой, по умолчанию синего цвета, когда курсор в окне «Слои», и серого цвета, когда фокус курсора смещен в другие области программы.

### ***Строка инструментов***

1. *Определить объекты* – открывает по клику на объект окошко информации об объекте.

2. Набор инструментов – открывает список инструментов по клику на него. Среди них:

*Измерить линию* – для быстрого измерения расстояние от точки до точки;

*Измерить площадь* – для быстрого измерения площади полигона (многоугольника);

*Измерить угол* – для быстрого измерения угла между линиями.

Для использования инструмента *Определить объекты* нужно:

1. Сделать активным слой, по которому хотите получить информацию (к примеру, «Страны»).

2. Выбрать инструмент *Определить объекты*.

3. Кликнуть на интересующий объект – откроется окно информации (к примеру, кликнуть на территорию России).

Для использования инструмента *Измерить линию* нужно:

1. Выбрать инструмент *Измерить линию*.
2. Кликами в местах поворота указать линию (т. е. по отрезкам).
3. Кликом на правую клавишу мышки указать последнюю точку линии.

Информация по измеренному расстоянию будет отображаться в окне *Измерить*:

1. Линия измеряемой длины (серая).
2. Длина сегментов (отрезков) линии.
3. Общая длина линии.

### ***Инструмент Измерить площадь***

Для использования инструмента *Измерить площадь* нужно:

1. Выбрать инструмент *Измерить площадь*.
2. Кликами в местах поворота указать границу полигона.
3. Кликом на правую клавишу мышки указать последнюю точку границы.

Информация по измеренной площади будет отображаться в окне *Измерить*.

### ***Инструмент Измерить угол***

Для использования инструмента *Измерить угол* нужно:

1. Выбрать инструмент *Измерить угол*.
2. Тремя кликами указать линию, угол которой будет измеряться.

Слой пространственных данных представляет в Проекте набор пространственных данных. Наборы пространственных данных бывают разного типа, и из разных источников (ресурсов).

### ***Источники пространственных данных***

1. Набор пространственных данных может храниться в виде файла или нескольких взаимосвязанных файлов на жестком диске.

2. Набор пространственных данных может быть представлен в локальной или удаленной базе данных.

3. Набор пространственных данных может быть доступен в виде сетевого сервиса локальной сети или в сети Интернет.

Наборы пространственных данных, сохраняемые в виде файлов на жестком диске, имеют разные форматы записи. Программа QGIS поддерживает несколько основных форматов, среди которых: шейп-файлы (ESRI shape-файлы), растровые файлы и другие. Учебный набор пространственных данных содержит данные в формате шейп-файлов и растровых файлов (GeoTIFF).

### ***Типы пространственных данных***

1. Векторный – отдельно описывают каждый пространственный объект. Векторные данные – это аналог бумажной карты в ГИС.

2. Растровый – представляют территорию в виде цельной картинки (к примеру, спутниковый снимок).

3. Специализированные пространственные форматы (TIN, GRID) – представляют статистические поверхности (к примеру, рельеф или плотность населения).

### ***Геометрия векторных данных***

Объекты векторных слоев различаются по геометрии. Выделяют четыре основных типа:

1. Аннотации (надписи) – пояснительные обозначения или названия объектов.

2. Точечные (внемасштабные) объекты – указывают точку расположения. Первое, что необходимо понять, когда речь идет о точечных объектах, что их использование зависит от масштаба. Рассмотрим это на примере городов. Если у нас есть мелкомасштабная карта (которая охватывает большую область), города лучше обозначать точками. Однако, при увеличении карты и переходе к более крупным масштабам, границы городов лучше отображать как полигоны.

Выбор точек для представления объектов реального мира зависит от используемого масштаба (как далеко вы находитесь от объекта), удобства (создание точечных объектов занимает меньше времени) и типа объектов (некоторые вещи, например, телефонные будки, нет смысла хранить в виде полигонов).

3. Линейные объекты – указывают протяженность (длину) объекта.

4. Полигональные (площадные, масштабные, многоугольные) объекты – указывают границу расположения.

Для того чтобы создать собственную карту в Проекте, необходимо добавить слои пространственных данных. При частой работе нужно активировать панель инструментов для работы со слоями, кликнув правой клавишей мышки по панели Меню и выбрав нужную панель инструментов (аналогично можно активировать/деактивировать другие инструменты).

### ***Добавление векторного слоя пространственных данных в Проект***

Общий порядок добавления векторного слоя достаточно прост:

1. Нажать на кнопку **Добавить векторный слой** (или Меню **Слой** → **Добавить слой** → **Добавить векторный слой**).

2. В окне **Менеджер источников** данных указать тип источника набора пространственных данных и местоположение.

3. В том случае, если программа не может распознать систему координат набора пространственных данных, указать систему координат набо-

ра пространственных данных в окне выбора системы координат (как работать с этим окном рассматривается в предыдущем занятии).

Для того чтобы использовать многие дополнительные функции QGIS, необходимо установить модуль расширения (далее модуль). Модуль – это программа, расширяющая возможность QGIS. Чтобы ей воспользоваться, нужно ее получить и установить. Способов установки несколько. Выбор зависит от языка программирования, на котором написан модуль, и источника файлов расширения.

Расширения могут быть созданы на языках программирования C++ и Python.

По источнику модули могут быть следующими:

- входят в ядро QGIS (модули ядра), и могут быть на обоих языках;
- официальный репозиторий, только Python;
- неофициальные (авторские) репозитории, только Python.

Модули ядра устанавливаются при установке самого QGIS. Подавляющее число модулей создано на языке Python.

### **Задания**

1. Откройте проект «Карта мира» в папке данных и разверните окно программы на весь экран.

2. Выключите отображение всех слоев, кроме слоя «Города».

3. Увеличьте карту к Европе.

4. Сделайте активным слой «Города» (кликнуть на него).

5. Получите информацию по городу Москва с помощью инструмента *Определить объекты*.

6. В списке атрибутов пролистайте до атрибута «POP2010» (население к 2010 году) и запишите его в протокол.

7. Включите отображение других слоев.

8. Инструментом *Измерить линию* измерьте расстояние по линии Лондон – Берлин – Москва и запишите его в протокол. Если не можете определить расположение города, то найдите его с помощью инструмента *Определить объекты*.

9. Инструментом *Измерить площадь* измерьте площадь полигона по границе Москва – Киев – Минск – Санкт-Петербург и запишите в протокол.

10. Инструментом *Измерить угол* измерьте угол по линии Рим – Париж – Москва. Запишите величину угла в протокол.

### **Вопросы к отчету**

1. Перечислите основные функции QGIS.
2. Что понимается под «Рабочим пространством проекта» в QGIS?
3. Какие типы и форматы данных поддерживает программа QGIS?
4. Какие операции можно выполнить с помощью инструментов QGIS?

## Лабораторная работа № 2

### «Отображение пространственных данных»

**Цель работы:** получить навыки работы с двумерной визуализацией данных с использованием средств ГИС.

Данные в ГИС описывают, как правило, реальные объекты, такие как автомагистрали, различные виды строений, массивы растительности и многое другое. Реальные объекты можно разделить на две абстрактные категории: дискретные (дома, территориальные зоны) и непрерывные (рельеф, уровень осадков, среднегодовая температура). В зависимости от типа категории используют векторные и/или растровые данные.

Картографическая визуализация – это способ наглядно отобразить большое количество геоинформации и поместить имеющиеся данные в определенный географический контекст. Многочисленные социальные исследования, связанные с городским пространством, часто прибегают к картографированию данных. Однако существующие современные цифровые технологии позволяют значительно расширить диапазон объектов, с которыми можно работать, добавив при этом интерактивности во взаимодействие с ними, и создать новые способы взаимодействия благодаря совмещению разных слоев данных.

QGIS позволяет использовать сторонние слои карт, полученных из множества картографических сервисов. Рассмотрим, как осуществляется импортирование тайловых карт на примере сервиса Mapbox (<https://www.mapbox.com/>).

Для этого необходимо сформировать ссылку [https://api.mapbox.com/styles/v1/YOUR\\_USERNAME/YOUR\\_STYLE\\_ID/wmts?access\\_token=<YOUR\\_ACCESS\\_TOKEN>](https://api.mapbox.com/styles/v1/YOUR_USERNAME/YOUR_STYLE_ID/wmts?access_token=<YOUR_ACCESS_TOKEN>), проделав следующие действия:

1. В качестве YOUR\_USERNAME подставить название вашего аккаунта в Mapbox. Его можно узнать в Mapbox Studio (<https://studio.mapbox.com/>).

2. В качестве YOUR\_STYLE\_ID необходимо создать новый стиль в сервисе Mapbox Studio (<https://studio.mapbox.com/>), после чего подставить сформированный ID.

3. В качестве YOUR\_ACCESS\_TOKEN подставить ваш персональный токен, который доступен в профиле пользователя.

Полученная ссылка позволит загрузить определенную подложку карты.

Для добавления карты в QGIS Desktop необходимо добавить слой через меню Layer → AddLayer → Add WMS/WMTS layer... Далее следует убедиться, что вы находитесь на вкладке «Layers» и создать новый слой, нажав на кнопку «New». После чего необходимо заполнить поле названия слоя и указать сформированную выше ссылку (рис. 3.2.1).

После чего нужно нажать на кнопку «ОК».

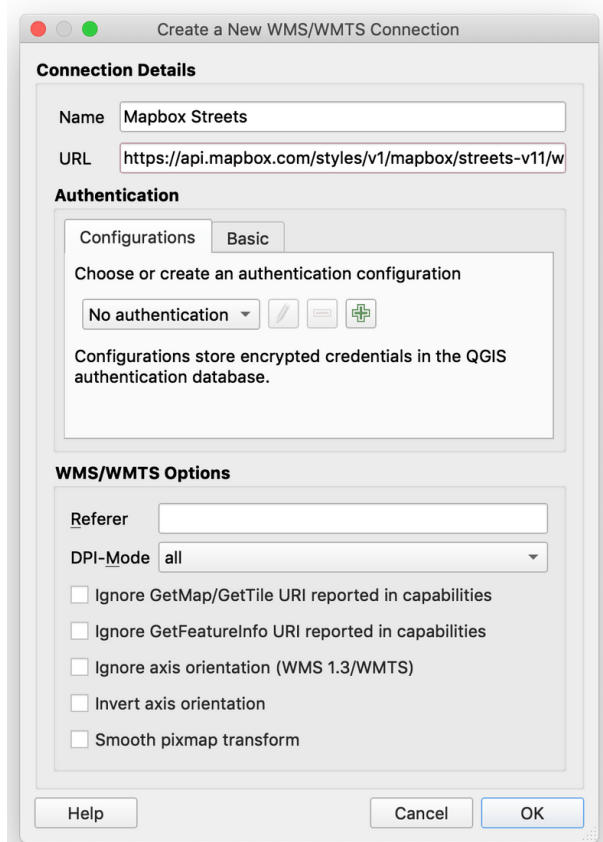


Рис.3.2.1. Окно создания нового слоя WMS/WMTS Connection

Далее необходимо выбрать слой Mapbox и нажать на кнопку «Connect» (рис. 3.2.2).

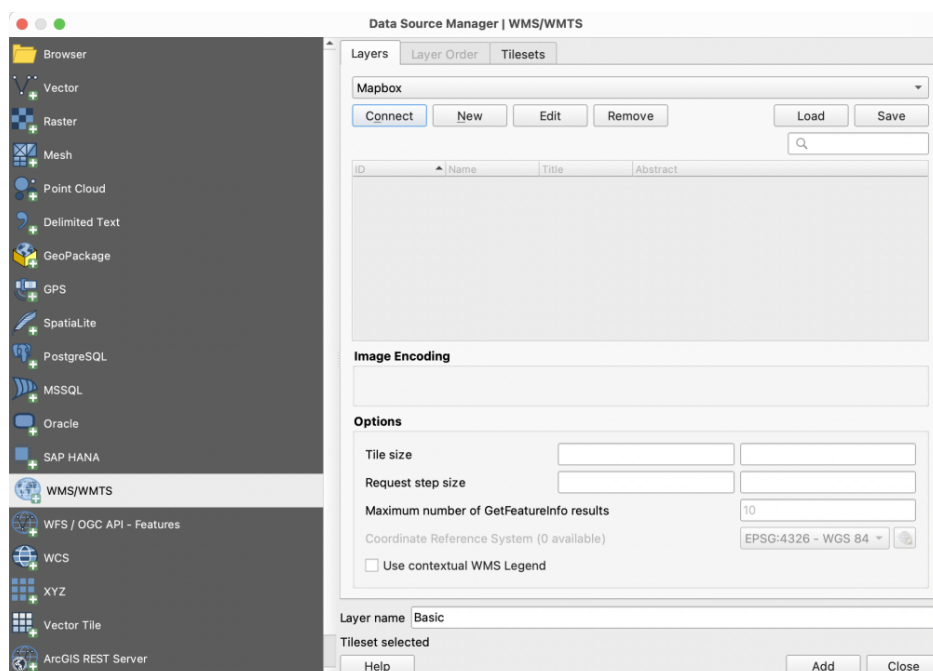


Рис. 3.2.2. Окно интерфейса WMS/WMTS Layers

После выбора нужного слоя следует нажать на кнопку «Add» (рис. 3.2.2.), после чего закрыть диалоговое окно.

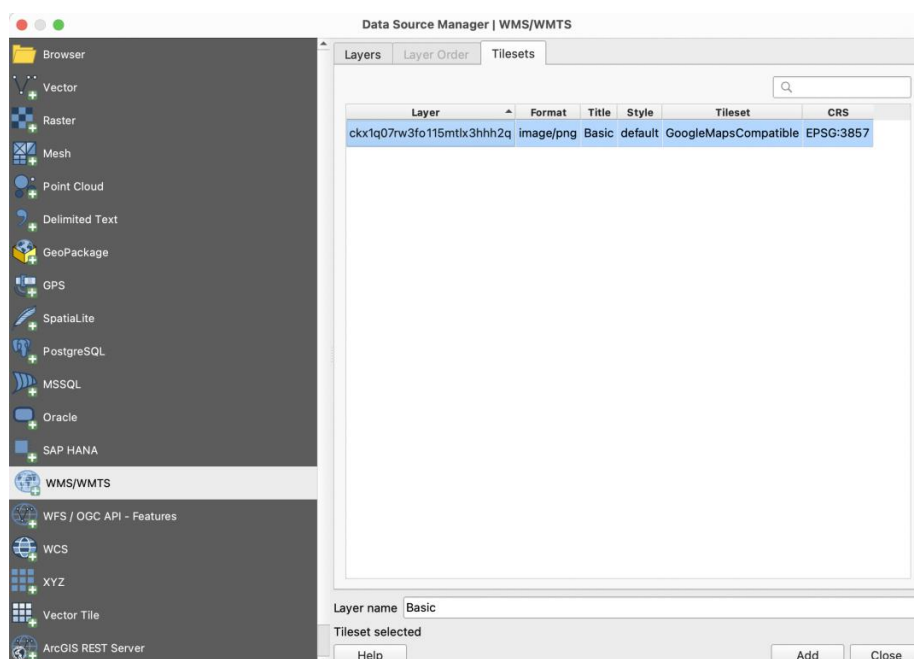


Рис. 3.2.3. Окно интерфейса WMS/WMTS Tilesets

Сервис BBBike позволяет извлекать области из Planet.osm в таких форматах, как OSM, PBF, ESRI Shapefile и так далее.

Для загрузки данных с использованием сервиса BBBike (<https://extract.bbbike.org>) (рис. 3.2.4) необходимо выполнить следующую последовательность шагов:

- 1) переместить карту к нужной локации;
- 2) создать полигон, захватывающий выбранную область на карте;
- 3) в качестве формата файла для экспорта выбрать Esri Shapefile;
- 4) экспортировать данные кнопкой «Extract»;
- 5) скачать готовый архив с данными на ПК.

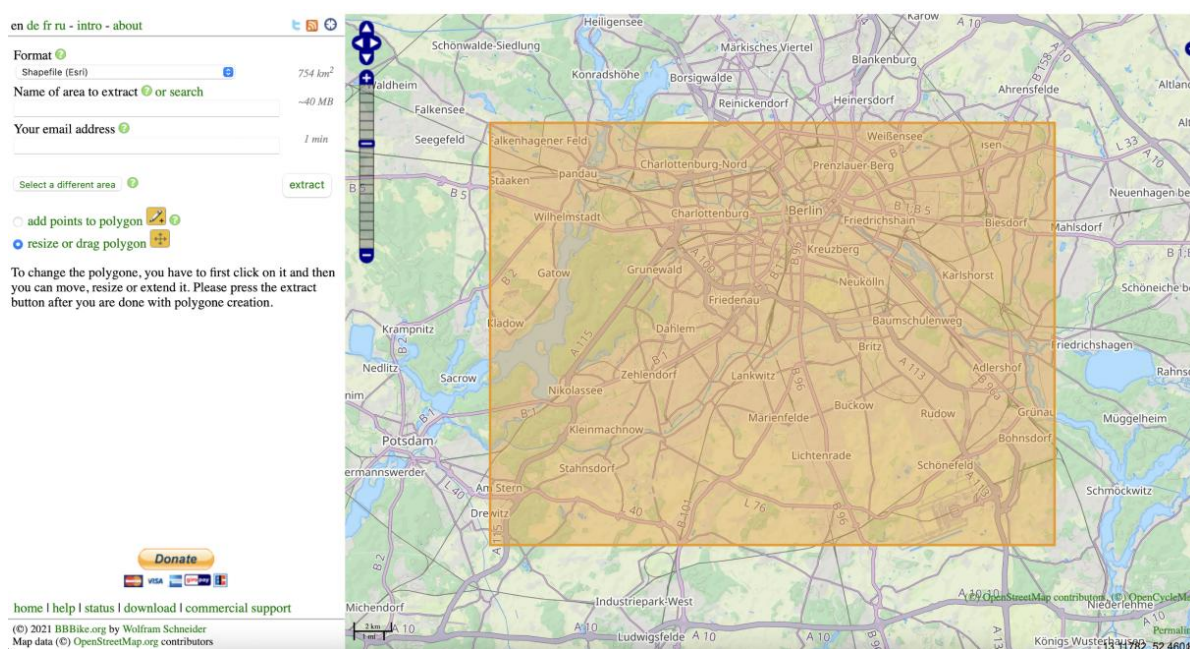


Рис. 3.2.4. Интерфейс сервиса BBBike

Добавить слой из полученного zip-архива с локацией в QGIS Desktop можно с помощью меню Layer → AddLayer → Add Vector Layer...

Для построения сеток понадобится плагин MMQGIS, в случае отсутствия нужно установить его через менеджер плагинов QGIS. Далее необходимо создать слой сетки Hexagon, используя меню MMQGIS → Create → Create Grid Layer.

Необходимо подсчитать, какое количество точек попадает в полигон. Для этого нужно воспользоваться инструментом анализа векторных данных: Processing Toolbox → Vector Analysis Tool → Count points in polygon.

Далее следует произвести фильтрацию полигонов, которые не содержат точек, с помощью инструмента Select features, с условием  $NUMPOINTS > 0$ . После чего нужно настроить цвет и прозрачность полигонов на карте в соответствии с рис. 3.2.5.

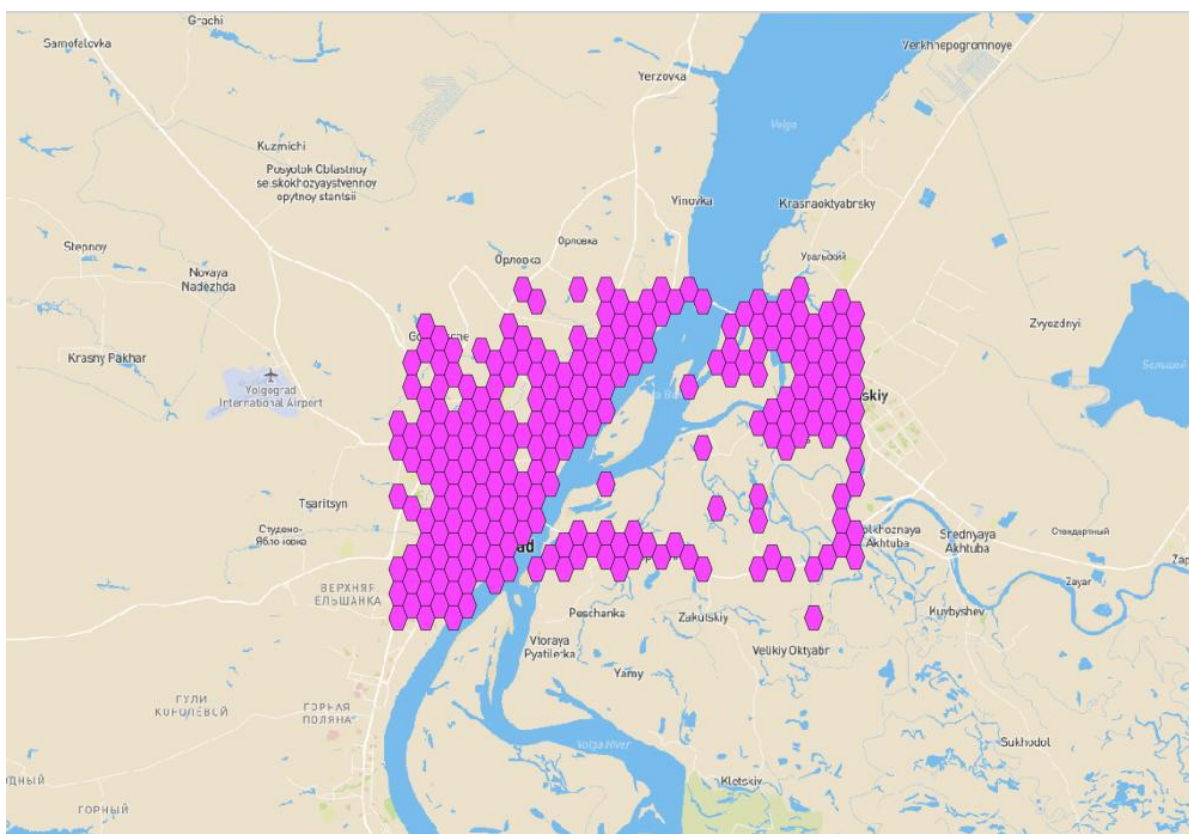


Рис. 3.2.5. Гексагональная сетка

Тепловая карта (англ. *heatmap*) – это графическое представление данных, где значения интересующей переменной выделяются цветом.

Одна из задач использования тепловых карт – это изучение плотности застройки города. Из полученного ранее zip-архива с локацией необходимо открыть слой *buildings* (файл *buildings.shp*). Далее нужно извлечь центроиды из каждого полигона с помощью команды Processing Toolbox →

Polygon centroids. После этого открыть панель «Layer Styling», в соответствии с рис. 3.2.6.

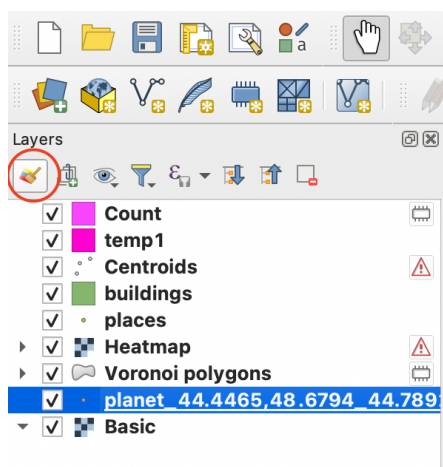


Рис. 3.2.6. Панель слоев

Далее в выпадающем меню нужно выбрать опцию «Heatmap», как показано на рис. 3.2.7.

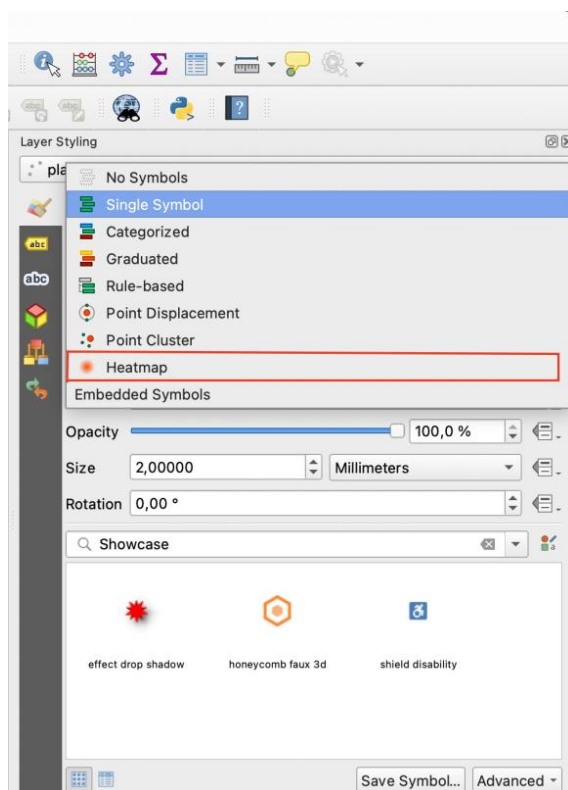


Рис. 3.2.7. Панель «Layer Styling»

После чего необходимо установить цвет и прозрачность слоя тепловой карты. В результате получится тепловая карта, показанная на рис. 3.2.8.

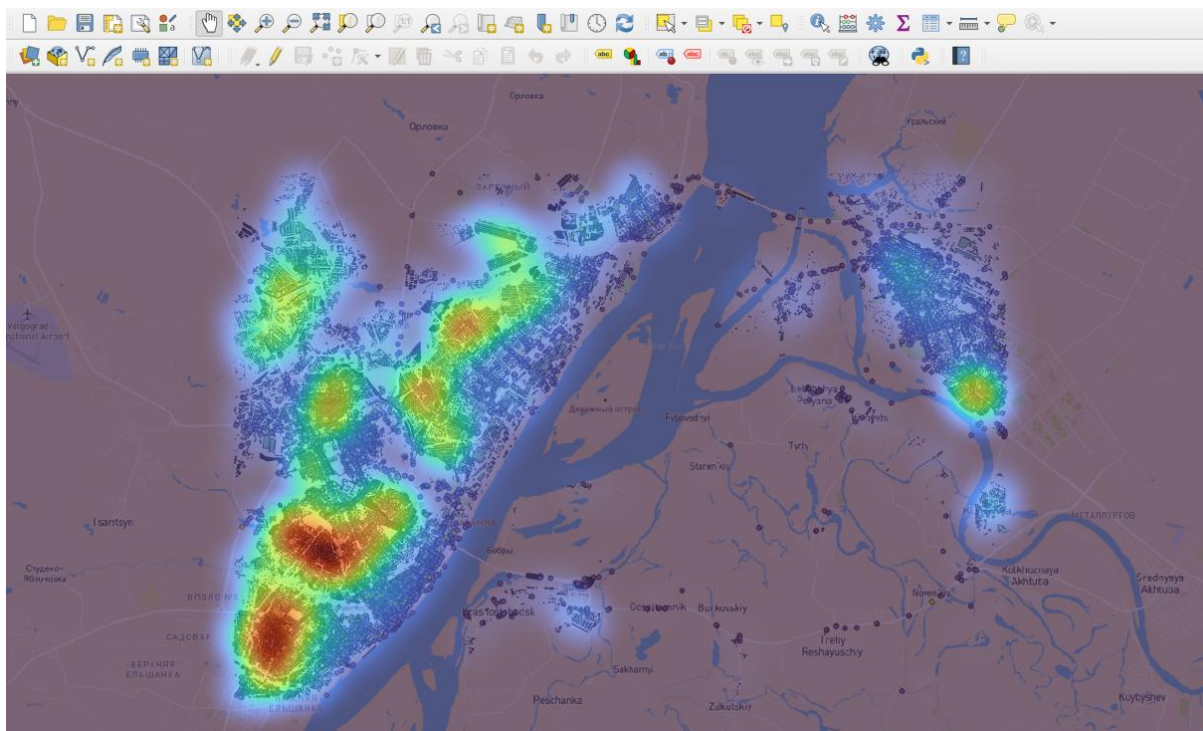


Рис. 3.2.8. Тепловая карта

Диаграмма Вороного (названа в честь российского ученого Георгия Феодосьевича Вороного) представляет собой компактную структуру данных, хранящую всю необходимую информацию для решения множества задач о близости. В результате разбиения пространства диаграммой Вороного по данным будет визуализирован набор областей, каждая из которых содержит ровно одну точку.

В качестве примера изучим плотность остановок общественного транспорта в Волгограде. Для этого необходимо выбрать значения (`type = 'bus_stop'` OR `type = 'tram_stop'`) из слоя точек интереса. Для создания диаграммы в основном меню нужно выбрать пункт `Vector → Geometry`

Tools → Voronoi polygons (либо MMQGIS → Create → Voronoi diagram).  
В итоге получится диаграмма, как на рис. 3.2.9.

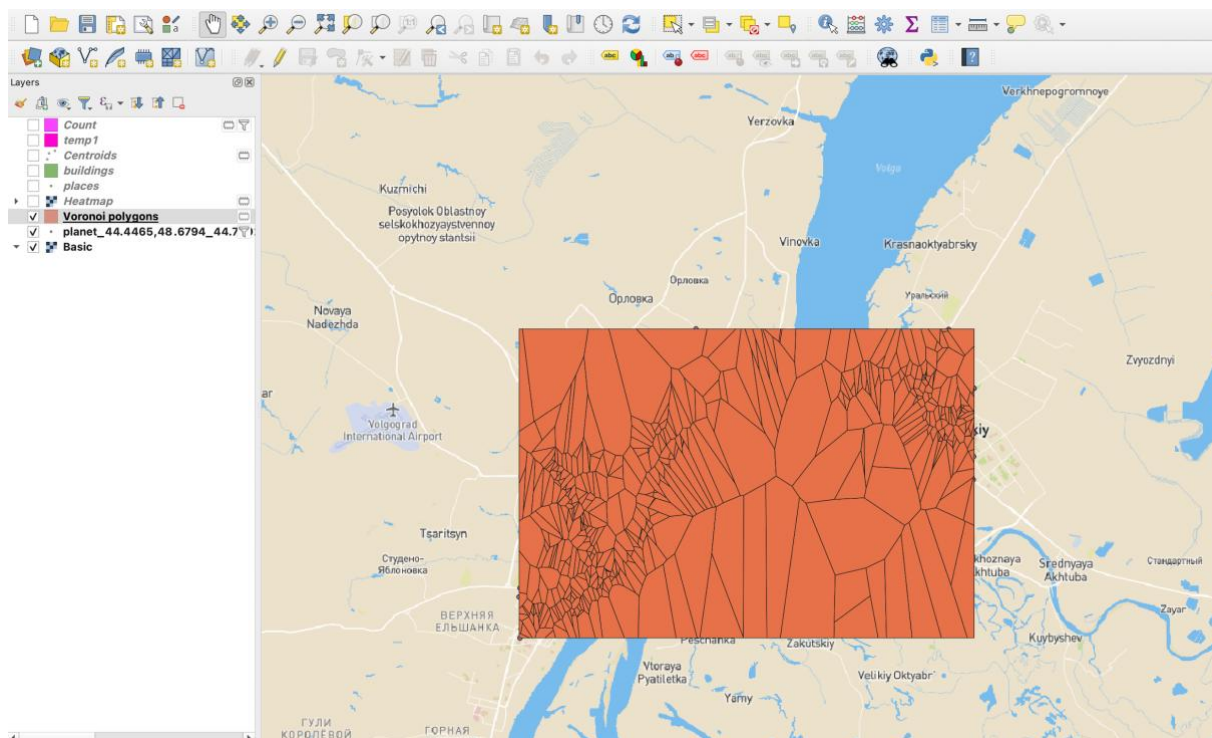


Рис. 3.2.9. Диаграмма Вороного

Получившееся изображение плохо отображает ситуацию на карте. Один из способов сделать изображение более информативным – это окрасить полигоны в зависимости от их площади.

Для этого необходимо открыть панель «Layer styling», выбрать тип стилизации «Graduated»; далее в поле «Value» установить значение «\$area» и подобрать подходящую цветовую схему. В итоге получится диаграмма, как на рис. 3.2.10.

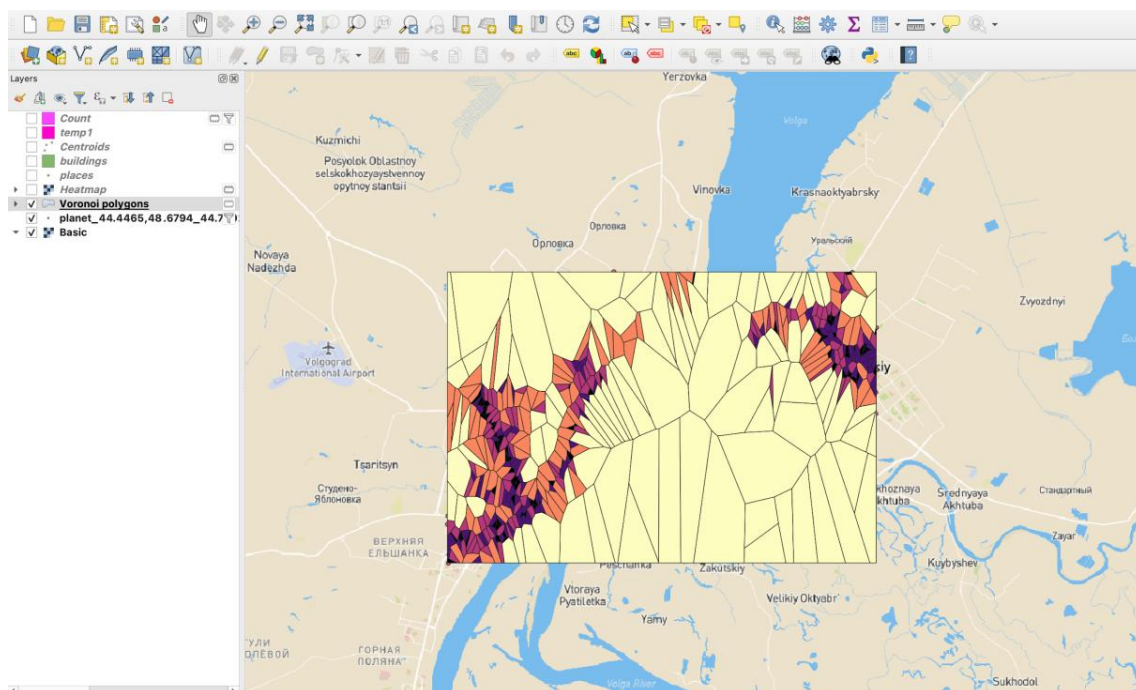


Рис. 3.2.10. Диаграмма Вороного с настроенными стилями

QGIS Desktop имеет функционал подготовки карты для вывода на печать. В качестве исходных данных будем использовать точки интереса из ранее выбранной локации.

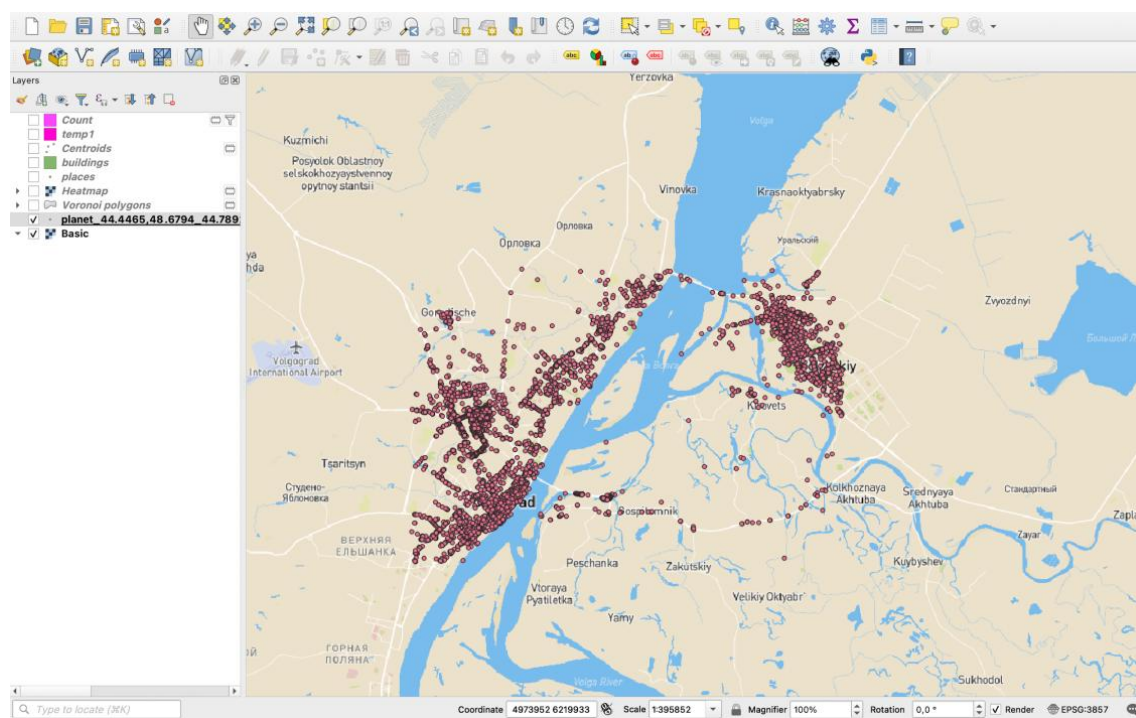


Рис. 3.2.11. Данные о точках интереса г. Волгограда

Далее необходимо в основном меню перейти в Project → New Print Layout. Интерфейс окна подготовки печати изображения представлен на рис. 3.2.12.

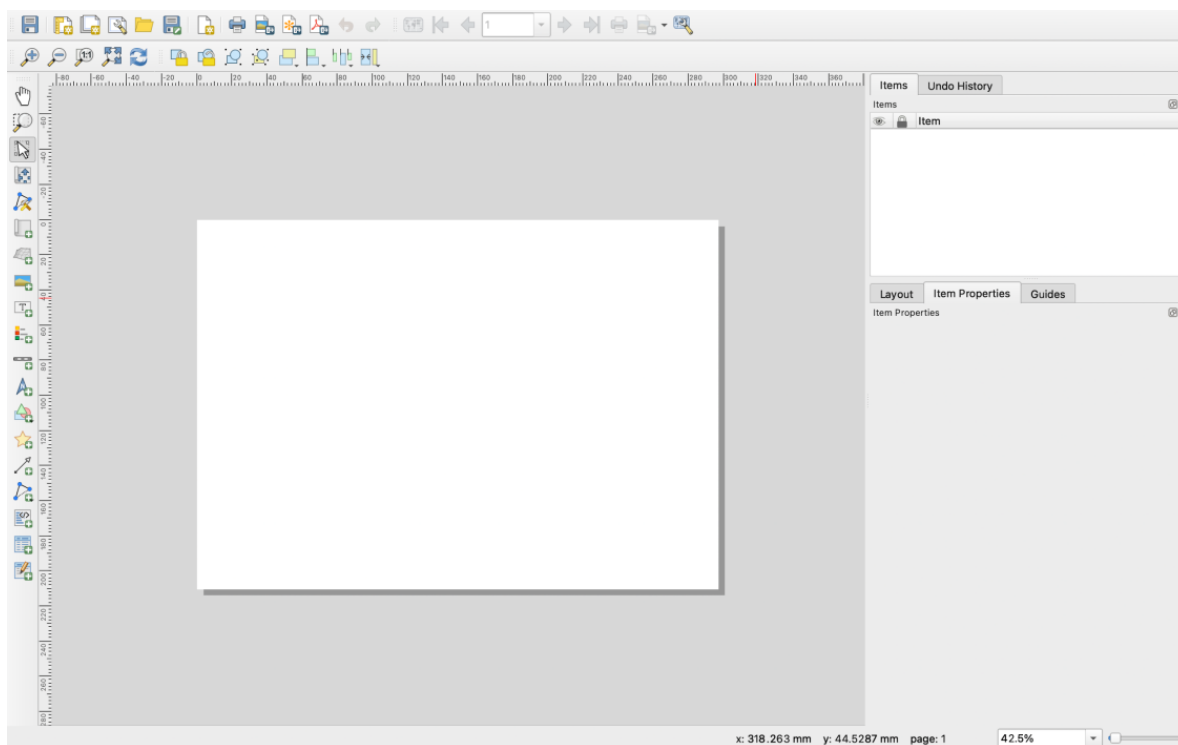


Рис. 3.2.12. Интерфейс окна Print Layout

Используя кнопку «Add Map» на панели инструментов слева, необходимо нарисовать прямоугольник в произвольной области листа, где вы желаете разместить карту. Однако следует учесть, что карта должна содержать заголовок и легенду, тем самым необходимо заранее выделить для этого место на листе.

Обычно карта отображает область, идентичную той, что отображена в QGIS Desktop. Если необходимо изменить отображаемую область на карте или ее масштаб, то нужно перейти в QGIS и выполнить необходимые действия. После чего вернуться в Print Layout и воспользоваться кнопкой «Refresh».

Для добавления заголовка необходимо воспользоваться инструментом «Add label». Чтобы задать свойства заголовка, нужно выбрать текстовый слой во вкладке Items. Также в окне свойств (рис. 3.2.13) можно настроить параметры шрифта и многое другое.

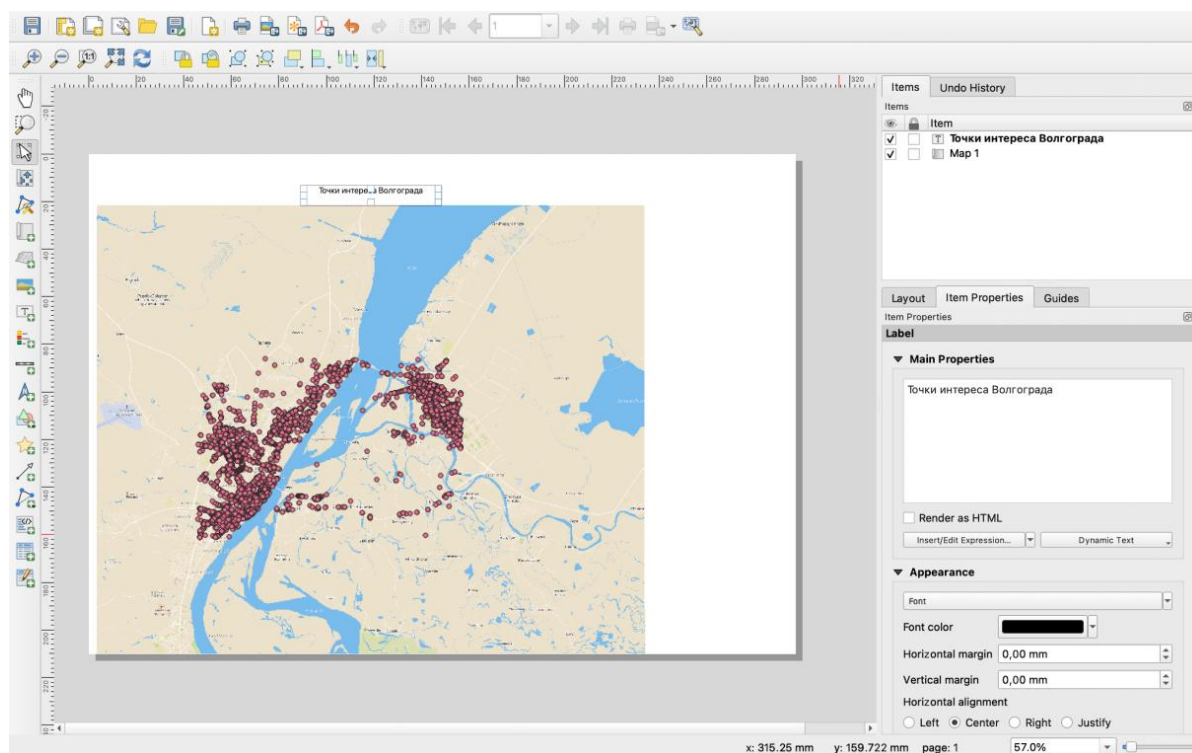


Рисунок 3.2.13 – Добавление заголовка

Для добавления легенды карты используется инструмент «Add legend». Необходимо поместить легенду в правую часть карты в свободное для этого место. Настройка параметров отображения легенды осуществляется через окно свойств. Для возможности редактирования элементов легенды, необходимо отключить автообновление кнопкой «Auto update». Итоговая карта представлена ниже на рис. 3.2.14.

По завершении работы необходимо сохранить получившуюся карту, воспользовавшись кнопкой экспорта «Export as image». Обязательно нужно задать разрешение не ниже 350 DPI, так мы сохраним все детали при печати.

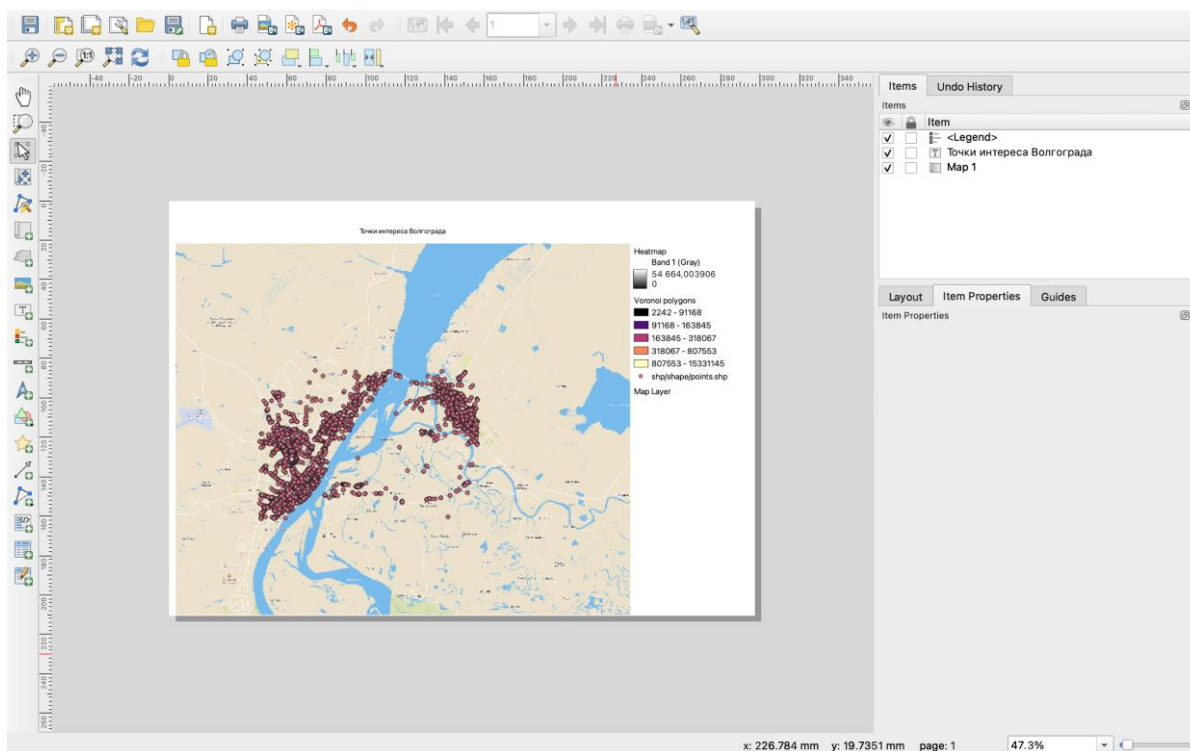


Рис. 3.2.14. Карта с легендой

### Задания

1. Построить гексагональную сетку для выбранных точек.
2. Построить тепловую карту плотности застройки вашего района.
3. Построить диаграмму Вороного для плотности общественного транспорта в выбранном районе.

### Вопросы к отчету

1. Назовите существующие способы представления географически привязанной информации.
2. Какие существуют особенности картографической визуализации?
3. Назовите существующие способы районирования пространства и их особенности.
4. Какие данные можно анализировать, используя ГИС?

Лабораторная работа № 3  
«Платформа **OpenStreetMap**:  
основные функции, инструменты и библиотеки»

**Цель работы:** формирование навыков работы с данными платформы OpenStreetMap.

**OpenStreetMap** ([www.openstreetmap.org](http://www.openstreetmap.org)) – это картографический сервер, на котором карта создается пользователями (как в Википедии). Карта охватывает весь мир. Есть регионы, где карта очень подробна; есть регионы, где она практически отсутствует. Любой пользователь, в том числе и вы, может принять участие в создании этой карты.

Карту **OpenStreetMap** можно использовать и для создания своих карт на определенных лицензионных условиях. На сайте NextGIS из данных OpenStreetMap и других открытых источников создаются обновляемые наборы слоев по любой точке мира. Данные наборы слоев доступны в форматах ESRI Shapefile, ESRI Geodatabase, Mapinfo TAB, GeoJSON, PBF, OSM (XML), SQL (PostgreSQL), CSV и PDF, что позволяет использовать их практически в любой ГИС. Например, можно заказать выгрузку данных OpenStreetMap по Волгоградской области ([data.nextgis.com/ru/region/RU-VGG/](http://data.nextgis.com/ru/region/RU-VGG/)).

В данной лабораторной работе мы рассмотрим, как выгрузить данные **OpenStreetMap** самостоятельно, используя плагин **QuickOSM** ([https://docs.qgis.org/3.16/en/docs/training\\_manual/qgis\\_plugins/plugin\\_examples.html](https://docs.qgis.org/3.16/en/docs/training_manual/qgis_plugins/plugin_examples.html)), который следует установить из официального репозитория плагинов QGIS.

Напомним, что для работы с модулями на языке Python (а это подавляющее большинство модулей) служит специальный модуль – «Установщик модулей», предназначенный для загрузки дополнительных расширений из официального и/или авторского репозитория. «Управление модулями» тоже является модулем ядра QGIS, поэтому устанавливать и вклю-

чать его не нужно. Вызвать его можно сразу после установки и запуска QGIS (рис. 3.3.1).

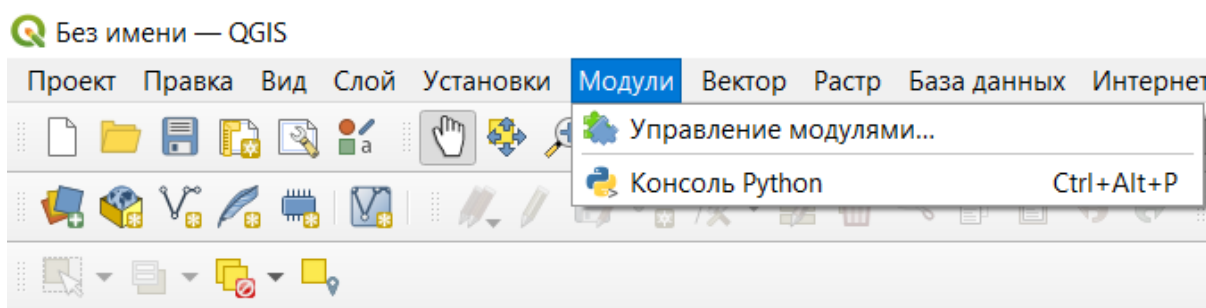


Рис. 3.3.1. Вызов модуля Управление модулями в меню Модули

Для установки модуля переключимся на закладку «Не установленные» и, либо отсортировав по имени, либо введя часть названия модуля в строку поиска, найдем интересующий нас модуль.

После того как он найден, выберем его и нажмем «Установить модуль» (рис. 3.3.2).

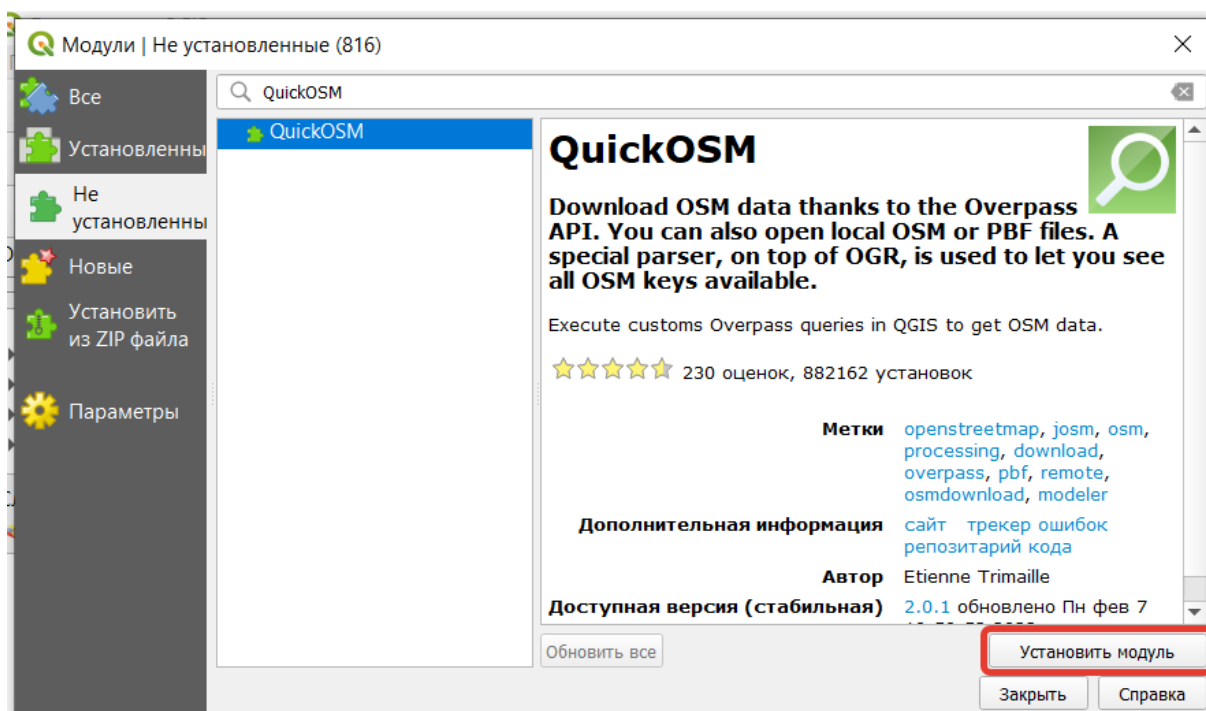


Рис. 3.3.2. Установка модуля QuickOSM

После загрузки модуля он будет автоматически активирован.

После того, как модуль **QuickOSM** установлен, в меню Вектор появится соответствующий пункт **QuickOSM** (рис. 3.3.3).

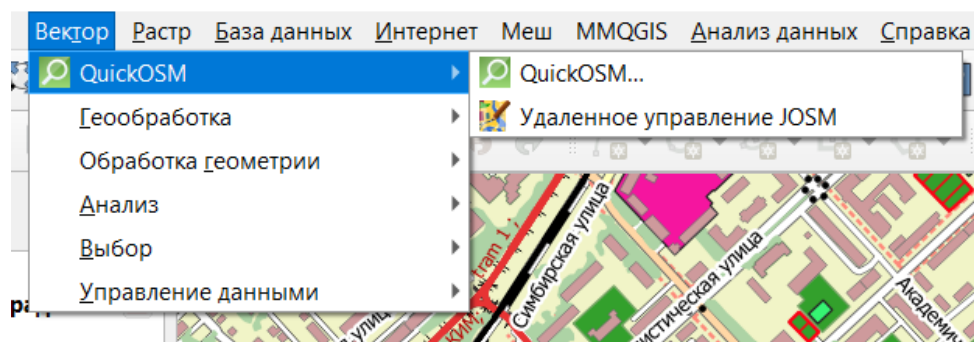


Рис. 3.3.3. QuickOSM в меню Вектор

### ***Выгрузка данных OpenStreetMap с использованием плагина QuickOSM***

Создадим пустой проект и дадим ему имя **VolgogradRegion**.

В качестве подложки установим карту **Google Road**, используя модуль **QuickMapServices**, и отмасштабируем ее до Волгоградской области.

Далее загрузим данные о районах Волгоградской области и их границах.

Семантика OSM состоит из набора свойств (называемых «тегами»), описывающих географические классы, использование которых определяется участниками проекта на специальном веб-сайте Wiki ([https://wiki.openstreetmap.org/wiki/RU:Объекты\\_карты](https://wiki.openstreetmap.org/wiki/RU:Объекты_карты)). Модель данных проста и состоит из узлов, путей и отношений. Каждый сопоставленный объект сопровождается тегом, который представляет из себя соединение ключа и его значения.

Для загрузки данных о районах Волгоградской области и их границах воспользуемся тегом с парой ключ-значение **admin\_level** и **6**, указав в поле «В» **Волгоградская область**. Укажем, что результат нужно сохранить в рабочем каталоге, как ESRI Shapefile (рис. 3.3.4).

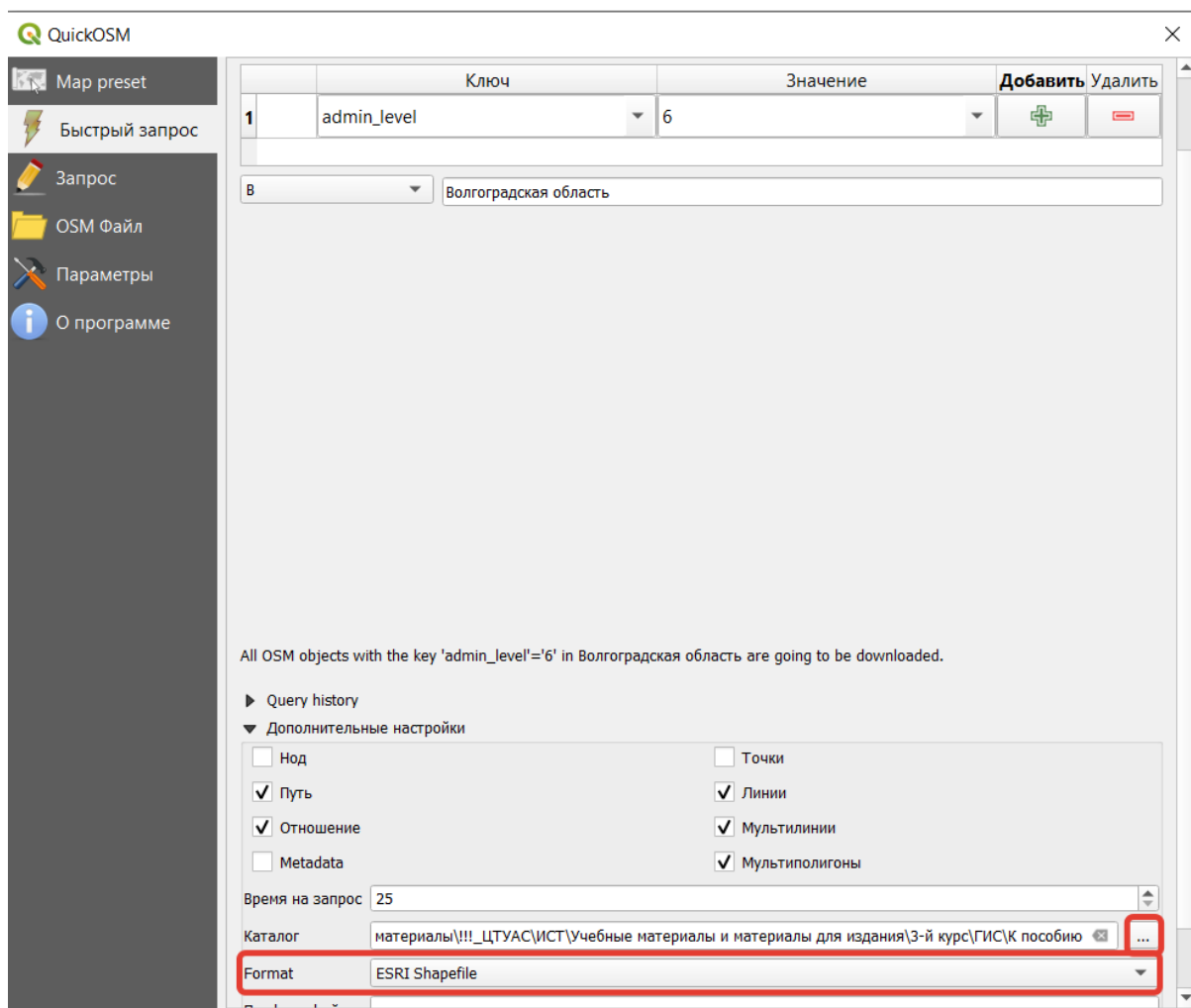


Рис. 3.3.4. Загрузка данных о районах Волгоградской области и их границах

Выполнив запрос, мы получим два векторных слоя: полигональный с территориями районов и линейный с их границами.

Для корректного отображения кириллических символов в атрибутах элементов, нужно в свойствах слоев изменить значение кодировки с **System** на **UTF-8** (рис. 3.3.5).

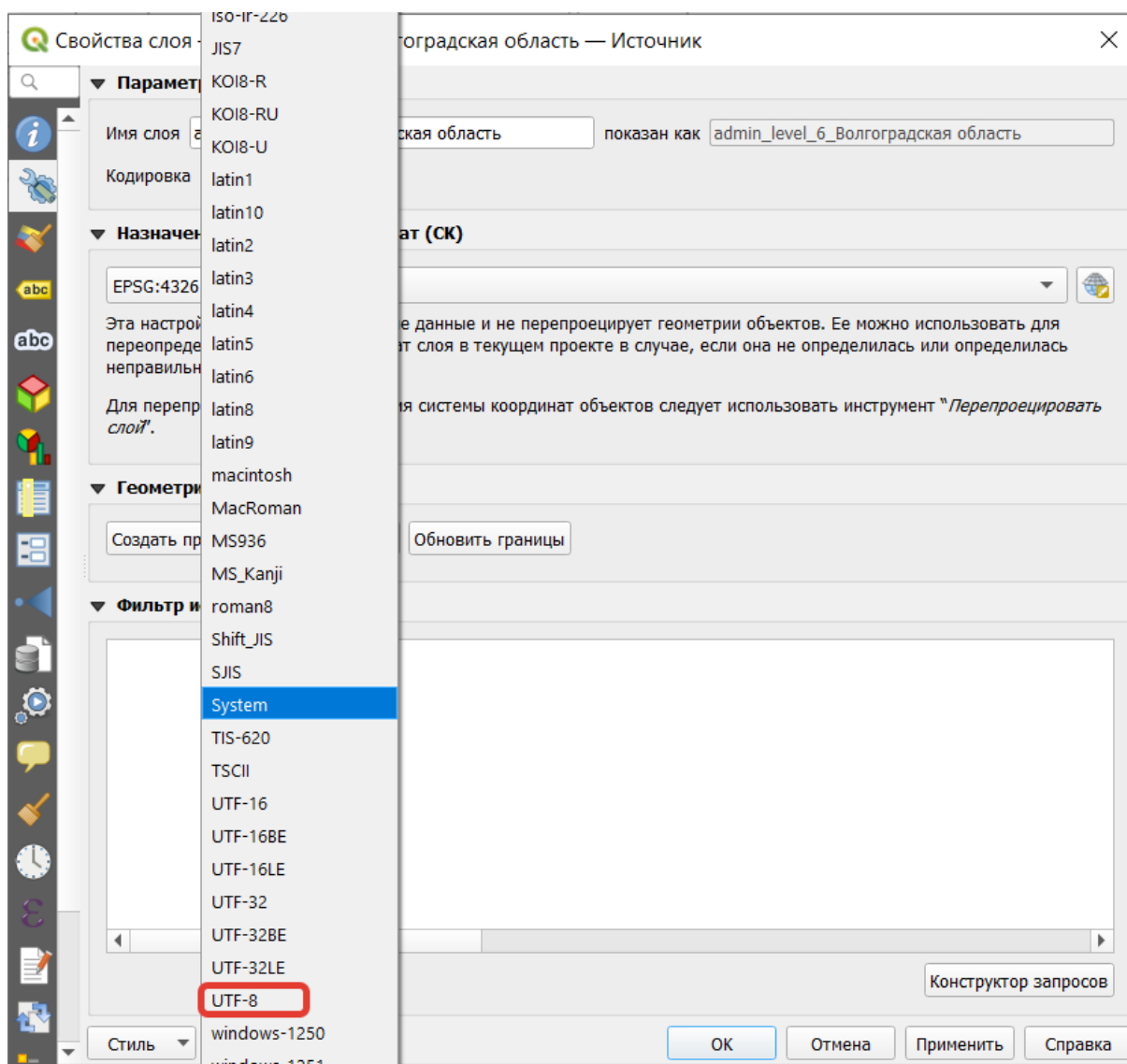


Рис. 3.3.5. Изменение значения кодировки в свойстве слоя

## Задания

1. Для полигонального слоя, содержащего данные о районах Волгоградской области, укажите тип легенды **Уникальные значения** – поле классификации «*name*» и сделайте скриншот результата.

2. По ключу **place** со значениями **city** и **town**, указав в дополнительных настройках только **Нод** и **Точки**, получите точечный слой с основными населенными пунктами (рис. 3.3.6).

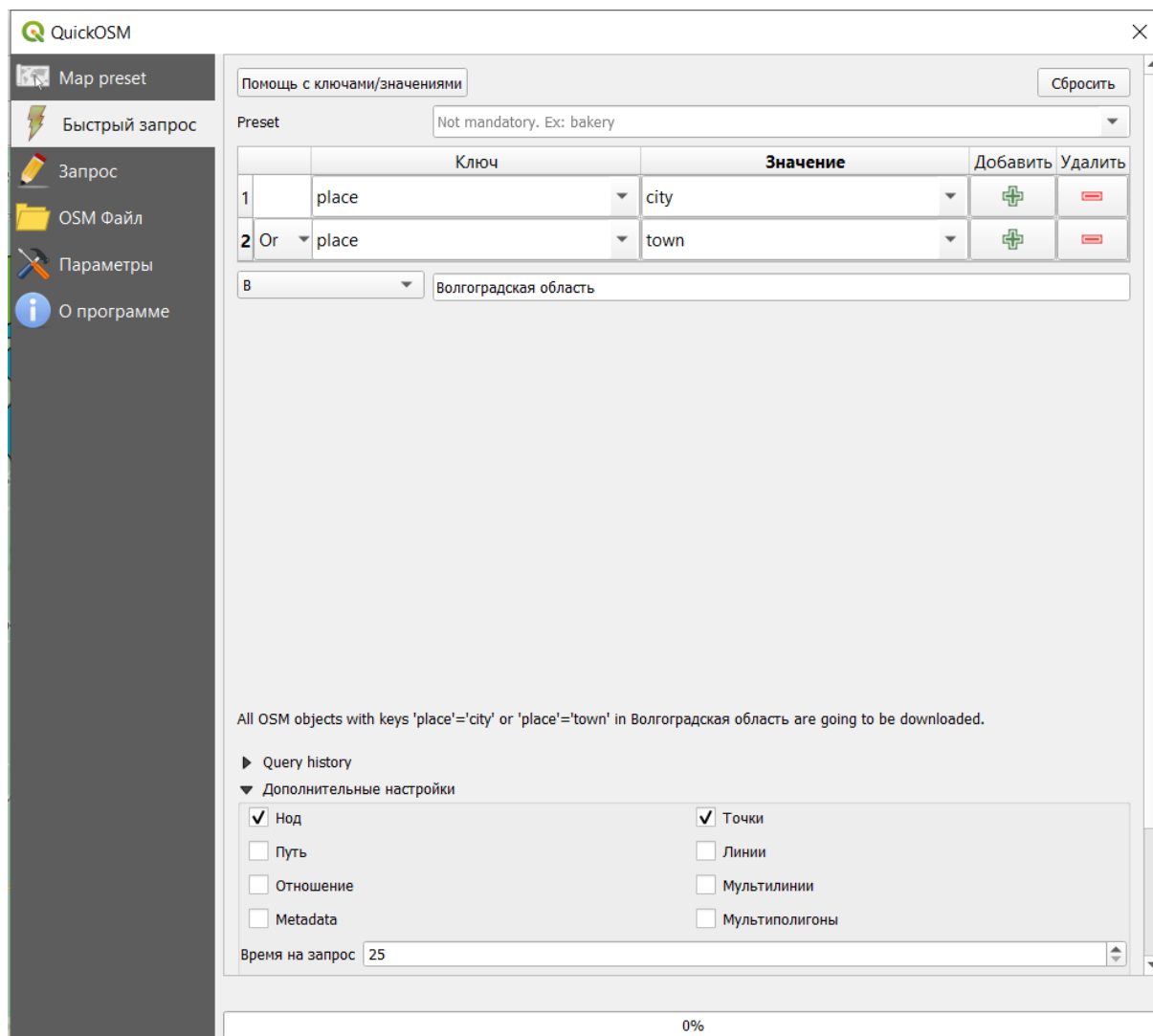


Рис. 3.3.6. Параметры запроса для получения точечного слоя с основными населенными пунктами

3. Для полученного слоя:

а) задайте тип легенды **Уникальные значения** – поле классификации «**place**»;

б) для значения «**city**» выберите в библиотеке символов символ **diamond red**. Укажите размер 4,5 мм;

в) в легенде укажите «**Крупные города**»;

г) для значения «**town**» укажите размер маркера 3 мм и задайте для него синий цвет;

- д) в легенде укажите «Крупные населенные пункты»;
- е) задайте подписи для слоя по полю надписи «*name*», шрифт «*Arial Rounded MT Bold*», размер 8;
- ж) установите буфер размером 1 мм;
- з) сделайте скриншот результата.

### Вопросы к отчету

1. Опишите назначение проекта OpenStreetMap.
2. Опишите модель и структуру данных OpenStreetMap.
3. Что содержится в файле формата OSM XML?
4. Что такое тег элемента OSM?

## Лабораторная работа № 4

### «Задачи пространственного анализа»

**Цель работы:** получить навыки выполнения пространственного анализа на примере задач городской логистики.

Для решения задач городской логистики активно используются методы пространственного анализа данных. Они связаны с определением оптимальных маршрутов, построением зон обслуживания, определением ближайших сервисных точек, размещением сервисных точек.

В основе решения этих задач лежит сетевая модель данных, являющаяся частным случаем векторной модели. В сетевой модели дорожная сеть представляется в виде графа.

#### *Добавление исходных данных*

Создадим пустой проект и дадим ему имя **Volgograd**.

Загрузим данные о районах Волгограда и их границах, воспользовавшись для этого модулем **QuickOSM**.

Создадим быстрый запрос по тегу с парой ключ-значение **admin\_level** и **9**, указав в поле «В» **Volgograd**. Укажем, что результат нужно сохранить в рабочем каталоге, как ESRI Shapefile (рис. 3.4.1).

Для полигонального слоя, содержащего данные о районах Волгограда, укажем тип легенды **Уникальные значения** – поле классификации «**name**» (рис. 3.4.2). Непрозрачность слоя зададим равной 35,8 %.

Далее добавим в проект слои **RoadNetworkVolgograd** (улично-дорожная сеть) и **VolgogradBuildings** (здания и сооружения). Оба этих слоя подготовлены в формате **GeoPackage** на основе данных **OpenStreetMap**.

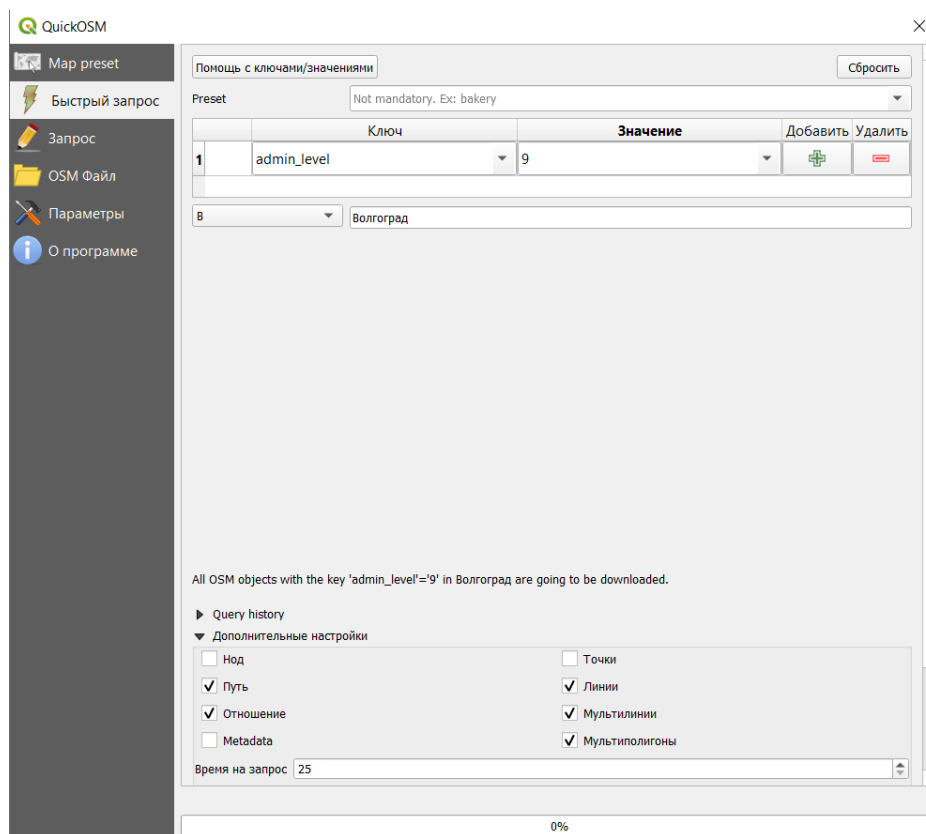


Рис. 3.4.1. Параметры запроса для получения данных о районах Волгограда и их границах

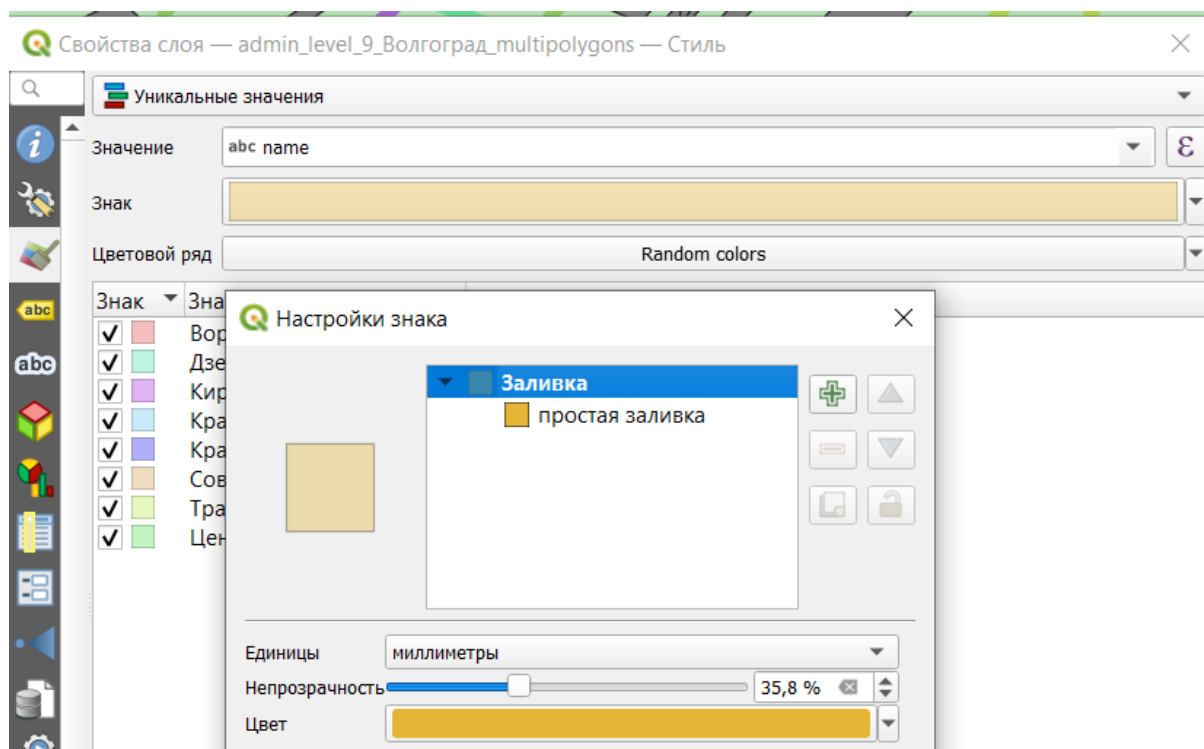


Рис. 3.4.2. Классификация данных о районах

Зададим для слоя *Здания и сооружения* следующие настройки отображения:

- полигоны серого цвета (#8a8a8a) без обводки;
- подписи для слоя по полю «A\_HSNMBR», размер 6.

Для слоя *Улично-дорожная сеть* – линии синего цвета толщиной 0,26 мм. С подписями для слоя по полю «NAME», размер 83.

Фрагмент фрейма карты после увеличения масштаба изображения до 1:5000 показан на рис. 3.4.3.

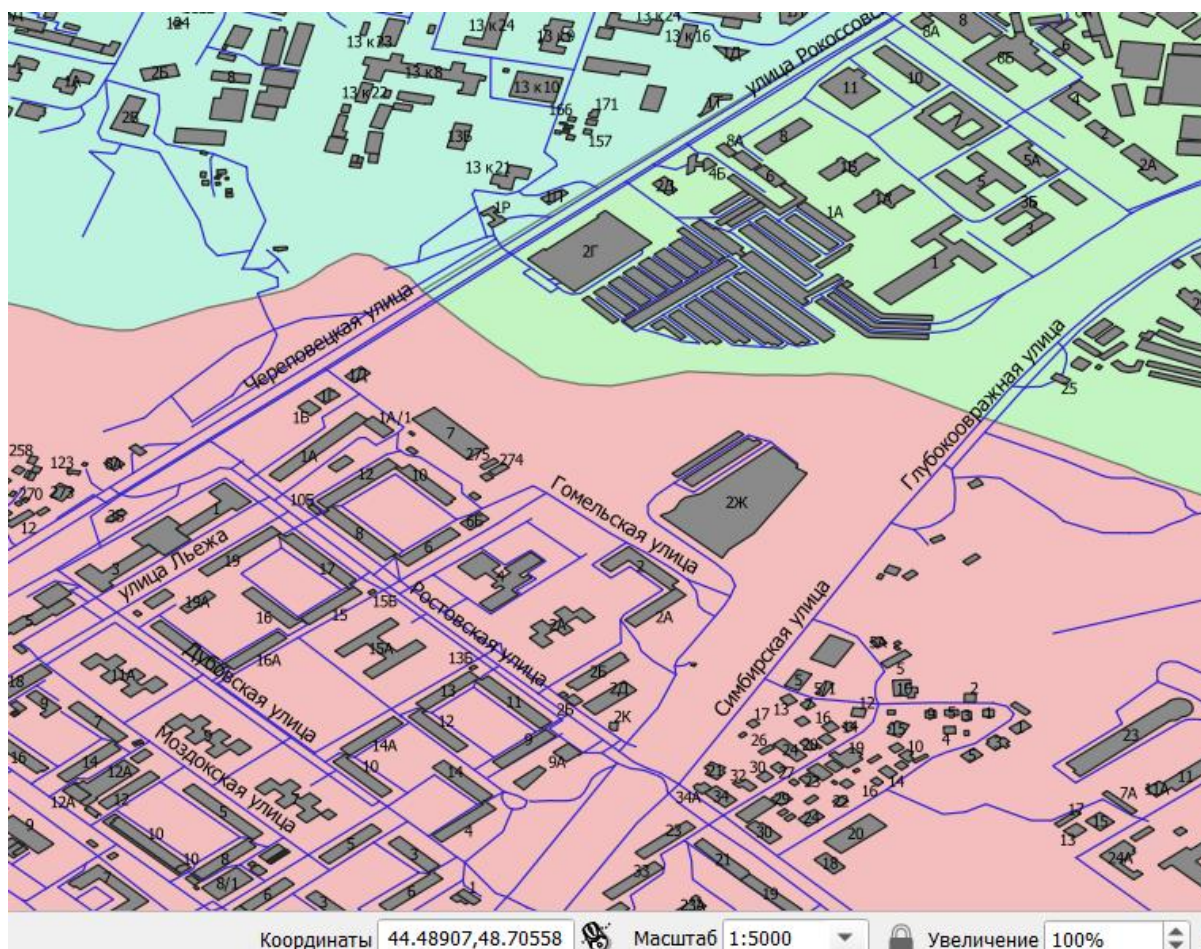


Рис. 3.4.3. Фрагмент фрейма карты после увеличения масштаба изображения до 1:5000

### ***Выбор элементов дорожной сети***

В рамках этого упражнения мы будем решать задачи логистики на примере движения автомобилей. Однако набор данных об улично-дорожной сети, предоставленный вам в качестве исходных данных, содержит не только действующие автомобильные дороги, но и другие типы объектов. Тип объекта хранится в поле «**HIGHWAY**». В табл. 3.4.1 перечислены значения атрибута «**HIGHWAY**» для тех объектов, которые не должны участвовать в анализе.

Таблица 3.4.1

#### **Значения атрибута «HIGHWAY» для объектов, которые не должны участвовать в анализе**

| Значения атрибута «HIGHWAY» | Пояснение                    |
|-----------------------------|------------------------------|
| Construction                | Строящиеся дороги            |
| Cycleway                    | Велодорожки                  |
| Footway, pedestrian         | Пешеходные дороги и тротуары |
| Path                        | Тропы                        |
| Proposed                    | Планируемые дороги           |
| Raceway                     | Гоночные треки               |
| Steps                       | Лестницы                     |

Составим запрос, чтобы выбрать в слое **RoadNetworkVolgograd** все объекты, **кроме** перечисленных выше.

Для этого выполним следующую последовательность действий (рис. 3.4.4):

1. Выберем слой **RoadNetworkVolgograd** на панели **Слои**.
2. Включим таблицу атрибутов в меню.
3. Откроем ее во встроенной панели.
4. Включим редактор для построения запросов.

5. Построим запрос.

6. Выделим объекты, удовлетворяющие условиям этого запроса.

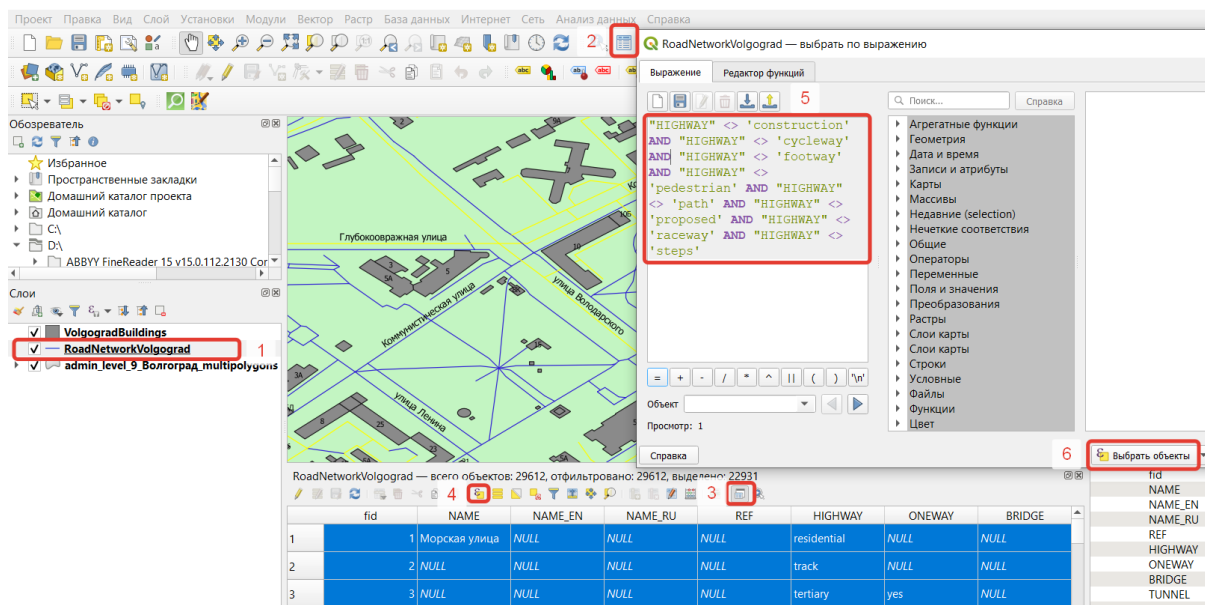


Рис. 3.4.4. Последовательность действий для создания запроса

Сохраним выборку в новый набор данных формата **GeoPackage**, как **FilteredRoadNetworkVolgograd** (правой кнопкой мыши – на слой «Экспорт...» – «Сохранить выбранные объекты как...»).

Закроем дополнительные окна и снимем выделение объектов, нажав на иконку  на панели инструментов.

Когда новый слой будет добавлен к проекту, изменим настройки его визуализации: установите толщину линий равной 0,26 мм, а цвет – любой, кроме того, который уже используется для улично-дорожной сети.

Фрагмент фрейма карты после сохранения выбранных дорог в отдельный набор показан на рис. 3.4.5.



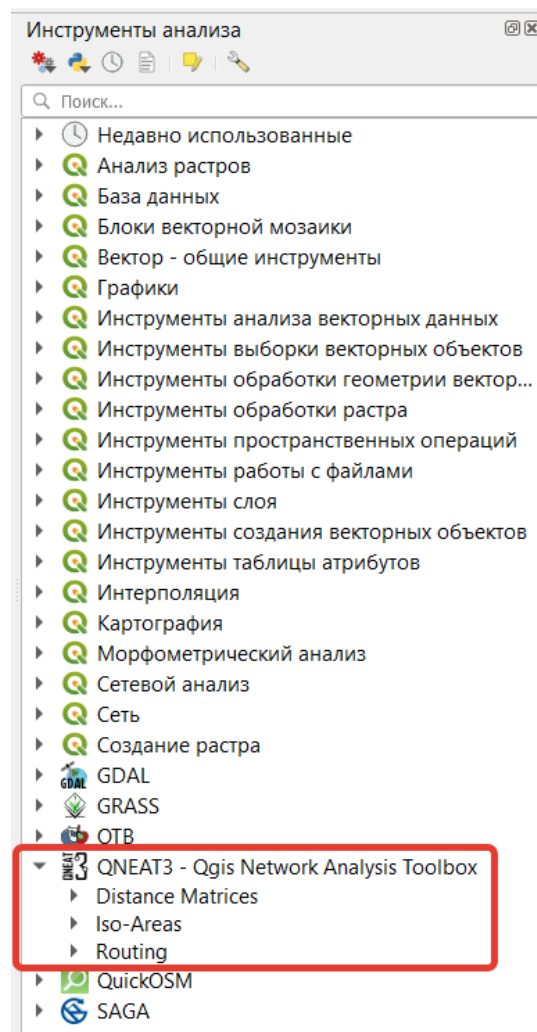


Рис. 3.4.6. Панель инструментов анализа  
с группой инструментов **QNEAT3**

Теперь построим маршрут между двумя точками. Для решения этой задачи используется инструмент **Shortest path (point to point)** из группы **Routing** набора **QNEAT3**.

Найдем здание главного корпуса ВолгГТУ (проспект Ленина, 28) и здание главного корпуса ИАиС ВолгГТУ (улица Академическая, 1). Запомним расположение этих зданий, создав пространственные закладки.

Запустим инструмент **Shortest path (point to point)**. Откроется интерфейс настройки инструмента.

В качестве слоя дорожной сети (**Network Layer**) укажем набор данных, созданный на предыдущем шаге.

Чтобы указать начальную точку маршрута, нажмем многоточие справа от поля ввода **Start point**. Интерфейс настройки инструмента будет свернут, а курсор в окне **QGIS** примет вид мишени. Найдем здание главного корпуса ВолгГТУ и щелкнем левой кнопкой мыши в произвольном месте внутри здания. Координаты курсора будут считаны и введены в поле **Start point**.

Аналогичным образом укажем конечную точку маршрута в пределах здания главного корпуса ИАиС ВолгГТУ.

Проверим, что в качестве критерия оптимизации указано **Shortest Path** (кратчайшее расстояние), а выходной набор данных сохраняется во временный файл (рис. 3.4.7).

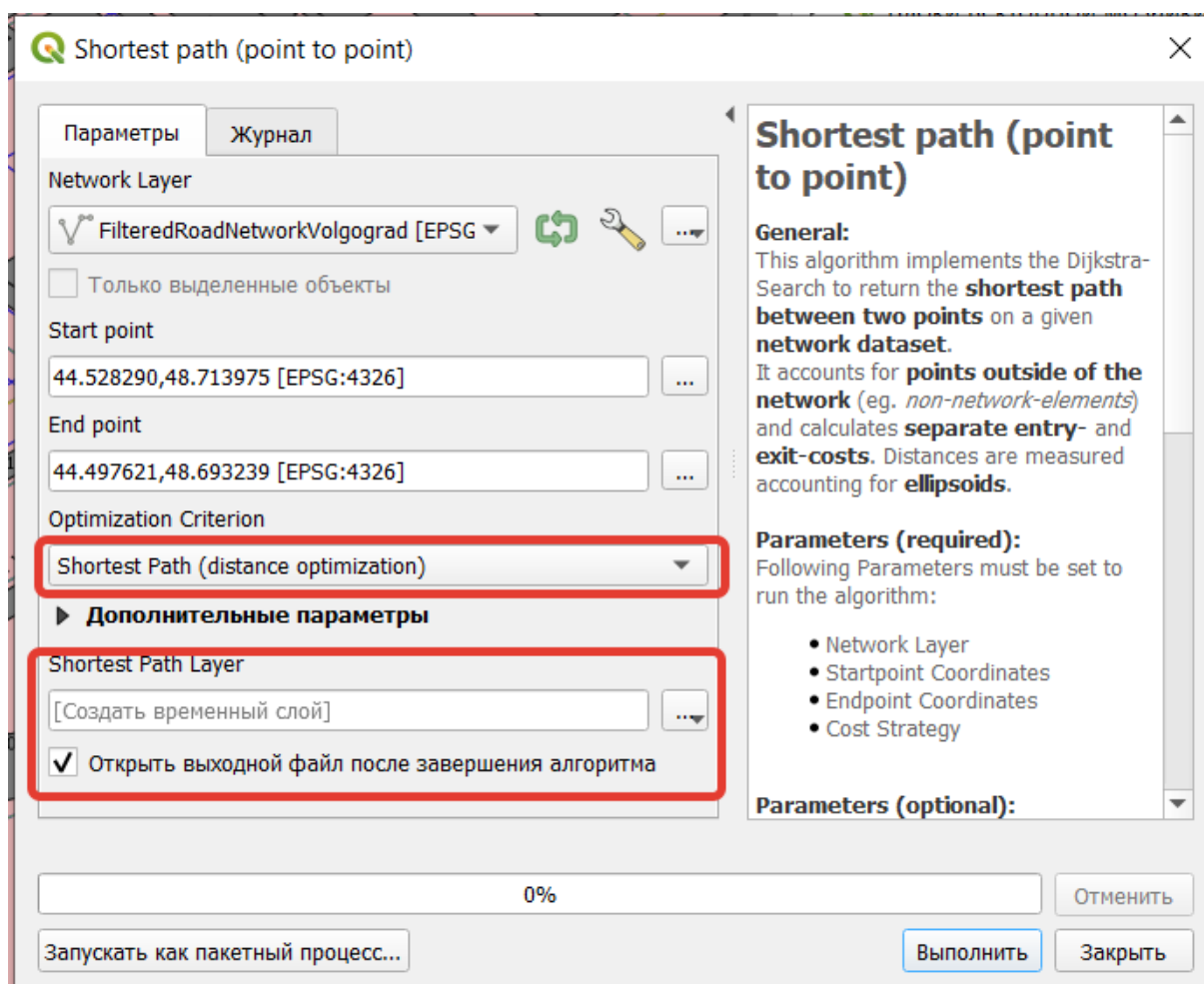


Рис. 3.4.7. Интерфейс настройки инструмента **Shortest path (point to point)**

Запустим инструмент и дождемся, пока будет рассчитан кратчайший маршрут. Интерфейс инструмента закрывать не будем, так как мы воспользуемся им снова, изменив параметры.

Построенный кратчайший маршрут появится в таблице слоев как временный слой с названием ***Shortest Path Layer***. Изменим стиль этого слоя на линию красного цвета толщиной 1 мм. Отобразим карту в охвате слоя ***Shortest Path Layer*** (рис. 3.4.8).

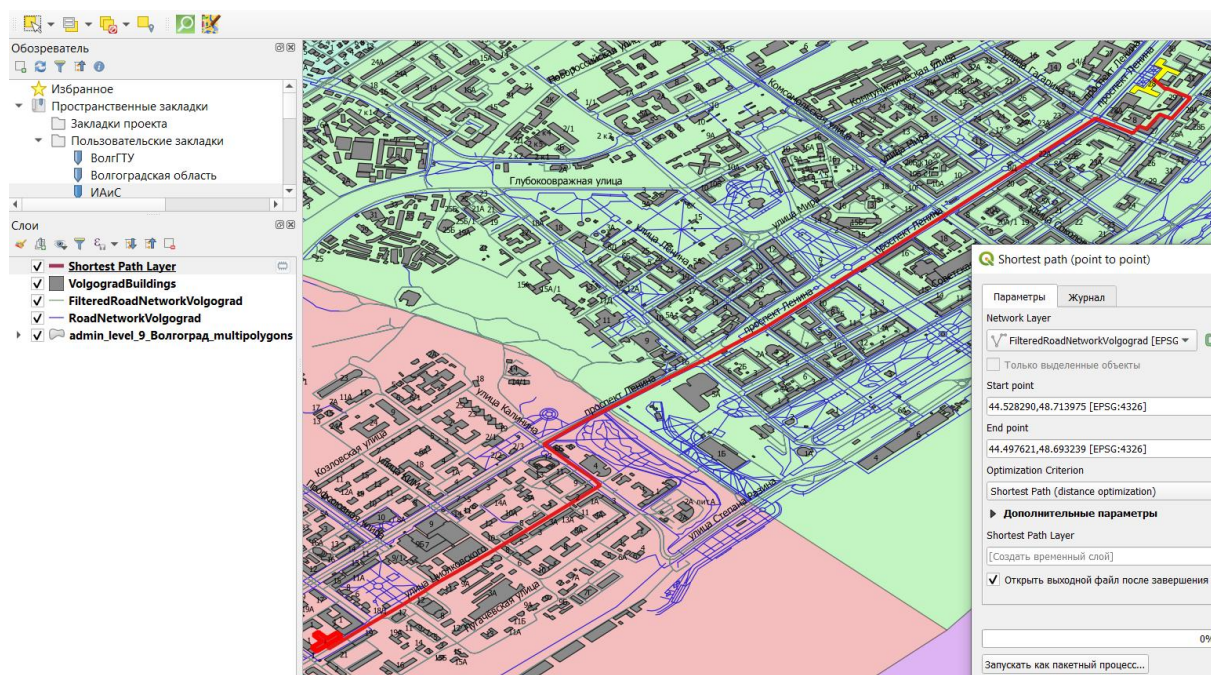


Рис. 3.4.8. Карта в охвате слоя ***Shortest Path Layer***

При построении этого маршрута учитывались только длины сегментов пути, но не учитывались характерные скорости и разрешенные направления движения. На следующих шагах мы внесем соответствующие коррективы.

Произведем реорганизацию таблицы (рис. 3.4.9), оставив только те столбцы, которые могут быть использованы для дальнейшей работы (**fid**, **HIGHWAY**, **HIGHWAY**, **ONEWAY**, **MAXSPEED**).

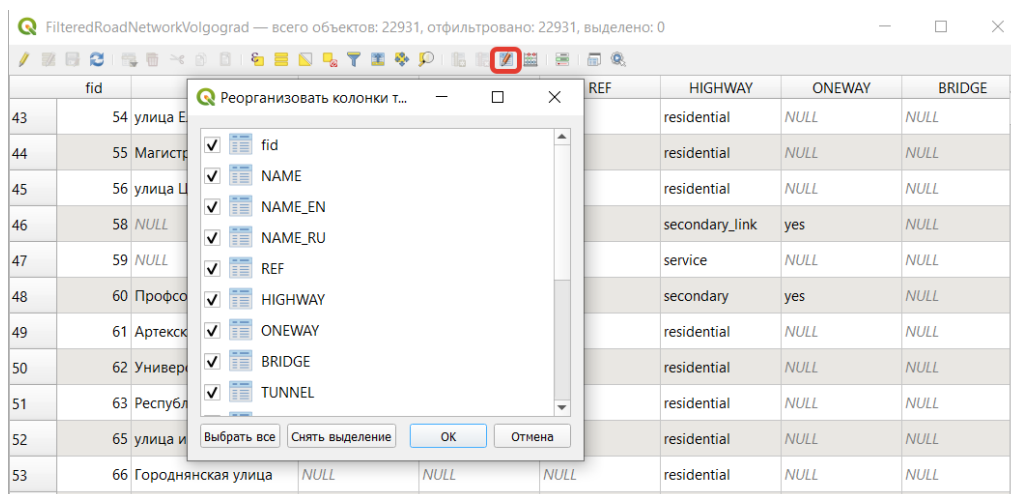


Рис. 3.4.9. Реорганизация таблицы атрибутов дорожного слоя

В столбцах **ONEWAY** и **MAXSPEED** должна содержаться информация о разрешенных направлениях движения и разрешенной скорости движения. Однако мы видим, что они заполнены далеко не везде.

Поэтому создадим два вычисляемых поля: **CalcOneway** и **CalcMaxSpeed**.

Значения поля **CalcOneway** будем вычислять на основе значений записей в поле **ONEWAY** (рис. 3.4.10).

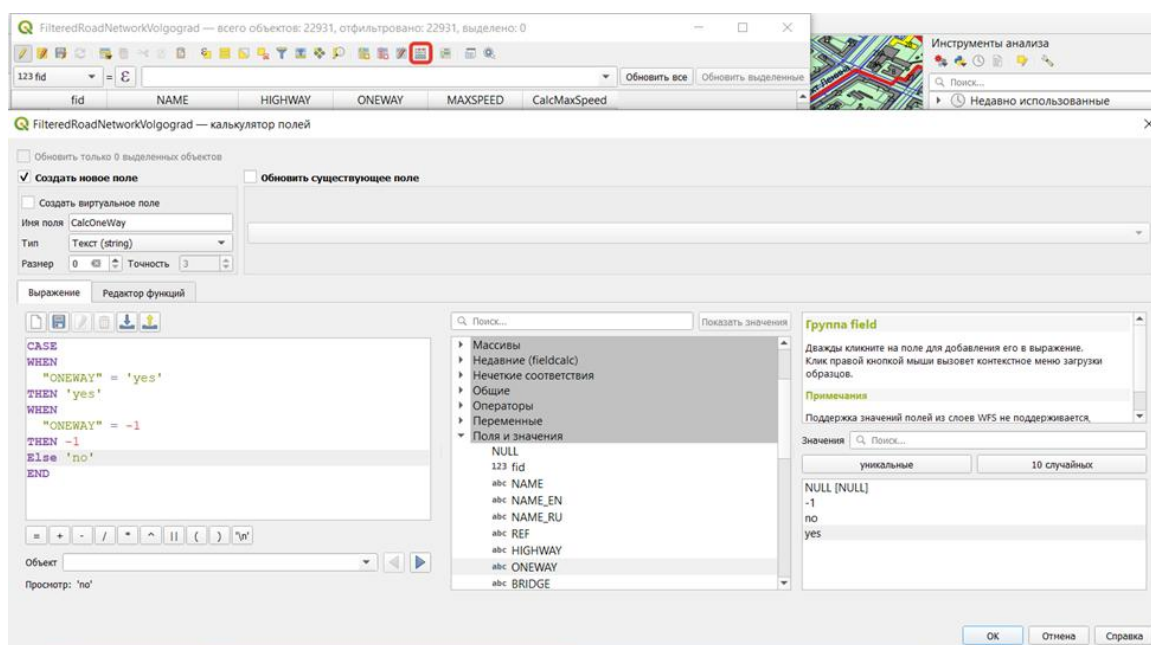


Рис. 3.4.10. Вычисление значений поля **CalcOneway**

Значения поля **CalcMaxSpeed** будем вычислять на основе значений записей в поле **HIGHWAY** и соответствующих им максимальных значений скоростей, взятых из табл. 3.4.2 (рис. 3.4.11).

Таблица 3.4.2

**Значения атрибута «HIGHWAY»  
и соответствующие им максимальные скорости**

| Значения атрибута «HIGHWAY»                                         | Максимальная скорость (км/ч) |
|---------------------------------------------------------------------|------------------------------|
| Living_street, service                                              | 10                           |
| Residential                                                         | 20                           |
| Road, tertiary, tertiary_link, track, unclassified                  | 40                           |
| Primary, primary_link, secondary, secondary_link, trunk, trunk_link | 60                           |

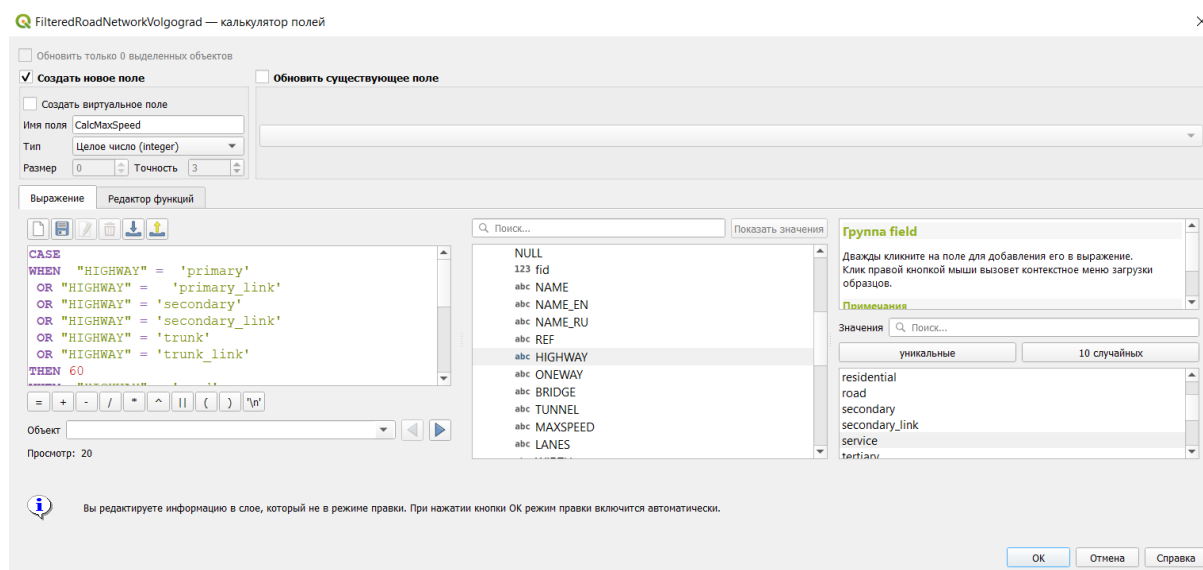


Рис. 3.4.11. Вычисление значений поля **CalcMaxSpeed**

Вернемся в интерфейс инструмента **Shortest path (point to point)** (напомним, что его не нужно было закрывать).

Изменим критерий оптимизации с кратчайшего пути на наиболее быстрый путь.

В настройке **Direction field** следует указать, в каком поле содержится информация о разрешенных направлениях движения (**CalcOneway**).

В следующих полях введем значения, соответствующие разрешенным направлениям: **yes** – для движения только в прямом направлении; **-1** – для движения только в обратном направлении; **no** – для разрешения движения в обоих направлениях.

В настройке **Speed field** укажем, из какого поля следует взять разрешенную скорость движения (**CalcMaxSpeed**).

Снова запустим расчет пути (рис. 3.4.12).

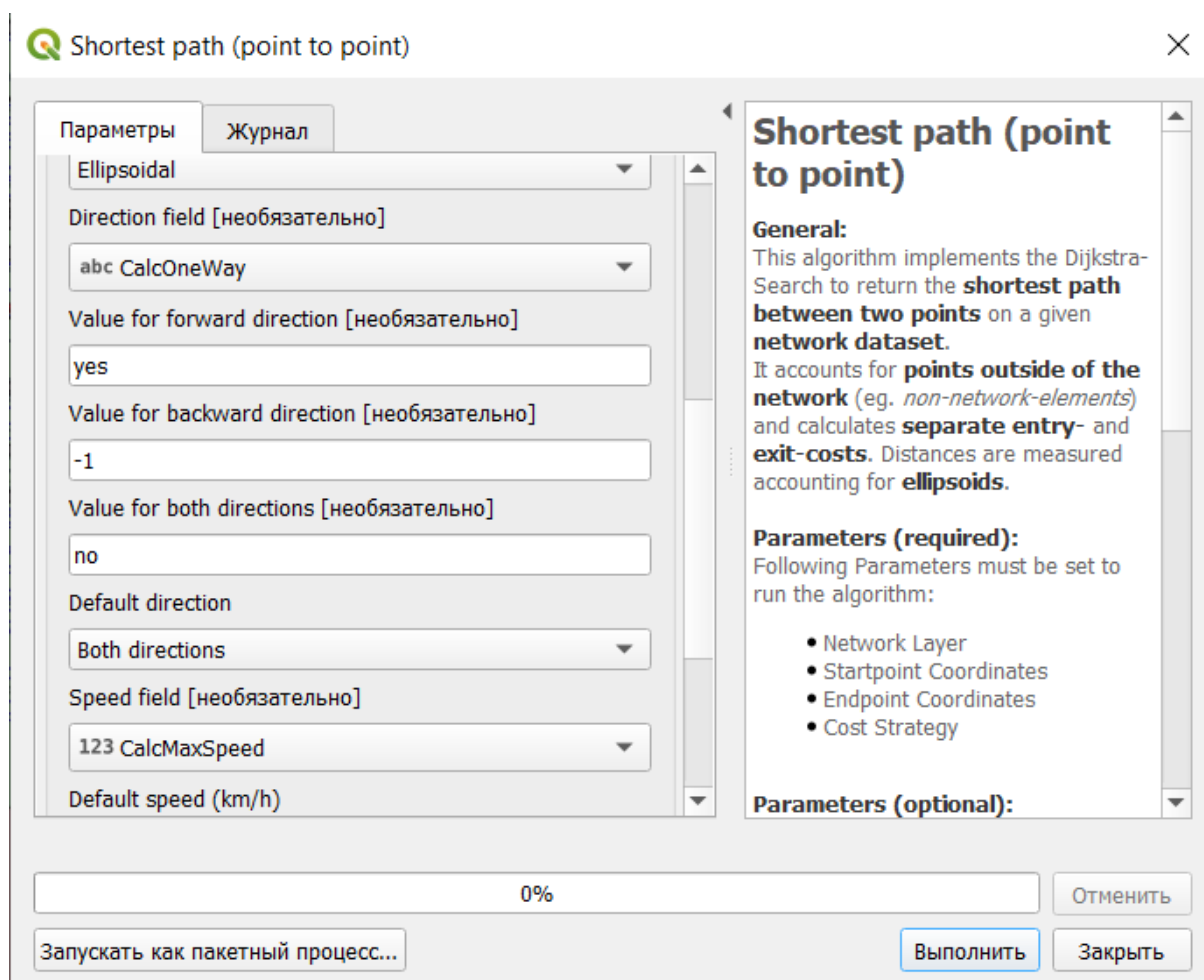


Рис. 3.4.12. Интерфейс настройки инструмента **Shortest path (point to point)** с новыми параметрами

Дождемся, пока инструмент завершит работу, и новый временный слой будет добавлен в проект. После этого окно инструмента можно закрыть.

Переименуем добавленный слой в ***Fastest Path Layer*** и изменим его отображение на линию зеленого цвета толщиной 1 мм (рис. 3.4.13).



Рис. 3.4.13. Карта со слоями ***Shortest Path Layer*** и ***Fastest Path Layer***

## Задания

Самостоятельно повторите расчет маршрутов по кратчайшему расстоянию и скорейшему времени для пары «источник – назначение», взяв в качестве «источника» вокзал Волгоград-1, а в качестве «точки назначения» – место вашего проживания. Сохраняйте построенные пути в «постоянные» наборы данных.

### **Вопросы к отчету**

1. Какие методы пространственного анализа данных вы знаете?
2. Какие методы сетевого анализа используют в геоинформатике?
3. Как можно представить дорожную сеть в **QGIS**?
4. Опишите процесс построения маршрута в **QGIS**.

## Лабораторная работа № 5

### «Трехмерная визуализация в ГИС»

**Цель работы:** Получение навыков работы с программным обеспечением для трехмерной визуализации данных в ГИС на примере отображения рельефа местности.

Трехмерная визуализация может использоваться для изучения рельефа и его морфометрических характеристик; моделирования полей распределения физических и химических показателей в океане, атмосфере и почвах; моделирования городской среды и транспортных коммуникаций и прочих задач.

Рассмотрим технологию создания трехмерных моделей на примере отображения рельефа. Данные о рельефе преимущественно хранятся в растровом формате в виде спутников фотоснимков с информацией о высотности. DEM (digital elevation model) является цифровой моделью поверхности рельефа. Наличие дополнительного измерения в рамках трехмерной визуализации позволяет более эффективно представлять данные, имеющие несколько параметров в рамках одной карты. Однако при использовании объемных визуализаций следует учесть особенности рельефа изучаемой области, либо отказаться от отображения рельефных особенностей территории.

Одним из источников данных о рельефе является ресурс USGS EarthExplorer (<http://earthexplorer.usgs.gov>), который предоставляет информацию для исследований бесплатно. Для работы с этим ресурсом необходимо зарегистрироваться.

Для получения DEM файла необходимо выбрать интересующую вас область; примеры окон представлены на рис. 3.5.1–3.5.3. Для отображения данного формата через QGIS Desktop понадобится плагин QGIS2Threejs.

## 1. Enter Search Criteria

To narrow your search area: type in an address or place name, enter coordinates or click the map to define your search area (for advanced map tools, view the [help documentation](#)), and/or choose a date range.

**Geocoder** | KML/Shapefile Upload

Select a Geocoding Method  
 Address/Place ▼

**Search Limits:** The search result limit is 100 records; select a Country, Feature Class, and/or Feature Type to reduce your chances of exceeding this limit.

US Features | **World Features**

Feature Name  
 (use % as wildcard)

State  
 All ▼

Feature Type  
 All ▼

Show Clear

---

Polygon | **Circle** | Predefined Area

*Circular polygons are created using a center point defined by decimal latitude and longitude values and a radius.*

Center Latitude: 49.49239300333961 Center Longitude: 43.570854818510384

Radius: 50 Kilometers ▼

Apply Clear Circle

Рис. 3.5.1. Установка критериев поиска области исследования

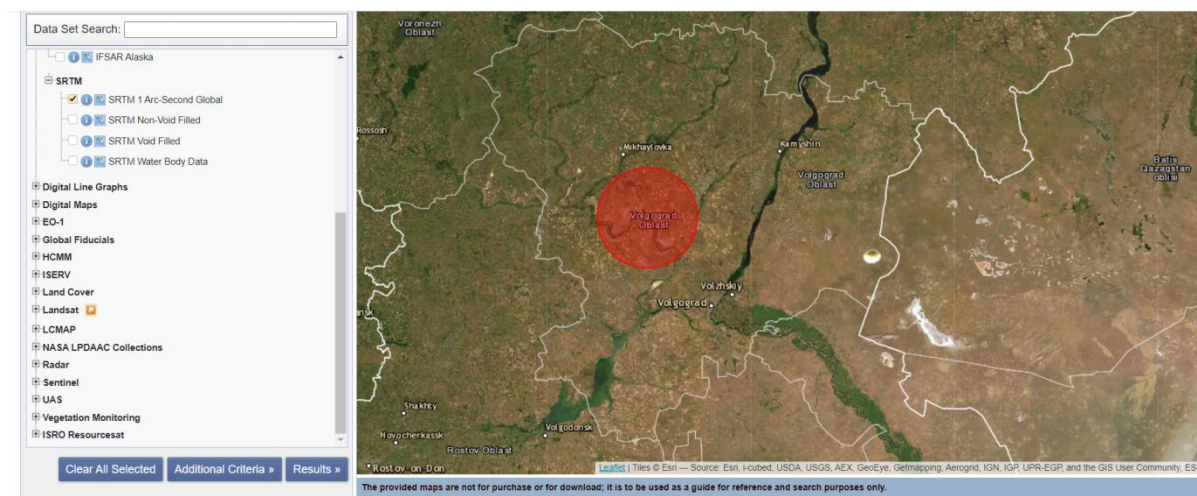


Рис. 3.5.2. Установка фильтров по типу карт для выбора SRTM



Рис. 3.5.3. Выбор датасета для трехмерной визуализации данных

На рис. 3.5.4–3.5.5 показано, как установить плагины; на рис. 3.5.6–3.5.7 – как добавить дефолтную карту, к которой будет привязываться карта SRTM.

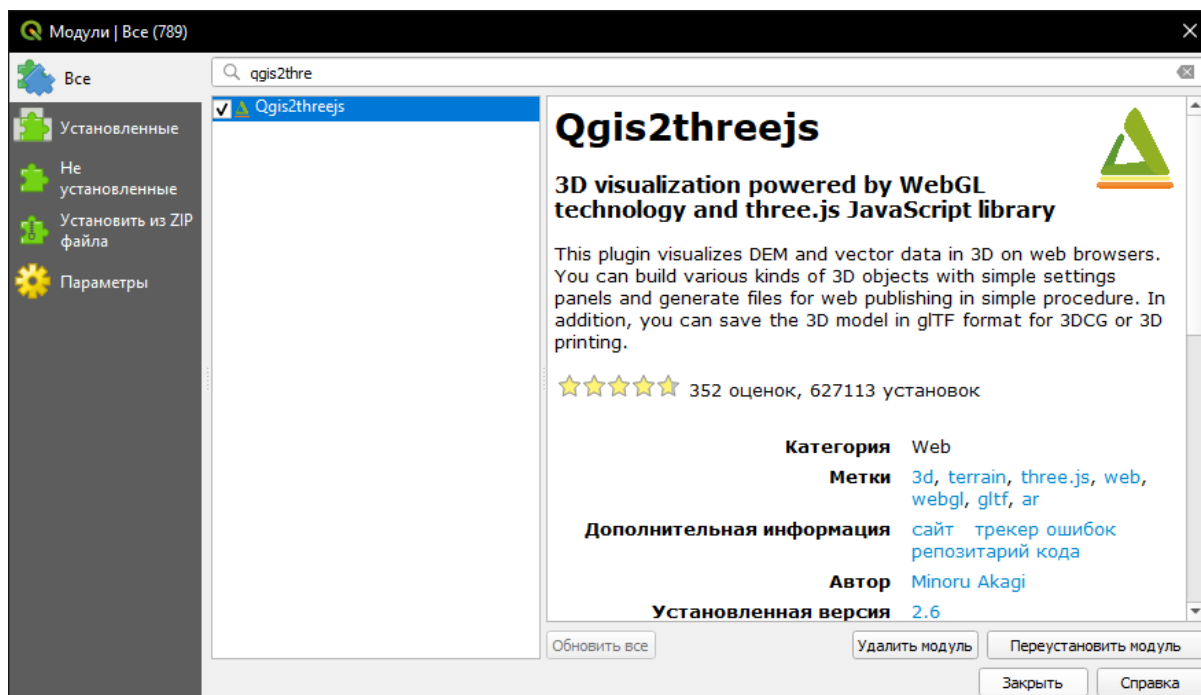


Рис. 3.5.4. Установка плагина Qgis2threejs

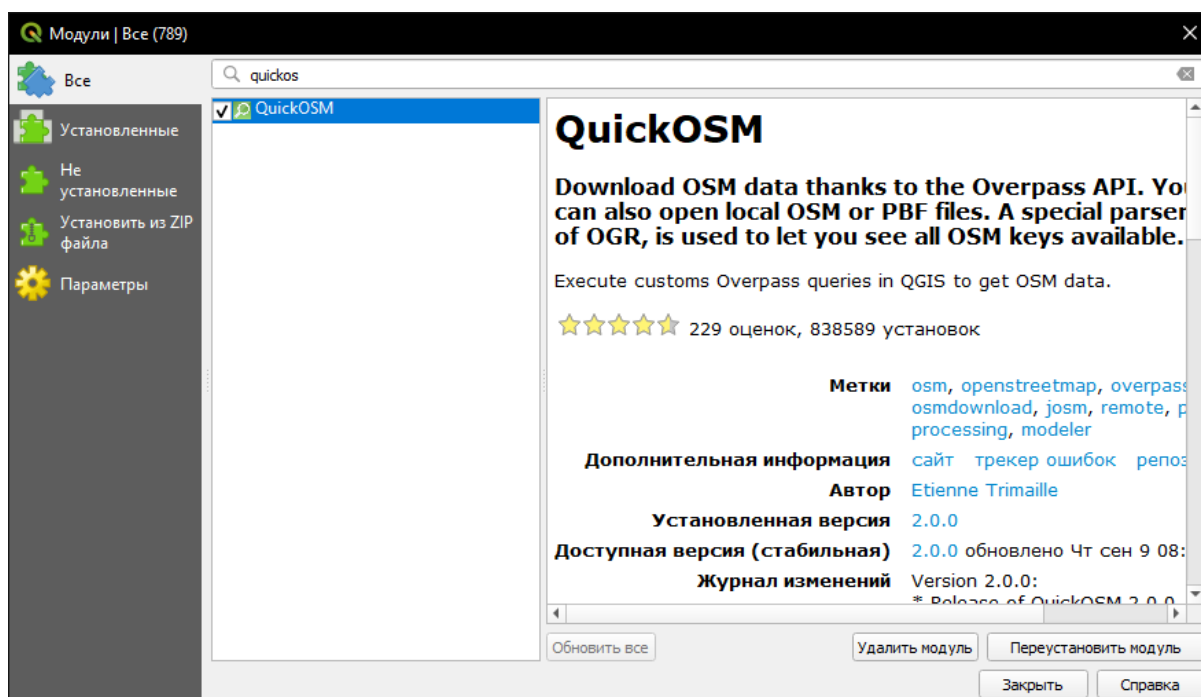


Рис. 3.5.5. Установка плагина QuickOSM

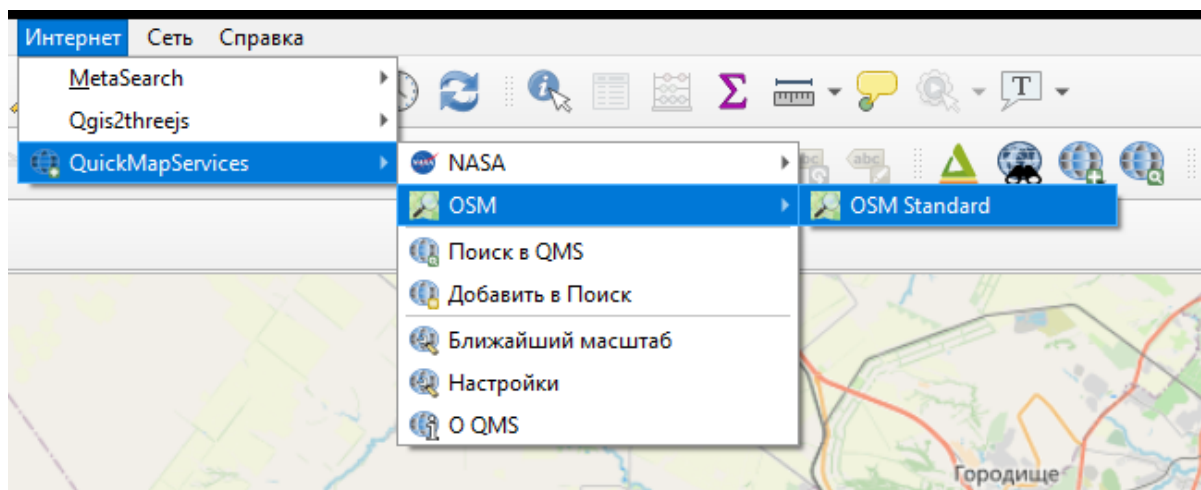


Рис. 3.5.6. Добавление на канвас дефолтной карты

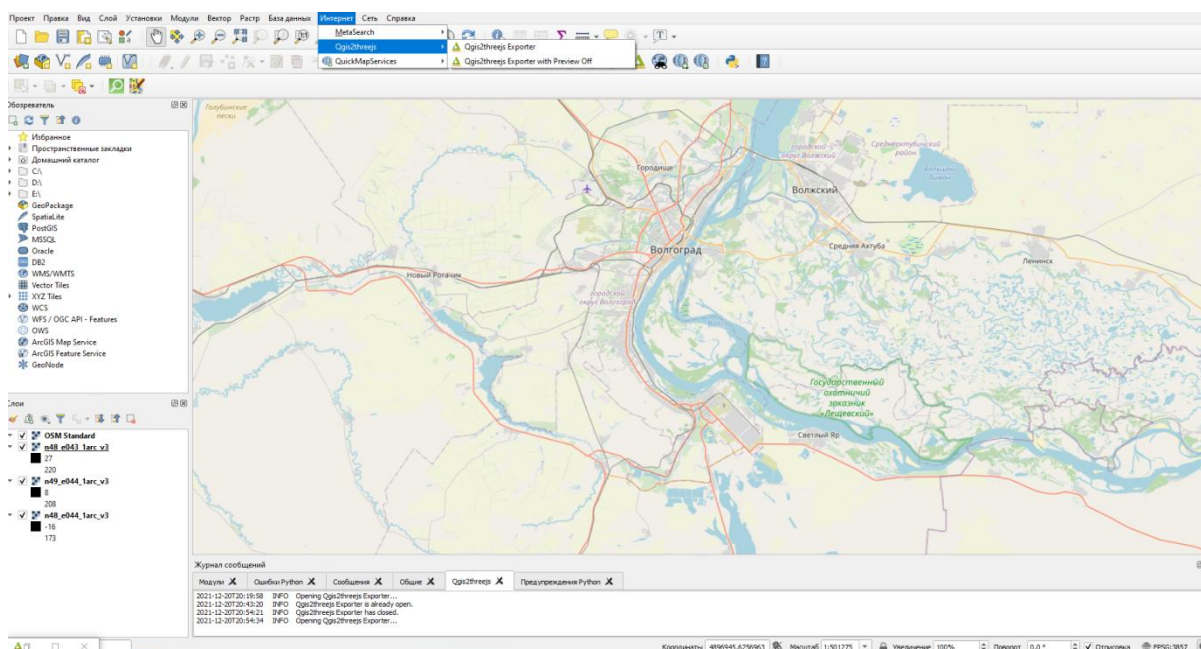


Рис. 3.5.7. Карта из плагина QuickOSM

После вывода дефолтной карты необходимо добавить карту рельефа. Для этого перенесем файл с картой в проект. Далее перейдем в плагин Qgis2threejs и изменим настройки так, чтобы рельеф совпадал с действительными пропорциями местности. На рис. 3.5.8 представлены настройки, используемые для файлов с сайта <https://earthexplorer.usgs.gov/>; данные на-

стройки могут отличаться при выборе другого датасета. На рис. 3.5.9 представлена получившаяся карта, которая отображает рельеф местности.

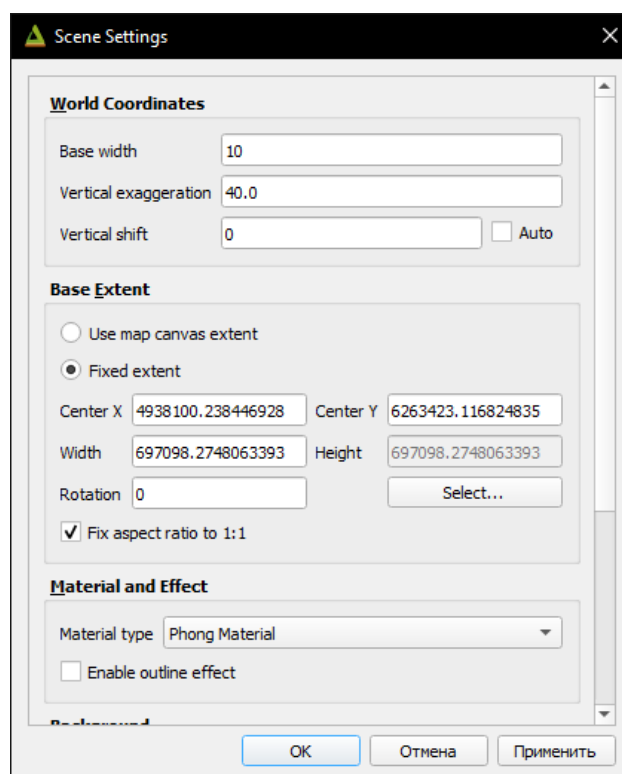


Рис. 3.5.8. Настройка плагина Qgis2threejs

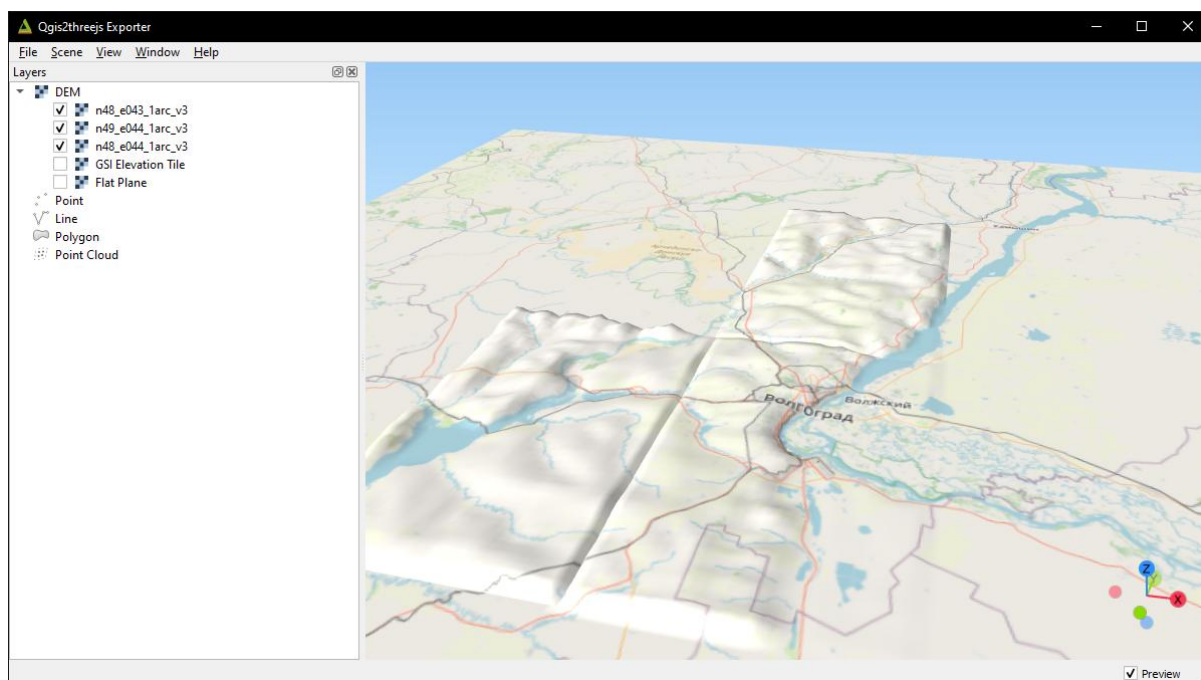


Рис. 3.5.9. Визуализация рельефа местности

Чтобы иметь возможность демонстрировать проделанную работу вне зависимости от того, установлен QGIS или нет, необходимо экспортировать полученную карту в web. Для этого выберем «Export to Web». После этого выберем директорию, в которую следует выгрузить необходимые файлы, и нажмем Export (рис. 3.5.10).

При экспорте мы получим сгенерированную папку, которая определяет web-страницу; при открытии файла с расширением .html мы перейдем в браузер, в котором сможем детальнее рассмотреть полученную карту. На рис. 3.5.11 отображена карта в web-версии.

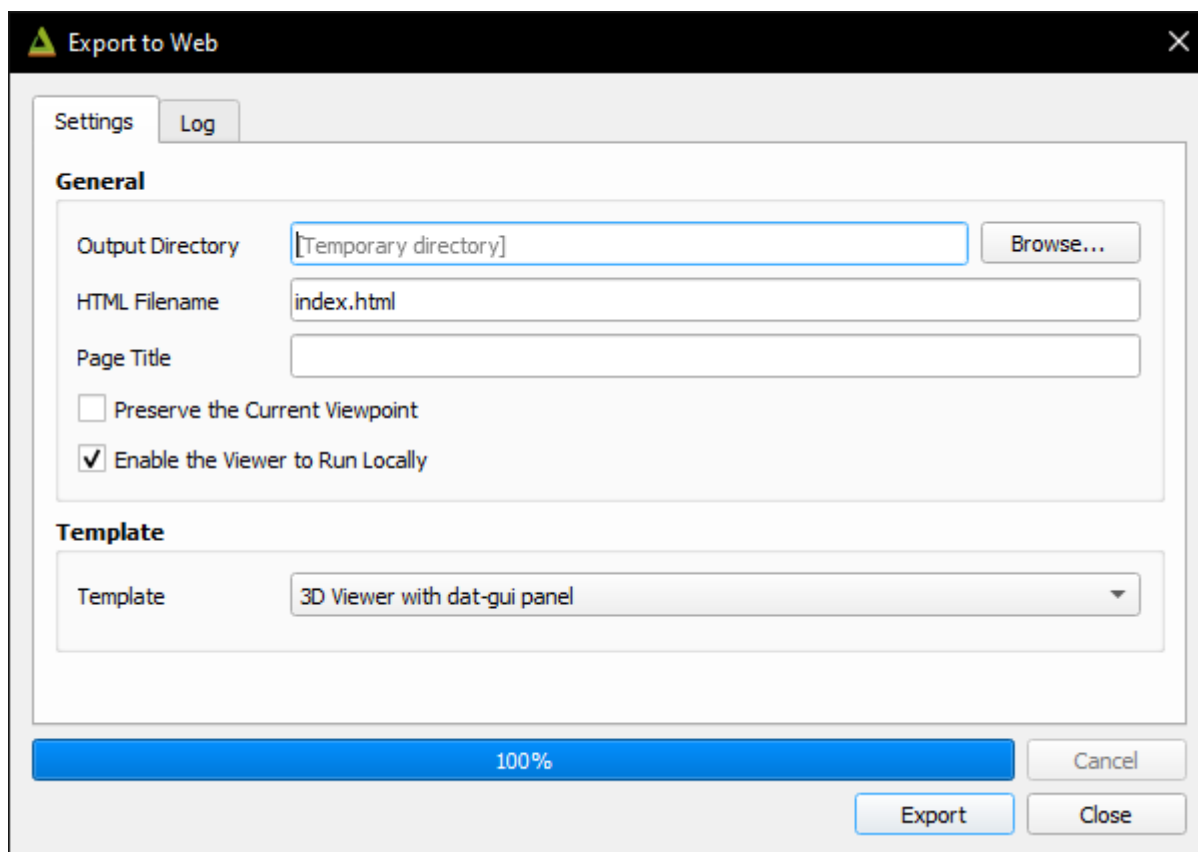


Рис. 3.5.10. Экспорт в web-версию



Рис. 3.5.11. Визуализация рельефа местности в web-версии

### Задания

Подготовьте трехмерную карту любого района или ограниченной территории (например, острова) Волгоградской области.

### Вопросы к отчету

1. Что такое DEM-формат? Что будет если открыть файл такого формата в графическом редакторе?
2. В чем преимущества и недостатки трехмерной визуализации данных?
3. В каких случаях использование трехмерной визуализации является целесообразным? Приведите не менее трех примеров.

Лабораторная работа № 6  
«Получение данных пространственного анализа  
на основе дистанционного зондирования»

**Цель работы:** получение навыков сбора данных для пространственного анализа на основе дистанционного зондирования Земли (ДЗЗ).

Методы космической съемки позволяют регулярно получать изображения больших участков Земли, которые можно использовать в качестве фонового слоя для векторных данных. Космические снимки используются для планирования городской инфраструктуры и развития транспортных систем, мониторинга состояния окружающей среды, для борьбы с природными катастрофами, повышения эффективности сельскохозяйственного производства и прочих целей.

Цифровые данные, полученные с помощью искусственных спутников, после небольшой предварительной обработки можно сразу же использовать для создания базы данных ГИС.

*Программные средства*

Python (python.org) – высокоуровневый язык программирования, обладающий высокой производительностью, при этом являясь простым и удобочитаемым. Широко применяется при операциях с данными, что и является целью этой учебной практики.

Keras (keras.io) – открытая нейросетевая библиотека, написанная на языке Python. Она представляет собой надстройку над фреймворками DeepLearning4j, TensorFlow и Theano. Нацелена на оперативную работу с сетями глубинного обучения; при этом спроектирована так, чтобы быть компактной, модульной и расширяемой.

TensorFlow (tensorflow.org) – открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и тренировки нейронной сети с целью автоматического нахож-

дения и классификации образов, достигая качества человеческого восприятия. Применяется как для исследований, так и для разработки собственных продуктов Google. Основное API для работы с библиотекой реализовано для Python.

Numpy (numpy.org) – библиотека с открытым исходным кодом для языка программирования Python. Возможности: поддержка многомерных массивов (включая матрицы); поддержка высокоуровневых математических функций, предназначенных для работы с многомерными массивами.

OpenCV (opencv.org) – библиотека компьютерного зрения с открытым исходным кодом, включающая алгоритмы компьютерного зрения, обработки изображений и численные алгоритмы общего назначения. Реализована на C/C++, также разрабатывается для Python, Java, Ruby, Matlab, Lua и других языков.

### *Ход работы*

Результатом работы является получение сгенерированного нейронной сетью изображения сегментированных участков анализируемой территории или определенных типов объектов на основе исходного снимка местности, полученного со спутника или беспилотного летательного аппарата (БПЛА). В качестве исходных изображений можно использовать онлайн-карты (слой спутниковых снимков), спутниковые снимки научных программ ДЗЗ (Landsat, Sentinel и др.), свободно распространяемые датасеты изображений БПЛА.

Перед реализацией нейронной сети необходимо подготовить данные, для того чтобы обучить нейросетевой классификатор. Входными данными являются изображения формата PNG/JPEG. На вход программе подается директория с имеющимися изображениями. Подготовка изображений выполняется в определенной последовательности:

- 1) отбираются изображения (делаются скриншоты) участков территории, содержащие необходимые для классификации объекты (рис. 3.6.1);
- 2) изображения форматируются до одного размера;
- 3) выполняется разметка изображений (фрагменты изображения на отдельном слое выделяются различными цветами для разных типов определяемых объектов) с сохранением получившейся цветовой маски (рис. 3.6.2);
- 4) все изображения разделяются на обучающую и тестовую выборки.



Рис. 3.6.1. Пример исходного изображения

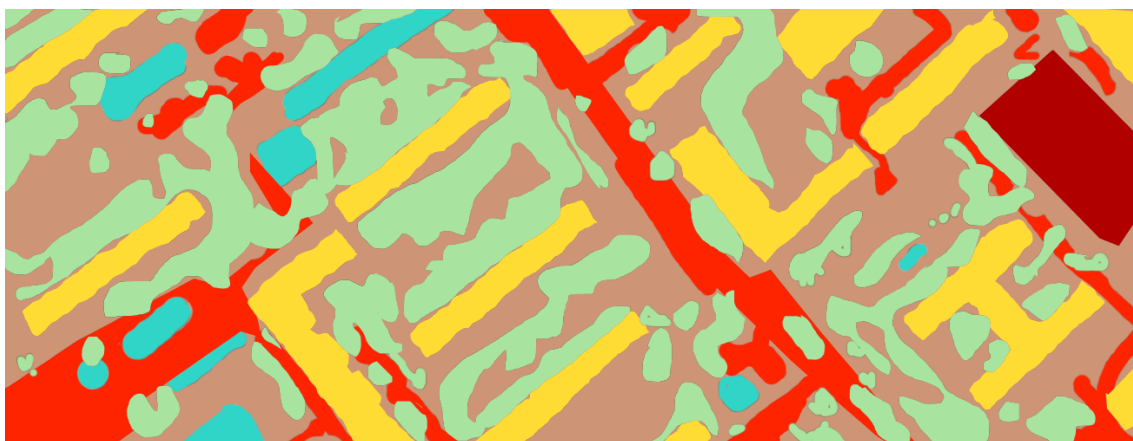


Рис. 3.6.2. Пример разметки исходного изображения на рис. 3.6.1

Исходные изображения должны соответствовать следующим требованиям:

- изображения участков территории должны отбираться со снимков в едином масштабе посредством создания скриншотов;

- фрагменты одного класса должны быть в равном количестве получены со снимков разных территорий (разнообразие географической выборки);

- выборка территорий (при наличии доступных данных) должна быть сформирована из населенных пунктов, равномерно распределенных по разной широте и долготе;

- фрагменты одного класса должны включать максимальное разнообразие состояний, представленное на снимках:

- а) для природных классов – разные сезоны (с листвой, без листвы, переходные состояния цвета), виды растений (хвойные, лиственные, кустарники, камыши, разнотравие, газоны), места произрастания (равнина, склоны, городская застройка разной плотности, парки), массовость (одиночные, массивы) и т. д.;

- б) для антропогенных классов – разные виды (жилая застройка, промышленная; дороги, инфраструктура), сочетания (комбинация объектов разного типа), массовость (одиночные, массивы), плотность и т. д.;

- однотипное содержание фрагментов одного класса должно быть в равной степени представлено на снимках разного периода и качества съемки;

- объем набора данных каждого класса не должен отличаться от любого другого набора более чем на 30 %.

Полученные изображения подаются на обучение нейросети. На основе предсказанных нейросетью данных программа преобразует изобра-

жения. Полученные изображения сохраняются в директорию, указанную пользователем (рис. 3.6.3).

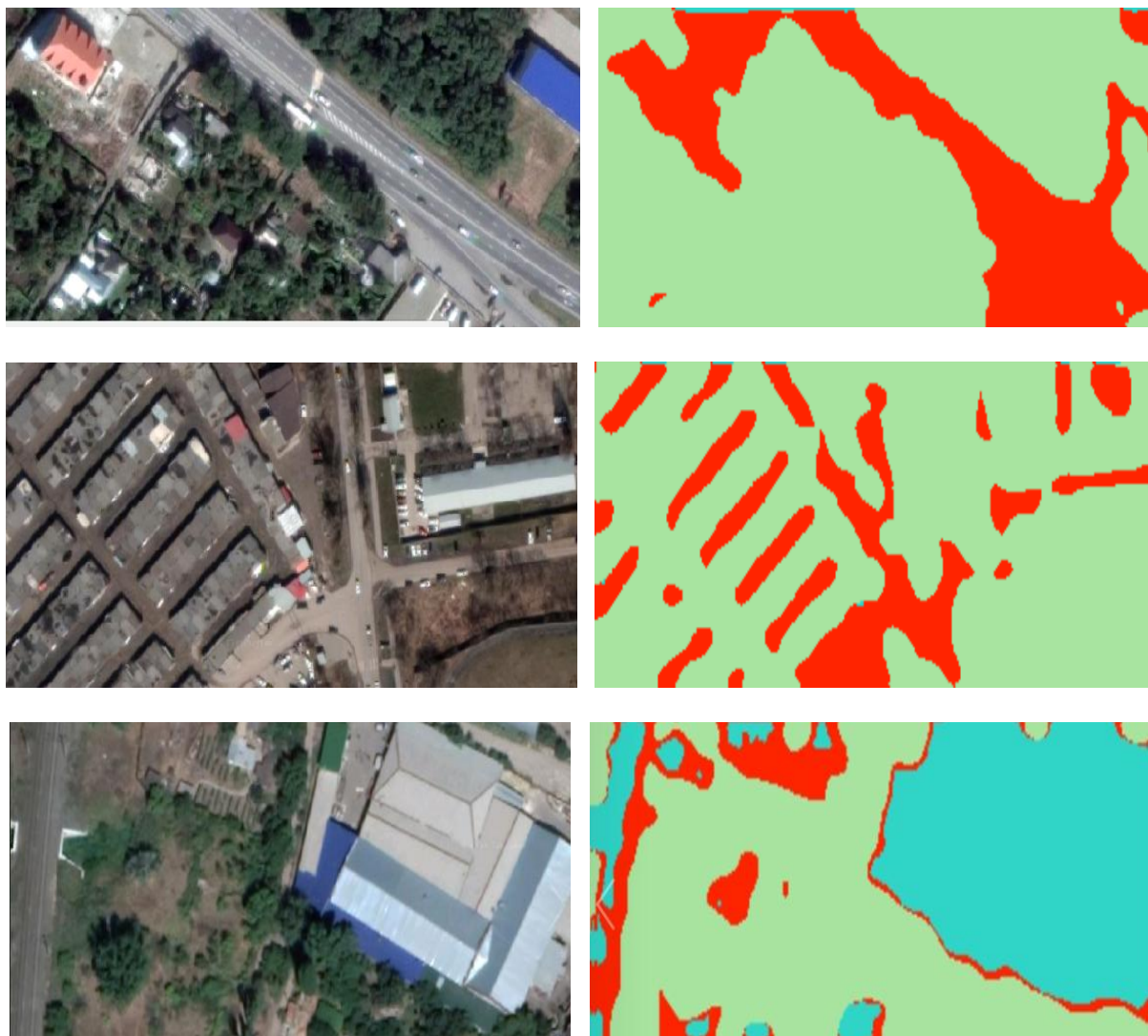


Рис. 3.6.3. Примеры результатов сегментации изображений местности

### Задания

Самостоятельно проведите обучение нейронной сети для определения одного из видов объектов территории:

1. Автомобильные дороги.
2. Железные дороги.
3. Парковка автотранспорта.
4. Деревья.

5. Водоемы.
6. Многоэтажные дома.
7. Места складирования (свалки) отходов.
8. Спортивные поля.
9. Гаражи.
10. Прочие территории по заданию преподавателя.

### **Вопросы к отчету**

1. Какие источники данных для автоматического построения базы данных ГИС вы знаете?
2. Какие программные средства применяются для распознавания изображений со спутников/БПЛА?
3. Опишите процесс подготовки данных ДЗЗ для автоматического определения объектов территории.

---

---

## МЕТОДИЧЕСКИЕ УКАЗАНИЯ К КОНТРОЛЬНОЙ РАБОТЕ

Целью выполнения контрольной работы является овладение современными технологиями, методами и средствами создания и использования информационных систем, ориентированных на интеллектуальный анализ пространственных (географических) данных в процессе поддержки принятия решений; освоение основных моделей, методов и инструментальных средств, используемых для работы с картографической информацией на основе технологий искусственного интеллекта, а также способов представления пространственных данных и практических умений и навыков разработки проектов с использованием пространственных данных.

В процессе выполнения контрольной работы студенты получают опыт применения ГИС-технологий для решения прикладных задач с использованием системы моделирования OSMLS [23–26].

### **4.1. Настройка и запуск системы моделирования OSMLS**

Для запуска инфраструктуры системы моделирования требуется:

- 1) установить Docker;
- 2) переключить Docker на работу с linux контейнерами (должно быть по-умолчанию);
- 3) установить Git;
- 4) клонировать репозиторий основной программы, используя команду «git clone --branch 21.03b <https://bitbucket.org/BUROVIC/osmlifesimulation>»;

5) выполнить переход в директорию программы командой «`cd osmlifesimulation`»;

6) в файле `Dockerfile` (в последней строке) клонированного репозитория изменить путь к файлу карты на любой другой, чтобы изменить участок карты, на котором будет происходить моделирование (опционально);

7) создать `docker image`, выполнив команду: «`docker build --pull -t osmls .`»;

8) запустить `docker` контейнер с помощью команды «`docker run --rm -it -p 80:80 osmls`» (для выбора другого порта требуется указать, например `8000:80`);

9) перейти в браузере по адресу «`http://localhost/index.html`»; может понадобиться также указать порт, если он был изменен на предыдущем шаге;

10) для сборки модулей требуется скачать и установить `.NET Core SDK` (версия 3.1 LTS рекомендуется для создания новых модулей);

11) выполнить сборку требуемых модулей из исходных файлов стандартными средствами `.NET`.

## **4.2. Создание и запуск тестового модуля**

Для создания модулей можно использовать любую IDE или редактор кода, например: `JetBrains Rider`, `Visual Studio` или `Visual Studio Code`.

В данном примере создание модуля будет продемонстрировано в среде `Visual Studio 2019` (в 2017-й версии также не должно возникнуть проблем). Перед созданием модуля необходимо убедиться, что был установлен `.NET Core SDK`. Также в `Visual Studio Installer` может понадобиться добавить пункт «Кроссплатформенная разработка `.NET Core`» (см. рис. 4.2.1), если шаблон проекта «`Class Library (.NET Core)`» отсутствует при его создании.

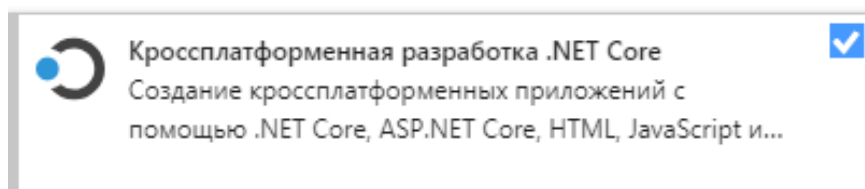


Рис. 4.2.1. Добавление необходимых компонентов в Visual Studio Installer

Для создания модуля требуется произвести следующую последовательность действий:

1. Создать новый проект типа Class Library (.NET Core), как показано на рис. 4.2.2.

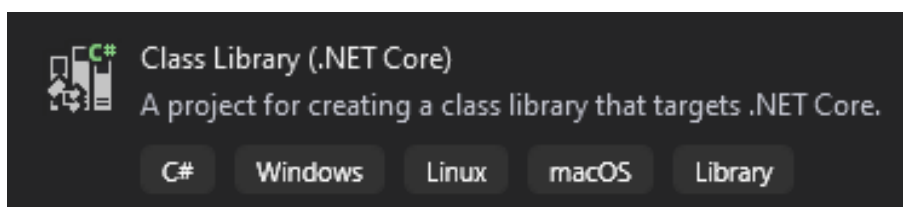


Рис. 4.2.2. Добавление необходимых компонентов в Visual Studio Installer

2. Задать произвольное имя проекта (в данном примере будет использоваться имя TestModule), как показано на рис. 4.2.3.

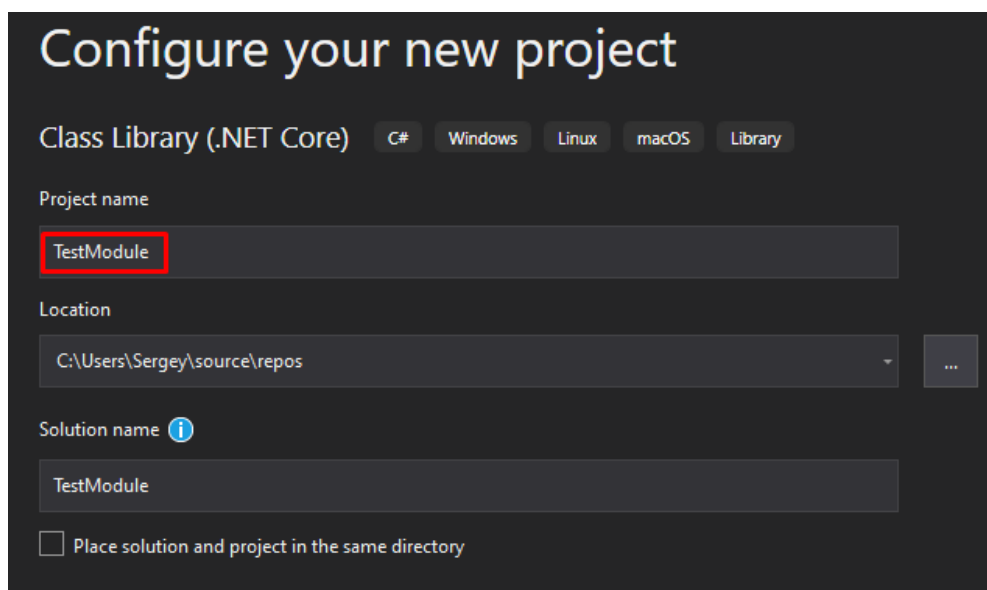


Рис. 4.2.3. Назначение имени проекта

3. Получить файл сборки глобальной библиотеки, выполнив команду «dotnet publish ./OSMLSGlobalLibrary -c release -o ./release» в корневой директории системы моделирования (osmlifesimulation); файл OSMLSGlobalLibrary.dll будет создан в директории release.

4. Перенести файл сборки в директорию проекта модуля.

5. Добавить в созданный проект модуля зависимость на файл OSMLSGlobalLibrary.dll, как показано на рис. 4.2.4 и 4.2.5.

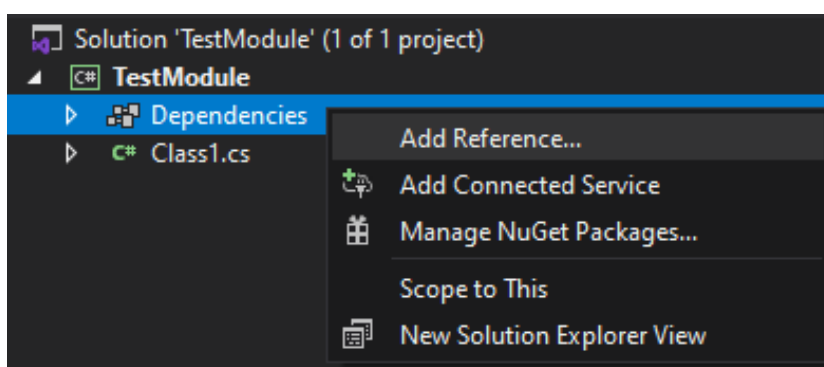


Рис. 4.2.4. Добавление зависимости в проект

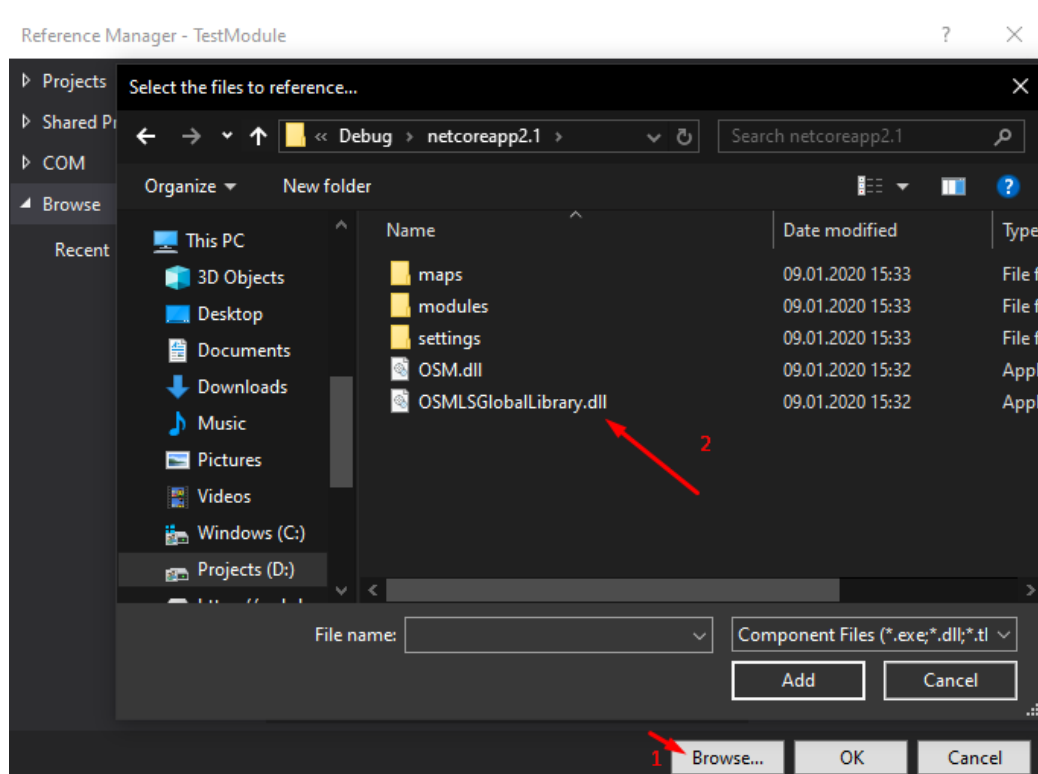


Рис. 4.2.5. Выбор добавляемой в проект зависимости

6. В консоли, находясь в директории проекта, выполнить команды «dotnet add TestModule package NetTopologySuite --version 2.0.0» и «dotnet add TestModule package NetTopologySuite.Features --version 2.0.0», чтобы установить библиотеку NetTopologySuite, необходимую для работы со всеми геометрическими объектами в проекте.

7. Переименовать созданный по умолчанию класс Class1.cs в TestModule.cs, как показано на рис. 4.2.6 и 4.2.7.

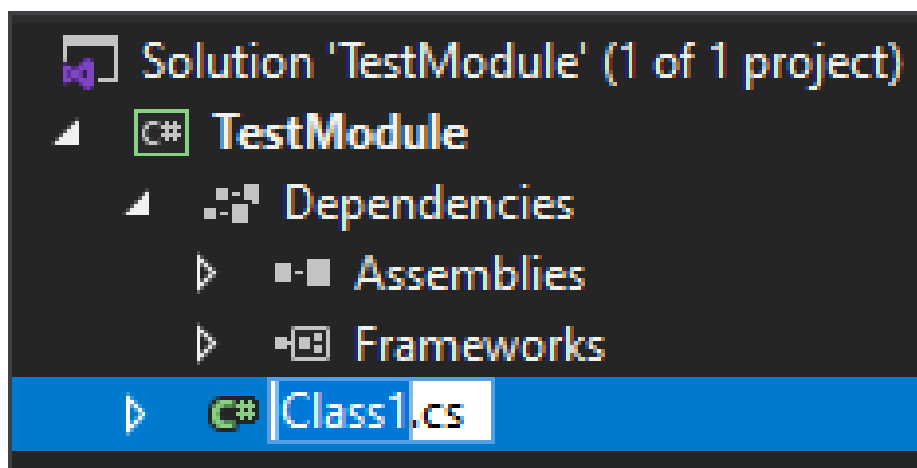


Рис. 4.2.6. Переименование файла класса

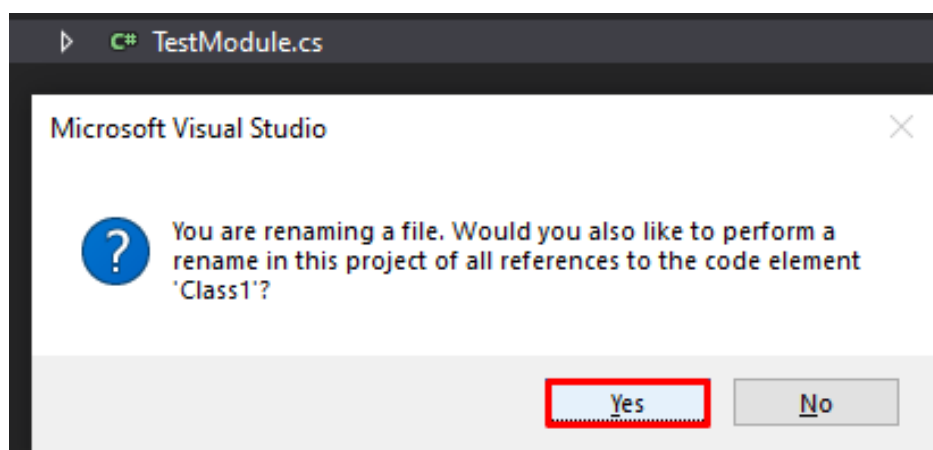


Рис. 4.2.7. Подтверждение переименования

8. Разместить в созданном классе код тестового модуля, доступный по ссылке <https://bitbucket.org/osmlifesimulation/testmodule/src/21.03b/TestModule/TestModule.cs>; проанализировать код.

9. Выполнить сборку тестового модуля с помощью команды «dotnet publish -c release -o release» и найти в директории release файл сборки (TestModule.dll).

10. Добавить файл тестового модуля в систему моделирования OSMLS (с помощью кнопки Add Assemblies и выбора модуля из выпадающего списка модулей, как показано на рис. 4.2.8) и перезапустить ее (нажатием на кнопку Stop, а затем – на кнопку Run); проанализировать результат.

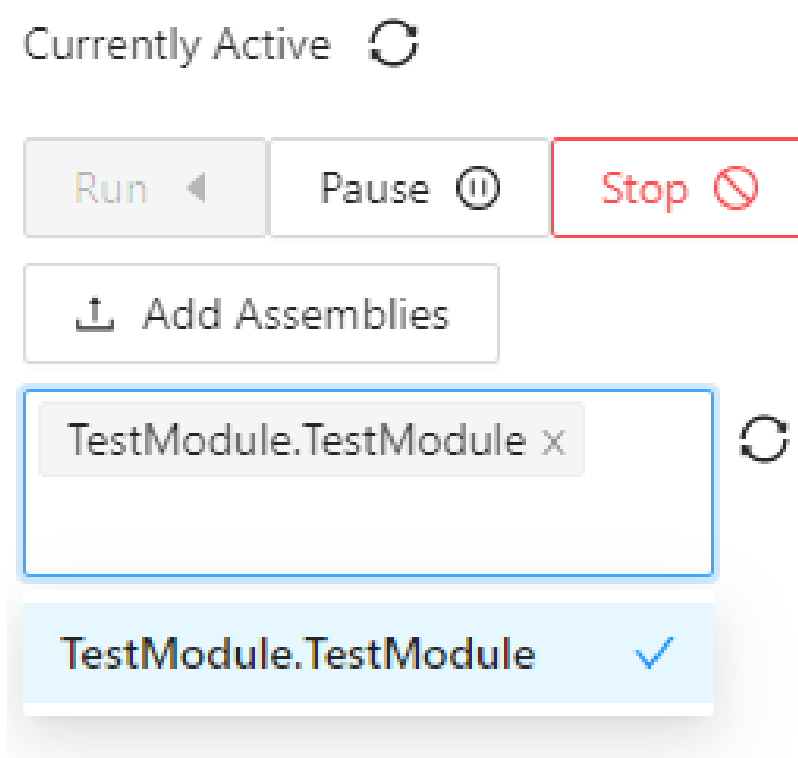


Рис. 4.2.8. Интерфейс управления статусом моделирования

Все объекты в системе моделирования OSMLS наследуются от геометрических примитивов библиотеки NetTopologySuite. Данная библиотека имеет свой репозиторий и официальную документацию. Все взаимодействия между объектами (такие как, например, пересечение линии и полигона) также можно выполнять с использованием данной библиотеки, вместо реализации своих дополнительных методов и функций.

Все объекты в системе моделирования OSMLS отображаются на клиенте с использованием библиотеки OpenLayers. Каждый тип (класс) объектов может иметь свой собственный стиль, задаваемый атрибутом CustomStyle. Если не задать объекту стиль, стиль по умолчанию будет таким, как показано на рис. 4.2.9.

```
new style.Style({
  image: new style.Circle({
    opacity: 1.0,
    scale: 1.0,
    radius: 3,
    fill: new style.Fill({
      color: 'rgba(255, 255, 255, 0.4)'
    }),
    stroke: new style.Stroke({
      color: 'rgba(0, 0, 0, 0.4)',
      width: 1
    }),
  }),
  fill: new style.Fill({
    color: 'rgba(255, 255, 255, 0.4)'
  }),
  stroke: new style.Stroke({
    color: 'rgba(0, 0, 0, 0.4)',
    width: 1
  })
});
```

Рис. 4.2.9. Openlayers-стиль объекта по умолчанию

### 4.3. Задание

Требуется разработать свой модуль в соответствии с индивидуальным заданием, для этого нужно:

- 1) выбрать вариант из табл. 4.3.1;
- 2) создать проект модуля с названием, соответствующим тематике варианта;
- 3) создать алгоритм, реализующий моделируемые действия. Все объекты, создаваемые в рамках задач, должны быть описаны отдельными классами и иметь отличный от стандартного стиль отображения, заданный атрибутом CustomStyle;
- 4) протестировать работу модуля в связке с системой моделирования.

Таблица 4.3.1

#### Индивидуальные задания на разработку модулей по вариантам

| № | Объект          | Базовый тип | Количество | Действия                                                                                             |
|---|-----------------|-------------|------------|------------------------------------------------------------------------------------------------------|
| 1 | Дикая местность | Polygon     | 1          | Постоянно генерирует <b>волков</b> и <b>олений</b> в случайной точке себя                            |
|   | Волк            | Point       | Много      | Пытается догнать и съесть ближайших <b>олений</b> . Не может выйти за пределы <b>дикой местности</b> |
|   | Олень           | Point       | Много      | Двигается хаотично. Не может выйти за пределы <b>дикой местности</b>                                 |

| № | Объект           | Базовый тип | Количество | Действия                                                                                                                                                                 |
|---|------------------|-------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2 | Дикая местность  | Polygon     | 1          | При инициализации генерирует внутри себя несколько <b>природных барьеров</b> . Постоянно генерирует <b>волков</b> и оленей в случайной точке себя                        |
|   | Природный барьер | Polygon     | Много      |                                                                                                                                                                          |
|   | Волк             | Point       | Много      | Пытается догнать и съесть ближайших <b>олений</b> . Не может выйти за пределы <b>дикой местности</b> . Не может пересечь <b>природный барьер</b>                         |
|   | Олень            | Point       | Много      | Двигается хаотично. Не может выйти за пределы <b>дикой местности</b> . Не может пересечь <b>природный барьер</b>                                                         |
| 3 | Дикая местность  | Polygon     | 1          | Генерирует некоторое количество <b>волков</b> и <b>олений</b> при инициализации                                                                                          |
|   | Волк             | Point       | Много      | Пытается догнать и съесть ближайших <b>олений</b> . Создает новых <b>волков</b> с <b>волками</b> противоположного пола. Не может выйти за пределы <b>дикой местности</b> |
|   | Олень            | Point       | Много      | Создает новых <b>олений</b> с <b>оленьями</b> противоположного пола. Двигается хаотично. Не может выйти за пределы <b>дикой местности</b>                                |

| № | Объект          | Базовый тип | Количество | Действия                                                                                                                                                                                                     |
|---|-----------------|-------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4 | Дикая местность | Polygon     | 1          | Постоянно генерирует <b>волков</b> и <b>олений</b> в случайной точке себя                                                                                                                                    |
|   | Волк            | Point       | Много      | Имеет постоянно растущий голод, при достижении определенной отметки умирает. Пытается догнать и съесть ближайших <b>олений</b> , восполнив тем самым голод. Не может выйти за пределы <b>дикой местности</b> |
|   | Олень           | Point       | Много      | Двигается хаотично. Не может выйти за пределы <b>дикой местности</b>                                                                                                                                         |
| 5 | Дикая местность | Polygon     | 1          | Постоянно генерирует <b>волков</b> и <b>олений</b> в случайной точке себя                                                                                                                                    |
|   | Волк            | Point       | Много      | Пытается догнать и съесть ближайших <b>олений</b> . Не может выйти за пределы <b>дикой местности</b> . Может сбиваться с другими <b>волками</b> в стаю, в стае движется быстрее                              |
|   | Олень           | Point       | Много      | Двигается хаотично. Не может выйти за пределы <b>дикой местности</b>                                                                                                                                         |
| 6 | Аэропорт        | Point       | 5          | Создает <b>самолеты</b> , посылая их в другой случайный <b>аэропорт</b>                                                                                                                                      |
|   | Самолет         | Point       | Много      | Двигается до заданного <b>аэропорта</b> с заданной скоростью                                                                                                                                                 |

| № | Объект        | Базовый тип | Количество | Действия                                                                                                                                  |
|---|---------------|-------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| 7 | Аэропорт      | Point       | 5          | Хранит количество своих <b>самолетов</b> . Посылает <b>самолеты</b> в другой случайный <b>аэропорт</b> , если <b>самолеты</b> у него есть |
|   | Самолет       | Point       | Много      | Двигается до заданного <b>аэропорта</b> с заданной скоростью. При достижении цели добавляется в аэропорт                                  |
| 8 | Аэропорт      | Point       | 5          | Создает <b>самолеты</b> , посылая их в другой случайный <b>аэропорт</b>                                                                   |
|   | Самолет       | Point       | Много      | Двигается до заданного <b>аэропорта</b> с заданной скоростью. Должен избегать попадания в <b>грозовую бурю</b>                            |
|   | Грозовая буря | Polygon     | 10         | Создаются в начале работы модуля                                                                                                          |
| 9 | Аэропорт      | Point       | 5          | Создает <b>самолеты</b> , посылая их в другой случайный <b>аэропорт</b>                                                                   |
|   | Самолет       | Point       | Много      | Двигается до заданного <b>аэропорта</b> с заданной скоростью. Должен избегать попадания в <b>грозовую бурю</b>                            |
|   | Грозовая буря | Polygon     | Много      | Появляются и исчезают в случайные моменты времени                                                                                         |

| №  | Объект                  | Базовый тип | Количество | Действия                                                                                                                      |
|----|-------------------------|-------------|------------|-------------------------------------------------------------------------------------------------------------------------------|
| 10 | Аэропорт                | Point       | 5          | Создает <b>самолеты</b> , посылая их в другой случайный <b>аэропорт</b> так, чтобы у них на пути не было <b>грозовых бурь</b> |
|    | Самолет                 | Point       | Много      | Двигается до заданного <b>аэропорта</b> с заданной скоростью                                                                  |
|    | Грозовая буря           | Polygon     | 10         | Создаются в начале работы модуля                                                                                              |
| 11 | Регион подводной лодки  | Polygon     | 1          | Постоянно генерирует <b>объекты для исследования</b> в случайной точке себя                                                   |
|    | Подводная лодка         | Point       | 1          | Двигается в сторону <b>объектов для исследования</b> ; после достижения цели объект для исследования исчезает                 |
|    | Объект для исследования | Point       | Много      |                                                                                                                               |
| 12 | Регион подводной лодки  | Polygon     | 1          | Постоянно генерирует <b>объекты для исследования</b> в случайной точке себя                                                   |

| №  | Объект                  | Базовый тип | Количество | Действия                                                                                                                                                           |
|----|-------------------------|-------------|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | Подводная лодка         | Point       | 1          | Двигается в сторону <b>объектов для исследования</b> ; после достижения цели и прошествия времени, необходимого для исследования, объект для исследования исчезает |
|    | Объект для исследования | Point       | Много      |                                                                                                                                                                    |
| 13 | Регион подводной лодки  | Polygon     | 1          | Постоянно генерирует <b>объекты для исследования</b> в случайной точке себя                                                                                        |
|    | Подводная лодка         | Point       | 1          | Двигается в сторону <b>объектов для исследования</b> ; после достижения цели объект для исследования исчезает                                                      |
|    | Объект для исследования | Point       | Много      | Может потерять актуальность и исчезнуть после прошествия определенного времени                                                                                     |
| 14 | Регион подводной лодки  | Polygon     | 1          | Постоянно генерирует <b>объекты для исследования</b> в случайной точке себя                                                                                        |
|    | Подводная лодка         | Point       | 1          | Двигается в сторону <b>объектов для исследования</b> ; после достижения цели объект для исследования исчезает. Не может пересекать <b>барьерные рифы</b>           |

| №  | Объект                  | Базовый тип | Количество | Действия                                                                                                                           |
|----|-------------------------|-------------|------------|------------------------------------------------------------------------------------------------------------------------------------|
|    | Объект для исследования | Point       | Много      |                                                                                                                                    |
|    | Барьерный риф           | Polygon     | 5          | Создаются в начале работы модуля внутри <b>региона подводной лодки</b>                                                             |
| 15 | Регион подводной лодки  | Polygon     | 1          | Постоянно генерирует объекты для исследования в случайных точках себя                                                              |
|    | Подводная лодка         | Point       | 1          | Двигается в сторону <b>объектов для исследования</b> ; после достижения цели объект для исследования исчезает                      |
|    | Объект для исследования | Polygon     | Много      | Следует обратить внимание, что в данном случае это полигон                                                                         |
| 16 | Порт                    | Point       | 5          | Создает <b>корабли</b> , посылая их в другой случайный <b>порт</b>                                                                 |
|    | Корабль                 | Point       | Много      | Двигается до заданного <b>порта</b> с заданной скоростью                                                                           |
| 17 | Порт                    | Point       | 5          | Хранит количество своих <b>кораблей</b> . Посылает <b>корабли</b> в другой случайный <b>порт</b> , если <b>корабль</b> у него есть |

| №  | Объект        | Базовый тип | Количество | Действия                                                                                                          |
|----|---------------|-------------|------------|-------------------------------------------------------------------------------------------------------------------|
|    | Корабль       | Point       | Много      | Двигается до заданного <b>порта</b> с заданной скоростью. При достижении цели добавляется в <b>порт</b>           |
| 18 | Порт          | Point       | 5          | Создает <b>корабли</b> , посылая их в другой случайный порт                                                       |
|    | Корабль       | Point       | Много      | Двигается до заданного <b>порта</b> с заданной скоростью. Должен избегать попадания в <b>грозовую бурю</b>        |
|    | Грозовая буря | Polygon     | 10         | Создаются в начале работы модуля                                                                                  |
| 19 | Порт          | Point       | 5          | Создает <b>корабли</b> , посылая их в другой случайный <b>порт</b>                                                |
|    | Корабль       | Point       | Много      | Двигается до заданного <b>порта</b> с заданной скоростью. Должен избегать попадания в <b>грозовую бурю</b>        |
|    | Грозовая буря | Polygon     | Много      | Появляются и исчезают в случайные моменты времени                                                                 |
| 20 | Порт          | Point       | 5          | Создает <b>корабли</b> , посылая их в другой случайный порт так, чтобы у них на пути не было <b>грозовых бурь</b> |

| №  | Объект        | Базовый тип | Количество | Действия                                                                                                                  |
|----|---------------|-------------|------------|---------------------------------------------------------------------------------------------------------------------------|
|    | Корабль       | Point       | Много      | Двигается до заданного <b>порта</b> с заданной скоростью                                                                  |
|    | Грозовая буря | Polygon     | 10         | Создаются в начале работы модуля                                                                                          |
| 21 | Вокзал        | Point       | 5          | Создает <b>поезда</b> , посылая их в другой случайный <b>вокзал</b>                                                       |
|    | Поезд         | Point       | Много      | Двигается до заданного <b>порта</b> с заданной скоростью                                                                  |
| 22 | Вокзал        | Point       | 5          | Хранит количество своих <b>поездов</b> . Посылает <b>поезда</b> в другой случайный <b>вокзал</b> , если поезд у него есть |
|    | Поезд         | Point       | Много      | Двигается до заданного <b>вокзала</b> с заданной скоростью. При достижении цели добавляется в <b>вокзал</b>               |
| 23 | Вокзал        | Point       | 5          | Создает <b>поезда</b> , посылая их в другой случайный <b>вокзал</b>                                                       |
|    | Поезд         | Point       | Много      | Двигается до заданного <b>вокзала</b> с заданной скоростью. Должен дожидаться исчезновения <b>преграды</b>                |
|    | Преграда      | Polygon     | 10         | Создаются в начале работы модуля, исчезают через какое-то время                                                           |

| №  | Объект   | Базовый тип | Количество | Действия                                                                                                                                                                                        |
|----|----------|-------------|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 24 | Вокзал   | Point       | 5          | Создает <b>поезда</b> , посылая их в другой случайный <b>вокзал</b>                                                                                                                             |
|    | Поезд    | Point       | Много      | Двигается до заданного <b>вокзала</b> с заданной скоростью. Должен дожидаться исчезновения <b>преграды</b>                                                                                      |
|    | Преграда | Polygon     | Много      | Появляются и исчезают в случайные моменты времени                                                                                                                                               |
| 25 | Вокзал   | Point       | 5          | Создает <b>поезда</b> , посылая их в другой случайный <b>вокзал</b> так, чтобы у них на пути не было <b>преград</b>                                                                             |
|    | Поезд    | Point       | Много      | Двигается до заданного <b>вокзала</b> с заданной скоростью                                                                                                                                      |
|    | Преграда | Polygon     | 10         | Создаются в начале работы модуля                                                                                                                                                                |
| 26 | Комбайн  | Polygon     | 1          | Собирает урожай с <b>поля</b>                                                                                                                                                                   |
|    | Поле     | Polygon     | 1          | Полигон, на котором изначально есть урожай. Урожай может быть собран (здесь и в других подобных задачах можно использовать ILinearRing [] Holes, свойство полигона, отвечающее за «дыры» в нем) |
| 27 | Комбайн  | Polygon     | 2          | Собирает урожай с ближайшего <b>поля</b>                                                                                                                                                        |
|    | Поле     | Polygon     | 5          | Полигон, на котором изначально есть урожай. Урожай может быть собран                                                                                                                            |

| №  | Объект   | Базовый тип | Количество | Действия                                                                                                                                                               |
|----|----------|-------------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 28 | Комбайн  | Polygon     | 1          | Собирает урожай с <b>поля</b>                                                                                                                                          |
|    | Поле     | Polygon     | 1          | Полигон, на котором изначально есть урожай. Урожай может быть собран. Урожай снова вырастает через какое-то время                                                      |
| 29 | Комбайн  | Polygon     | 1          | Собирает урожай с <b>поля</b> . Есть максимальная вместимость. При достижении максимальной вместимости должен отвезти урожай на <b>склад</b> и снова продолжить работу |
|    | Поле     | Polygon     | 1          | Полигон, на котором изначально есть урожай. Урожай может быть собран                                                                                                   |
|    | Склад    | Point       | 1          |                                                                                                                                                                        |
| 30 | Комбайн  | Polygon     | 1          | Собирает урожай с <b>поля</b> , не может пересекать <b>преграды</b>                                                                                                    |
|    | Поле     | Polygon     | 1          | Полигон, на котором изначально есть урожай                                                                                                                             |
|    | Преграда | Polygon     | 5          |                                                                                                                                                                        |

## СПИСОК РЕКОМЕНДУЕМОЙ ПО КУРСУ ЛИТЕРАТУРЫ

1. Инженерная геодезия и геоинформатика. Краткий курс / М. Я. Брынь [и др.]. – СПб. : Лань, 2015. – 288 с.
2. *Шипулин, В. Д.* Основные принципы геоинформационных систем : учеб. пособие / В. Д. Шипулин. – Х.: ХНАГХ, 2010. – 337 с.
3. *Турлапов, В. Е.* Геоинформационные системы в экономике : Учебно-методическое пособие / В. Е. Турлапов. – Нижний Новгород : НФ ГУВШЭ, 2007. – 118 с.
4. *Чандра, А. М.* Дистанционное зондирование и географические информационные системы / А. М. Чандра, С. К. Гош. – М. : Техносфера, 2008. – 312 с.
5. Пространственные данные [Электронный ресурс] // Блог Т. Самойлова. – Режим доступа : <https://tsamsonov.github.io/r-geo-course/spatial.html>
6. *Кащенко, Н. А.* Геоинформационные системы : Учебное пособие для вузов / Н. А. Кащенко, Е. В. Попов, А. В. Чечин ; Нижегородский государственный архитектурно-строительный университет. – Н. Новгород : ННГАСУ, 2012. – 130 с.
7. *Митчелл, Э.* Руководство по ГИС-анализу. Часть 1. Пространственные модели и взаимосвязи : пер. с англ. / Э. Митчелл. – Киев : ЗАО ЕСОММ Со; Стилос, 2000. – 198 с.

8. 9 лучших картографических сервисов [Электронный ресурс] // Информационный сайт fornote.net. – Режим доступа : <https://fornote.net/2021/03/9-maps/> .

9. Apple delivers a new redesigned Maps for all users in the United States [Электронный ресурс] // Сайт Apple. – Режим доступа: <https://www.apple.com/newsroom/2020/01/apple-delivers-a-new-redesigned-maps-for-all-users-in-the-united-states/>

10. OpenStreetMap [Электронный ресурс] // Некоммерческий веб-картографический проект по созданию подробной и бесплатной географической карты мира. – Режим доступа : [www.openstreetmap.org](http://www.openstreetmap.org)

11. Пространственные данные: потребности экономики в условиях цифровизации / Е. Б. Белогурова [и др.] ; Федеральная служба государственной регистрации, кадастра и картографии ; Национальный исследовательский университет «Высшая школа экономики» ; НИИ «АЭРОКОСМОС». – М : НИУ ВШЭ, 2020. – 128 с.

12. Применение ГИС-технологий для повышения эффективности имитационного моделирования каскадов ГЭС / А. Ю. Арестова, С. В. Митрофанов, А. Г. Русина, А. А. Колесников // Журнал Сибирского федерального университета. Серия : Техника и технологии. – 2020. – Т. 13. – № 6. – С. 732–744. – DOI 10.17516/1999-494X-0262. – EDN BPNYRI.

13. Tangramob: an agent-based simulation framework for validating urban smart mobility solutions / Carlo Castagnari, Flavio Corradini, Francesco De Angelis [et al.] // Cornell University, 2018. – URL : <https://arxiv.org/pdf/1805.10906.pdf>

14. A multi-scale agent-based modelling framework for urban freight distribution // André Alho, Bhavathrathan B. K., Monique Stinson [et al.] // Transportation Research Procedia: Elsevier, 2018г. – V. 27. – P. 188–196. – URL : <https://www.sciencedirect.com/science/article/pii/S2352146517310359>

15. TOXI-CITY: an agent-based model for exploring the effects of risk awareness and spatial configuration on the survival rate in the case of industrial accidents [Электронный ресурс] // Open Edition journals. – Режим доступа : <https://journals.openedition.org/cybergeogeo/26522?lang=en&gathStatIcon=tru>

16. Новостной портал РБК [Электронный ресурс]. – Режим доступа : <https://www.rbc.ru/politics/04/03/2019/5c7cd5fe9a794760d9cfb900>

17. *Zhang, C., Yan, J., Li, C., Rui, X., Liu, L., Bie, R.* On estimating air pollution from photos using convolutional neural network. In: Proceedings of the 2016 ACM on Multimedia Conference, MM 2016, pp. 297–301. ACM, New York (2016).

18. *Cheang, E. K., Cheang, T. K., Tay, Y. H.* Using convolutional neural networks to count palm trees in satellite images. arXiv preprint arXiv:1701.06462 (2017).

19. *Taleai, M. Mesgari, M. S., Sharifi, M. A., Sliuzas, R. V.* [et al.]. A spatial decision support system for evaluation various land uses in built up urban area. – 2005.

20. *Birant, D.* Comparison of Decision Tree Algorithms for Predicting Potential Air Pollutant Emissions with Data Mining Models, Journal of Environmental Informatics. Vol. 17 (2011) No.1, p. 46.

21. *Golubev, A., Sadovnikova, N., Parygin, D., Glinyanova, I., Finogeev, A., Shcherbakov, M.* Woody plants area estimation using ordinary satellite images and deep learning (2018) Communications in Computer and Information Science, 858, pp. 302–313. DOI: 10.1007/978-3-030-02843-5\_24 Тип документа: Conference Paper.

22. Подходы к визуализации экологической информации на карте города / Д. С. Парыгин, Н. П. Садовникова, А. В. Голубев, А. О. Недоступов, А. Г. Финогеев // Известия ВолгГТУ : научный журнал № 5 (215) / Волг-

ГТУ. – Волгоград, 2018. – Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах»). – С. 60–64.

23. Distributed Administration of Multi-Agent Model Properties / A. Burova, S. Burov, D. Parygin, A. Gurtyakov, N. Rashevskiy // CEUR Workshop Proceedings : 24rd International Conference “Internet and Modern Society”, IMS 2021, St. Petersburg, Russia, 23–26 June 2021. – Vol. 3090. – P. 24–33. – Mode of access : <http://ceur-ws.org/Vol-3090/paper02.pdf>

24. Rule-Based Pedestrian Simulation / S. Burov, D. Parygin, A. Finogeev, D. Ather, N. Rashevskiy // Proceedings of the 2nd International Conference on “Advancement in Electronics & Communication Engineering” (AECE 2022), Ghaziabad, India, 14–15 July 2022. – SSRN, 2022. – Mode of access : [https://papers.ssrn.com/sol3/Delivery.cfm/SSRN\\_ID4160252\\_code5284765.pdf?abstractid=4160252&mirid=1](https://papers.ssrn.com/sol3/Delivery.cfm/SSRN_ID4160252_code5284765.pdf?abstractid=4160252&mirid=1) – DOI : 10.2139/ssrn.4160252

25. Development of Scenarios for Modeling the Behavior of People in an Urban Environment / A. Anokhin, S. Burov, D. Parygin, V. Rent, N. Sadovnikova, A. Finogeev // Studies in Systems, Decision and Control : Society 5.0: Cyberspace for Advanced Human-Centered Society. – Springer, 2021. – Vol. 333. – P. 103–114. – DOI : 10.1007/978-3-030-63563-3\_9

26. Multi-agent Approach to Modeling the Dynamics of Urban Processes (on the Example of Urban Movements) / D. Parygin, A. Usov, S. Burov, N. Sadovnikova, P. Ostroukhov, A. Pyannikova // Communications in Computer and Information Science : Proceedings of the 6th International Conference on Electronic Governance and Open Society: Challenges in Eurasia (EGOSE 2019), St. Petersburg, Russia, 13–14 November 2019. – Springer, 2020. – Vol. 1135. – P. 243–257. – DOI : 10.1007/978-3-030-39296-3\_18

Учебное издание

Данила Сергеевич **Парыгин**  
Александр Владимирович **Игнатьев**  
Наталья Петровна **Садовникова**  
Александр Сергеевич **Гуртяков**

## **ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ**

*Учебно-методическое пособие*

Редактор *Л. Н. Рыжих*  
Компьютерная верстка *Е. В. Макаровой*

Темплан 2023 г. (учебники и учебные пособия). Поз. № 66.  
Подписано в печать 10.03.2023. Формат 60×84 1/16. Бумага офсетная.  
Гарнитура Times. Печать офсетная. Усл. печ. л. 7,44. Уч.-изд. л. 5,57.  
Тираж 30 экз. Заказ 0000

Волгоградский государственный технический университет.  
400005, г. Волгоград, просп. В. И. Ленина, 28, корп. 1.

Отпечатано в типографии Издательства ВолгГТУ.  
400005, г. Волгоград, просп. В. И. Ленина, 28, корп. 7.

[illegible]