Оглавление

[Геоинформационные технологии 2](#_Toc48152428)

[Алгоритм отображения движения точечного объекта на карте геоинформационной системы 2](#_Toc48152429)

[Совместное использование вычислительной системы Matlab и языка программирования Java при разработке научных и прикладных программных продуктов для решения геодезических задач 16](#_Toc48152430)

[Использование сетевых геоинформационных технологий в рамках проекта европейского сотрудничества Upgrade Black Sea Scene 26](#_Toc48152431)

[Исследование алгоритмов автоматизированной генерализации линейных объектов 51](#_Toc48152432)

# Геоинформационные технологии

# Алгоритм отображения движения точечного объекта на карте геоинформационной системы

Аннотация. Разработан обобщенный алгоритм отображения движения точечного объекта на карте геоинформационной системы. Исходными данными для алгоритма являются координаты траектории движения объекта, записываемые GPS‑приемником в виде NMEA-файла и цифровая карта ГИС на соответствующий участок местности.

Современные мобильные геоинформационные системы обладают возможностью отображать свое местоположение на карте. В основе этого важного свойства мобильных ГИС лежит использование двух технологий: геоинформационной и глобального позиционирования. Если мобильное устройство имеет встроенный GPS‑приемник, то с его помощью можно определить местоположение мобильного устройства и, следовательно, местоположение самой ГИС. К сожалению, в русскоязычной научной литературе немного публикаций о мобильных ГИС [1, 2] и практически полностью отсутствует информация об алгоритмах, лежащих в основе функциональных возможностей современных ГИС.

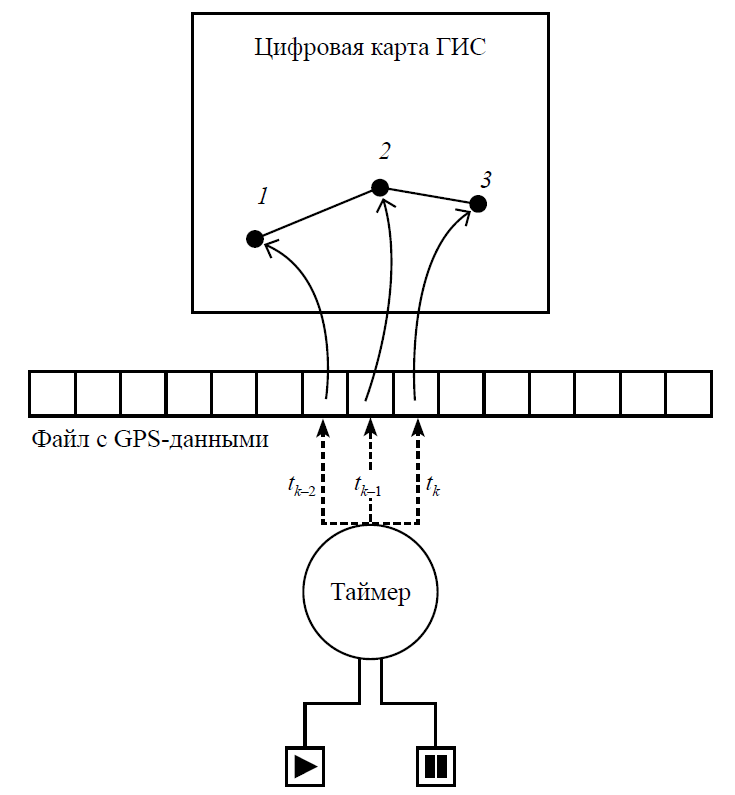
Цель данной работы заключалась в разработке алгоритма, позволяющего отображать движение объекта на карте ГИС, включая и перемещение мобильного устройства вместе с картой. Чтобы изобразить точку на карте, необходимо знать, по крайней мере, две координаты X и Y, при отображении движения объекта на карте потребуется последовательность координат (Xk и Yk), соответствующая разным моментам времени. Данную последовательность координат можно извлечь из стандартного NMEA-файла GPS‑приемника, выполнившего реальное движение на местности. Таким образом, в основе рассматриваемого здесь алгоритма лежит использование данных NMEA-файла с координатами траектории движения объекта на местности. Необходимые координаты можно получить также в результате моделирования процесса движения на основе компьютерных экспериментов.

Исходными данными для разработки алгоритма были: набор координат (Xk и Yk), соответствующий некоторой траектории движения объекта в виде NMEA-файла и цифровая карта ГИС на выбранный участок местности. Задача заключалась в разработке алгоритма, отображающего движение точечного объекта на карте. В ходе работ были проанализированы три алгоритма, лежащих в основе решения поставленной задачи. Первый алгоритм для анализа данных NMEA-файла с целью извлечения из него последовательности координат (Xk и Yk), второй алгоритм для вычисления путевого угла объекта, при этом отсчет угла выполнялся от направления на восток и третий алгоритм для определения курса объекта относительно стран света. Обобщенный алгоритм отображения движения точечного объекта на карте ГИС включает три указанных ранее: алгоритм анализа данных NMEA, алгоритм вычисления путевого угла объекта и алгоритм определения курса объекта относительно стран света.

Основным управляющим элементом обобщенного алгоритма является таймер (Timer). Стандартный элемент управления таймер позволяет программе генерировать через определенные промежутки времени события. С помощью объекта типа таймер можно устанавливать требуемый период времени для выполнения набора процедур или функций. Например, для многократно выполняемого отсчета интервала времени в одну секунду, надо установить следующие свойства таймера:

|  |  |
| --- | --- |
| 1.  2. | Timer.Interval = 1000;  Timer.Enabled = True. |

В результате каждую секунду будет запускаться процедура считывания координат X, Y объекта из NMEA-файла, и далее точка с полученными координатами будет отображаться на карте ГИС. На рисунке представлена схема обобщенного алгоритма отображения движения точечного объекта на карте.



Основные компоненты и взаимосвязи между ними

На цифровой карте отмечены три точки 1, 2, 3 траектории движения объекта в различные моменты времени , , , соединенные сплошной линией. Включение и выключение таймера, управляющего отображением траектории движения объекта на карте, осуществляется с помощью кнопок, показанных на схеме и нажимаемых пользователем.

Опишем кратко состав NMEA-файла [3]. Это файл, записанный в формате ASCII, по сути, представляет собой протокол для обмена информацией между GPS‑приемником и другими устройствами, например ПК или КПК. Каждое сообщение NMEA начинается со знака $, за которым следует двухсимвольное обозначение устройства (для GPS‑приемника — GP) и заканчивается последовательностью \r\n — символом перевода каретки и перехода на новую строку. Все данные сообщения содержатся в одной строке, причем поля данных разделяются запятыми. Точность данных в сообщении зависит от вида сообщения. Для нашей цели вполне достаточно широко используемое сообщение типа $GPRMC, которое содержит минимальный, но основной набор данных: местоположение объекта, его скорость и время. В качестве примера приведем сообщение GPRMC из NMEA-файла:

|  |  |
| --- | --- |
| 1.  2. | $GPRMC,105452.000,A,5522.4271,N,03644.1610,E,0.0,  303.4,070115,,,A\*6C\r\n. |

В этом сообщении закодирована следующая информация: дата определения координат объекта — 7 января 2015 г., всемирное время UTC определения координат — 10h54m52s, координаты объекта — 55º22.4271' с.ш. и 36º44.1610' в.д. Подчеркнем, что координаты объекта представлены в градусах и минутах, причем минуты даются с точностью до четырех знаков после запятой (или точки как разделителя целой и дробной частей вещественного числа в формате USA). В дальнейшем нам понадобится то, что в NMEA-файле широта местоположения объекта находится в позиции после третьей запятой, а долгота — после пятой запятой. В конце сообщения передается контрольная сумма после символа '\*' в виде двух шестнадцатеричных цифр — 6C.

Рассмотрим алгоритм анализа NMEA-файла с целью извлечения набора координат ( и ), соответствующих траектории движения объекта.

Алгоритм анализа данных NMEA

Шаг 1. Прочитать строку GPRMC из NMEA-файла.

Определение координаты Y объекта.

Шаг 2. Найти позицию третьей запятой в строке (q).

Шаг 3. Найти позицию четвертой запятой в строке (r).

Шаг 4. Найти, начиная с позиции q, символ десятичной точки (t).

Шаг 5. Извлечь один символ из строки, находящийся в позиции (r+1).

Шаг 6. Если этот символ равен 'N', то переменная NorthernHemisphere получает значение 1, иначе −1.

Шаг 7. Извлечь (r – t + 2) символов строки, начиная с позиции (t – 2).

Шаг 8. Извлечь (t – q – 3) символов строки, начиная с позиции (q + 1).

Шаг 9. Преобразовать строки в вещественные числа и вычислить координату Y объекта в радианной мере.

Определение координаты X объекта.

Шаг 10. Найти позицию пятой запятой в строке (n).

Шаг 11. Найти позицию шестой запятой в строке (m).

Шаг 12. Найти, начиная с позиции n, символ десятичной точки (p).

Шаг 13. Извлечь один символ из строки, находящийся в позиции (m + 1).

Шаг 14. Если этот символ равен 'E', то переменная EasternHemisphere получает значение 1, иначе –1.

Шаг 15. Извлечь (m – p + 2) символов строки, начиная с позиции (p – 2).

Шаг 16. Извлечь (p – n + 2) символов строки, начиная с позиции (n + 1).

Шаг 17. Преобразовать строки в вещественные числа и вычислить координату X объекта в радианной мере.

Шаг 18. Если NMEA‑файл не прочитан до конца, то перейти к шагу 1, иначе перейти к шагу 19.

Шаг 19. Закончить алгоритм.

На шаге 6 и 16 данного алгоритма используются переменные NorthernHemisphere и EasternHemisphere для численного кодирования местоположения объекта на Земле. В северном (южном) полушарии переменная NorthernHemisphere принимает значение 1 (–1) соответственно, аналогично в восточном (западном) полушарии EasternHemisphere принимает значение 1 (‑1).

Представленный выше алгоритм можно реализовать в виде программы с использованием всего трех функций из набора работы со строками. Например, на языке Visual Basic потребуются следующие функции: InStr, Mid и Val. Функция InStr выполняет поиск одной строки в другой. Синтаксис функции в простейшем виде выглядит следующим образом.

|  |  |
| --- | --- |
| 1.  2. | Номер\_Первого\_Символа =  InStr(Исходная\_Строка, Искомая\_Строка) |

С помощью данной функции можно начать поиск, указав при этом начальную позицию в исходной строке, например:

|  |  |
| --- | --- |
| 1.  2. | Номер\_Первого\_Символа =  InStr(  Начальная\_Позиция,  Исходная\_Строка,  Искомая\_Строка) |

Второй функцией может быть функция Mid, которая возвращает указанное количество символов из середины строки. Синтаксис функции следующий.

|  |  |
| --- | --- |
| 1.  2. | Результат =  Mid(  Исходная\_Строка,  Начальная\_Позиция[,Число\_Символов]) |

В параметре Начальная\_Позиция указывается номер символа в исходной строке, с которого будет начинаться новая строка. С помощью необязательного параметра Число\_Символов указывается число символов, которые нужно извлечь из исходной строки. В качестве примера запишем вызов функции Mid для извлечения из NMEA-файла координаты Х объекта, выраженной в градусной мере:

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7. | stringDegrees =  Mid(  Message,  FifthCom maPosition + 1,  FollowimgDecimalPoint  – FifthCommaPosition  – 3) |

где Message — сообщение NMEA;  
FifthCommaPosition — положение пятой запятой в сообщении;  
FollowimgDecimalPoint — положение десятичной точки, следующей за пятой запятой в сообщении.

Выражение FollowimgDecimalPoint ‒ FifthCommaPosition ‒ 3 представляет количество символов извлекаемых из сообщения, а выражение FifthCommaPosition + 1 указывает на позицию символа в сообщении, начиная с которого следует извлекать координату X объекта. Заметим, что переменная stringDegrees содержит строку символов, определяющих координату X объекта. Для преобразования текстовой строки в вещественное число и получения координаты X понадобится стандартная функция Val. Функция имеет следующий синтаксис:

|  |  |
| --- | --- |
| 1. | Число = Val(Исходная\_Строка) |

Здесь Число представляет числовую переменную, в которой хранится возвращаемое функцией значение. Вместо параметра Исходная\_Строка нужно подставить строковый литерал. Функция анализирует первый символ строки, и если это число, то последовательно считываются символы исходной строки до тех пор, пока не встретится первый нечисловой символ. Выделив таким способом число из строки, функция Val конвертирует его в числовое значение и возвращает программе. Например, чтобы вычислить координату в градусах, включая тысячные доли минуты, можно использовать следующее выражение:

|  |  |
| --- | --- |
| 1.  2.  3.  4. | Xcoordinate =  (Val(stringDegrees)  + Val(stringMinutes) / 60)  \* EasternHemisphere |

Далее рассмотрим алгоритм вычисления путевого угла объекта, учтем при этом, что отсчет углов будем вести от направления на восток.

Алгоритм вычисления путевого угла объекта.

Шаг 1. Получить координаты объекта в момент времени tk‒1(x1, y1) и tk(x2, y2) с помощью алгоритма анализа NMEA-файла.

Шаг 2. Если x2 = x1 AND y2 = y1, то возвратить в вызывающую функцию угол θ' = 0.

Шаг 3. Вычислить квадрат разности координат X объекта: (x2 ‒ x1)2.

Шаг 4. Вычислить квадрат разности координат Y объекта: (y2 ‒ y1)2.

Шаг 5. Вычислить синус угла курса объекта:

Примечание 1. Во многих математических библиотеках программ имеется функция и обычно отсутствует функция , поэтому вместо вычисления функции будем рассчитывать по формуле

Шаг 6. Вычислить арктангенс угла курса объекта. Если модуль величины , то , иначе выбрать направления отсчета угла на запад, т. е. .

Шаг 7. Если , то .

Примечание 2. Поскольку , либо получаем две формулы: для второй четверти круга и для третьей четверти.

Шаг 8. Если AND , то .

Примечание 3. Учтем, что шаг 8 выполняется для четвертой четверти круга, где , и тогда формула вычисления угла имеет вид .

Шаг 9. Если , то . Сброс значения угла на нуль.

Шаг 10. Возвратить в вызывающую функцию.

Программная реализация рассмотренного выше алгоритма потребует трех математических функций: Sqr — извлечение квадратного корня, Abs — вычисления модуля числа (абсолютного значения) и Atan — вычисления арктангенса угла. Теперь рассмотрим третий алгоритм.

Алгоритм определения курса объекта относительно стран света.

Шаг 1. Получить путевой угол объекта в момент времени с помощью алгоритма вычисления путевого угла.

Шаг 2. Выбрать по кругу направлений сектор, в который попадает путевой угол объекта.

Данный алгоритм можно реализовать с помощью условных конструкций вида «Если условие\_1 истинно, то выполняется оператор\_1, если условие\_2 истинно, то выполняется оператор\_2 и так далее». Например, если путевой угол больше 337,5º, но меньше или равен 22,5º, то курс объекта соответствует направлению «восток», если угол больше 22,5º, но меньше или равен 67,5º — курс объекта «северо-восток» и так далее.

|  |  |
| --- | --- |
| 1.  2.  3.  4. | If Angle > 22.5 AND Angle <= 112.5  Then Direction = «СВ»  End If |

В таблице представлены возможные наименования курса и соответствующие им диапазоны путевых углов.

Соответствие наименования курса и путевого угла

|  |  |  |  |
| --- | --- | --- | --- |
| В | СВ | С | СЗ |
| 337.5º–22.5º | 22.5º–67.5º | 67.5º–112.5º | 112.5º–157.5º |
| З | ЮЗ | Ю | ЮВ |
| 157.5º–202.5º | 202.5º–247.5º | 247.5º–292.5º | 292.5º–337.5º |

Заключение. Разработан обобщенный алгоритм отображения движения точечного объекта на карте геоинформационной системы, состоящий из: алгоритма анализа данных NMEA-файла для извлечения координат траектории движения точечного объекта; алгоритма вычисления путевого угла объекта (отсчет угла выполнялся от направления на восток) и алгоритма определения курса объекта относительно стран света. Данный алгоритм можно использовать в программах моделирующих движение (реальное или гипотетическое) некоторого точечного объекта на карте ГИС. Дальнейшая модификация алгоритма может включать более детальный анализ NMEA-файла с использованием сообщений других типов, содержащих, например, информацию о путевой скорости объекта, расстояния до конечной точки.

# Совместное использование вычислительной системы Matlab и языка программирования Java при разработке научных и прикладных программных продуктов для решения геодезических задач

Аннотация. Рассматриваются вопросы совместного использования вычислительной системы Matlab с языком программирования Java для создания программного обеспечения для решения разных типов научных и прикладных задач в геодезии. Объединение этих вычислительных систем позволяет быстро создавать многофункциональные программы с хорошими показателями быстродействия и оптимального использования вычислительных ресурсов.

Введение. В современном мире существует большой набор разнообразных программных пакетов, в том числе и для решения множества задач в области геодезии. Несмотря на это в геодезической практике очень часто возникают нетривиальные задачи, решение которых не предусмотрено существующими программными продуктами. С расширением области применения программных продуктов растут и требования к ним. Часто при решении научно-производственных задач возникают специфические проблемы, для полноценного решения которых нужна разработка узкоспециализированного программного продукта или надстройка к уже существующему.

Вычислительная система Matlab. Matlab — это вычислительная система, которая включает в себя большой набор вычислительных процедур с очень высокими качественными показателями в виде небольшого количества необходимой оперативной памяти и небольшого процессорного времени. Эти показатели остаются высокими даже при обработке больших количеств данных и применении сложных алгоритмов обработки. Библиотека алгоритмов системы Matlab включает вычислительные процедуры многих направлений науки, а также и решения инженерно-прикладных задач [1].

Работа системы Matlab осуществляется посредством процедурного, интерпретируемого программного языка высокого уровня. Этим языком можно разрабатывать программные скрипты и функции. Программные приложения Matlab являются совокупностью .m файлов, содержащих в себе программную логику в виде скриптов и функций.

Преимущества. Система Matlab позволяет:

∙ быстро и эффективно разрабатывать алгоритмы и программы;

∙ тестировать и оптимизировать разрабатываемые алгоритмы;

∙ осуществлять работы со сложными типами данных без дополнительной разработки методов управления этих типов на низком уровне программирования, а также дает большой набор различных прикладных вычислительных процедур для решения разнообразных научных и инженерных задач.

Недостатки. Наряду с преимуществами, описанными выше, разработка прикладных программ в среде Matlab имеет и некоторые недостатки:

∙ небольшие возможности и трудоемкие средства для создания графического интерфейса пользователя;

∙ невозможность разработки интернет программного обеспечения;  
отсуствие интерфейса к реляционным базам данных;

∙ невозможность напрямую работать с классами данных поскольку язык процедурный, из‑за чего поддержка больших объектных моделей становится трудозатратной, так как невозможно скрывать реализацию класса при его использовании.

Язык программирования Java разработан в 90‑х годах ХХ в. Его синтаксис основан в большой степени на синтаксисе языка С++. Java отличает наличие виртуальной машины (Java Virtual Machine — JVM). Программный код языка Java компилируется не до исполнимого для конкретного типа процессора машинного кода (как при С / С++), а до исполнимого байт-кода, который выполняется виртуальной машиной языка — JVM [2].

Преимущества. Учитывая особенности языка программирования Java, можно отметить основные преимущества:

∙ разработка программного обеспечения обектно-ориентированным программированием, что позволяет создавать как небольшие программы, так и большие и сложные программные системы;

∙ автоматизированное управление памяти виртуальной машиной JVM;

∙ большой набор встроенных контейнеров данных (массивы, списки, стеки и т. д.);

∙ возможность разработки интернет приложений;

∙ наличие интерфейсов к различным видам реляционных баз данных;

∙ возможность разработки графического интерфейса пользователя любой сложности;

∙ платформенная независимость программного обеспечения.

Недостатки. Наряду с указанными выше преимуществами отметим и некоторые недостатки Java:

∙ для работы виртуальной машины JVM нужны дополнительные ресурсы в виде памяти и процессорного времени;

∙ кроме преимуществ наличие виртуальной машины, с одной стороны, приводит к замедлению работы программы, которое становится заметным при обработке больших количеств данных, а с другой стороны, не дает возможности оптимизации алгоритма на уровне реализации программного кода, поскольку нет прямого доступа к оперативной памяти;

∙ отсутствие встроенных стандартных алгоритмов, которые могли бы помочь в разработке специализированного программного обеспечения для решения геодезических задач и следовательно необходимость разработки таких базовых алгоритмов.

Совместное использование вычислительной системы Matlab с языком программирования Java. Преимущества. Производителем системы Matlab предусмотрена возможность интеграции Matlab-приложений с языком программирования Java. Таким образом можно сочетать преимущества обеих систем и вместе с тем исключить некоторые недостатки.

Способ интеграции. Чтобы использовать функцию Matlab в среде Java, нужно разработать ее в \*.m файле и протестировать в консоле-интерпретаторе Matlab. Системная команда deploytool позволяет паковать набор функций Matlab в библиотеку для использования в различных других средах разработки программного обеспечения (в том числе и Java). Результат — созданный Java-пакет (\*.jar), представляющий собой класс, содержащий в себе указанные функции в виде методов. Исполнение этих методов требует наличия виртуальной машины Matlab (MCR) на компьютере пользователя.

Передача параметров к Matlab-функции, вызываемой в Java, и получение результатов ее исполнения, делается интерфейсами MWNumericArray для численных массивов и MWStructArray для структур данных. Для Java они представлены в файле javabuider.jar — пакет, который включен в инсталяции Matlab.

Важно отметить, что из‑за того, что Matlab является нестрого типизированным языком следует все численные параметры (в том числе и целочисленные) из Java передавать в виде double. Другая особенность, которую следует учесть, — это то, что индексация массивов в Java начинается с нуля, а в Matlab — с единицы. При интеграции двух систем это указывается в виде первого параметра каждой вызываемой функции. Все входные параметры вызываемой в Java функции Matlab должны быть экземплярами некоторого из классов MWNumericArray или MWStructArray.

Вызванная в Java функция Matlab возвращает результат типа Object[], а дальше посредством динамического преобразования типа (dynamic casting) и использования метода get() результат присваивается соответственной переменной в Java. Ниже приведен простой пример, показывающий использование в Java метода mlFunc(), который является функцией Matlab.

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11. | double[] a = new double[3];  a = {1., 2., 3.};  double[][] b = new double[3][3];  b = {{1., 2., 3.}, {4., 5., 6.}, {7., 8., 9.}};  MWNumericArray mA = new MWNumericArray(a);  MWNumericArray mB = new MWNumericArray(b);  MatlabFunctions MLFuncs = new MLFuncs();  Object[] oResult = MLFuncs.mlFunc(1, mA, mB);  MWNumericArray mResult =  (MWNumericArray) oResult[0];  double[][] result = (double[][]) mResult.get(1); |

Совместное использование Matlab и Java при создании программного обеспечения деформационного анализа. Уравнивание практически всех видов измерений в геодезических сетях на сегодняшний день выполняется параметрическим методом. Реализация этого метода сводится в основном к составлению матрицы коэффициентов уравнений поправок, весовой матрицы и матрицы свободных членов уравнений поправок. Дальше производятся матричные вычисления с этими матрицами и таким образом определяются неизвестные параметры и их оценка точности.

Система Matlab обладает богатым набором функций для построения матриц и матричных вычислений. Поскольку возможно получить и псевдообратную матрицу вырожденной матрицы при критерии Moore-Penrose, то можно легко реализовать не только уравнивание сетей, но и выполнить предварительную оценку точности инженерно-геодезических сетей.

На основе выше описанных способов совместного использования Matlab и Java создано программное приложение для вычисления вертикальных деформаций пунктов нивелирной инженерно-геодезической сети. Программа составлена из четырех основных компонентов (рис. 1−3):

∙ MySQL база данных для сохранения данных измерений и результатов обработки каждого цикла;

∙ Matlab вычислительный модуль для матричных вычислений метода наименьших квадратов;

∙ Java интеграционный модуль;

∙ Java модуль графического интерфейса пользователя.

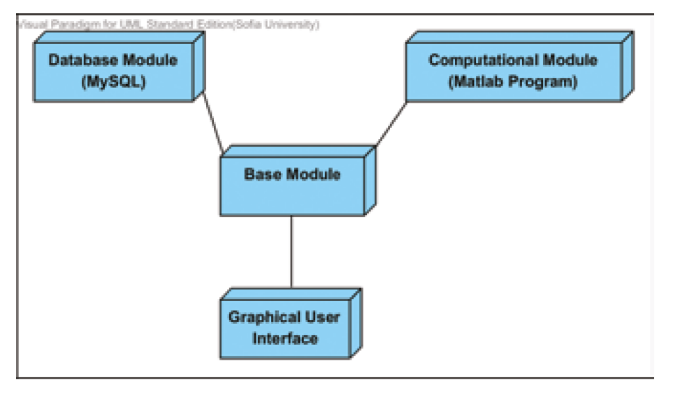


Рис. 1. Схема разложения архитектуры компонентов системы

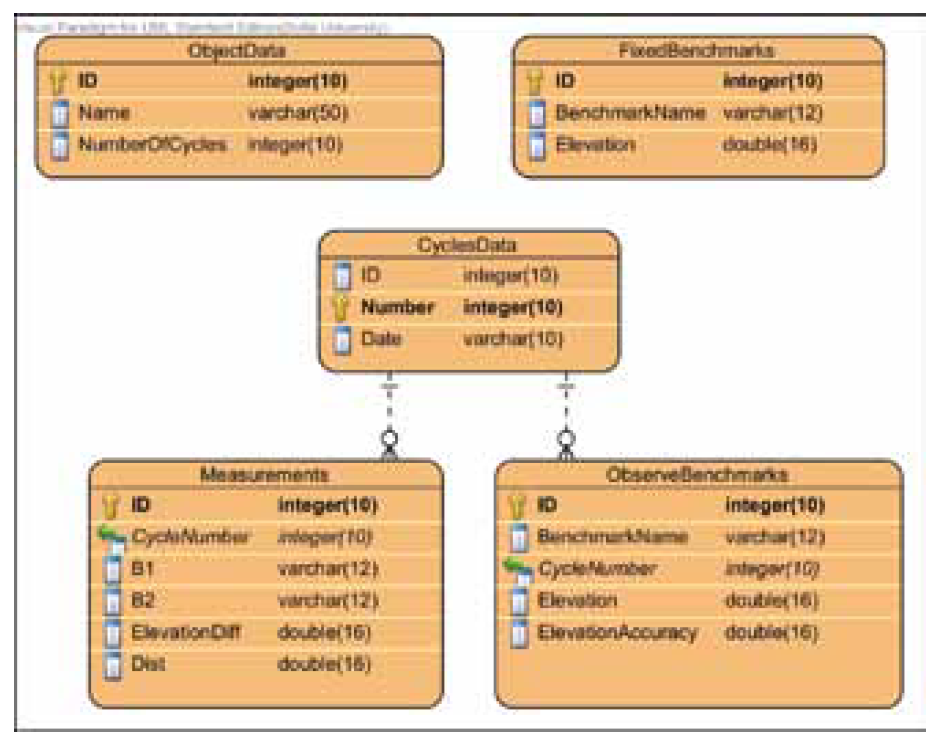


Рис. 2. Схема базы данных

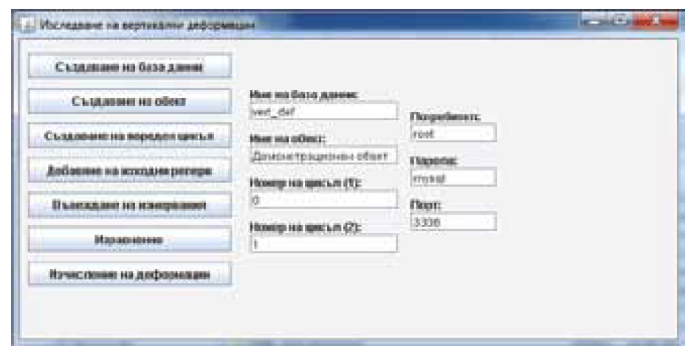


Рис. 3. Графический интерфейс пользователя

Разработанная программа обладает следующей функциональностью:

∙ создание базы данных для хранения данных конкретного объекта;

∙ добавление цикла измерений;

∙ ввод измерений, уравнивание сети и сохранение результатов данного цикла;

∙ вычисление деформаций между заданными циклами.

Заключение. Тестирование системы с реальными данными показывает, что такой способ интеграций систем действительно увеличивает производительность системы и существенно сокращает время ее разработки. Преимущество передачи матричных вычислений к Matlab становится существенным в случае обработки больших сетей с большими матрицами поскольку время умножения и обращения матриц в среде Matlab существенно не увеличивается в отличие от вычислительных процедур, реализованных в Java на основе стандартных алгоритмов.

# Использование сетевых геоинформационных технологий в рамках проекта европейского сотрудничества Upgrade Black Sea Scene

Аннотация. Статья посвящена вопросам использования сетевых геоинформационных технологий при разработке веб‑ГИС‑приложений. Дается описание и анализ применения технологий в рамках проекта европейского сотрудничества Upgrade Black Sea Scene.

Введение. Увеличение объемов разнородных тематических данных, расширение сферы использования информационных технологий, постоянно возрастающая потребность в пространственной информации, совместно с быстрорастущим рынком геоинформационных технологий определили современные тенденции развития и использования информационных сервисов, разрабатываемых для объектов различного территориального охвата. Реализация научных проектов и выполнение практических задач напрямую зависит от объема и качества используемой информации, корректности методик анализа и технологий представления данных. Поэтому не только геоинформационное сообщество и крупные игроки на рынке предоставления информационных услуг, но и государственные и межгосударственные организации вынуждены разрабатывать технологии в рамках различных проектов, ориентированных на поиск, сбор и доступ к информации [1]. В качестве примера можно привести следующие международные проекты:

SeaDataNet и SIMORG — крупнейшие европейские проекты, основными задачами которых являются совместимость и унификация международных и европейских стандартов метаданных, обеспечение доступа к распределенным базам морских и экологических данных, (http://www.seadatanet.org);

EDIOS — European Directory of the Oceanobserving System (http://www.seadatanet.org/metadata/edios);

CaspInfo — Caspian Sea Environmental And Industrial Data & Information Service — проект создания инфраструктуры Каспийского региона (http://www.caspinfo.net) и ряд других проектов [2].

Среди исключительно российских проектов необходимо отметить проект ЕСИМО (Единая государственная Система информации об обстановке в Мировом океане, http://www.esimo.ru) [3]. ЕСИМО отвечает за интеграцию разнородной и распределенной информации об обстановке в Мировом океане; обеспечивает федеральные, региональные и местные органы РФ, организации, осуществляющие морскую деятельность, комплексной информацией об обстановке в Мировом океане; взаимодействует с зарубежными информационными системами морской направленности и обеспечивает доступ к их ресурсам. Обмен и интеграция информацией происходит на основе сетевой модели взаимодействия и сервисно-ориентированных геоинформационных технологий.

В связи с тем, что объем пространственных данных ЕСИМО очень велик и каждый день появляются новые данные, большое внимание в системе уделяется разработке метаданных. ЕСИМО придерживается стандарта ISO 19115:2003 «Geographic information — Metadata». Состав и точность описания характеристик объектов метаданных зависит от уровня управления данными. Таким образом, метаданные на данные международного уровня отличаются от метаданных на данные уровня организации.

Характерной особенностью таких проектов является использование сетевых технологий, так как только эти технологии предоставляют единственное верное решение по доставке информации от производителей к потребителю, объединяя в единую логическую структуру распределенные источники информации. «Важная особенность компьютерной сети состоит в том, что она является, с одной стороны, высокоскоростной транспортной средой, а с другой — совокупностью распределенных информационных ресурсов» [4].

Современные сетевые ГИС-приложения разрабатываются на основе рекомендаций OGC (Open Geospatial Consortium). Открытый геопространственный консорциум является негосударственной некоммерческой организацией, созданной ведущими компаниями-разработчиками программного обеспечения и аппаратуры в области геоинформатики и дистанционного зондирования. В отличие от ISO, где стандарты утверждаются голосованием, в OGC спецификации утверждаются методом консенсуса, и их исполнение является добровольным. Наиболее важными спецификациями OGC являются WMS, WMF, WMC, CAT и ряд других.

WebMapService (WMS) — спецификация интерфейса картографических веб‑служб, выдающих клиентскому приложению растровое изображение карты, сформированное на основе его запроса. Это наиболее известная и широкоиспользуемая спецификация OGC.

Web Feature Service (WFS) — другой вид картографической веб‑службы, возвращающей, в отличие от WMS, набор векторных объектов. Назначение WFS — дать клиентскому приложению возможность создавать многослойные карты, в которых слои берутся из разных источников.

Web Coverage Service (WCS) — служба, аналогичная WFS, но ориентированная на передачу «покрытий» — сплошных распределений какого-либо признака в пространстве. Как правило, это растровые данные. Она также позволяет дополнить картографические изображения WMS слоями нового типа, которые можно сочетать с базовой картой.

Catalog Interface (CAT) — спецификация схемы каталога геоинформационных ресурсов и протоколов доступа к нему. Доступ к каталогу может осуществляться из различных приложений для поиска геоинформационных ресурсов и просмотра их характеристик.

Постановка проблемы. Проект Upgrade Black Sea Scene (UBSS) является логическим продолжением проекта Black Sea Scene (BSS), который стартовал в 2005 г. И закончился в 2008 гг. в рамках Шестой Рамочной европейской программы (Sixth Framework Programme for Research and Technological Development). Основной целью проекта BSS, который объединял 25 партнеров из стран черноморского региона и стран Европейского сообщества, было создание научной сети ведущих научно-исследовательских институтов, университетов, экологических и общественных организаций из стран Черноморского региона.

Такая научная сеть способствовала созданию и развитию виртуальной исследовательской инфраструктуры данных, которая поддерживается участниками проекта для улучшения сбора и идентификации данных, доступа, обмена, индикации качества данных и всестороннего использования информации о Черном море. Основная задача созданной научно-исследовательской сети состояла в стимуляции научного сотрудничества, обмене знаниями и опытом, повышении эффективности использования данных при решении научных и практических задач. Другой, не менее важной задачей, являлось согласование региональных и государственных стандартов метаданных и стандартов управления данными с принятыми европейскими стандартами. Это необходимое условие для обеспечения эффективного обмена информацией внутри черноморского региона и на европейском уровне в рамках крупнейшего европейского проекта по созданию инфраструктуры морских пространственных данных SeaDataNet (http://www.seadatanet.org).

В качестве основных задач проекта UBSS, как логического продолжения проекта BSS, были определены следующие задачи (http://www.blackseascene.net):

∙ расширение существующей научно-исследовательской сети, как следствие — улучшение регионального охвата информацией;

∙ существенное улучшение качественных характеристик, инструментов и технологий в области управления данными и информации путем принятия и реализации новейших технологий и стандартов проекта SeaDataNet;

∙ увеличение числа локальных тематических баз данных, которые могут быть запрошены и использованы в режиме реального времени через разработанные веб‑сервисы инфраструктуры;

∙ расширение существующей базы метаданных за счет добавления информации новыми партнерами;

∙ принятие стандартных методик для определения контроля качества наборов данных, включенных в инфраструктуру, а также определение принципов совместимости и согласованности наборов данных, которые будут добавлены;

∙ использование инновационных сетевых технологий визуализации данных.

Как уже было сказано выше, решить эти задачи можно исключительно посредством использования сетевых технологий. Портал проекта UBSS должен был стать точкой входа для пользователей независимо от уровня их подготовки и профессиональной направленности. Это могут быть и представители разных специальностей ученого сообщества, руководители разного уровня, представители общественных организаций, любой пользователь, который заинтересован в получении информации о черноморском регионе. Портал проекта должен обеспечить доступ к тематическим базам данных и обеспечить визуализацию пространственной информации. Особое внимание уделялось разработке такого интерфейса, который был бы интуитивно понятен любому пользователю.

Кафедра картографии и геоинформатики географического факультета МГУ принимала активное участие в реализации трех основных блоков создаваемой системы — сервисы баз данных, пользовательский интерфейс и ГИС‑приложения. Опыт их разработки, реализации и внедрения излагается в данной статье.

Сервисы баз данных. Разработка сервисов баз данных представляла собой главную и технически наиболее трудоемкую задачу. Требования к возможностям сервисов включали поиск и доступ необходимой информации, фильтрацию и сортировку данных, механизмы автоматического обновления информации. Можно выделить следующие этапы реализации поставленной задачи.

1. Создание локальных баз данных в структуре, определенной требованиями поставщиков информации (каталогов).

2. Разработка механизма обновления информации между локальными базами данных и каталогов.

3. Разработка пользовательских инструментов поиска, фильтрации, сортировки и визуализации данных, включая пространственный запрос.

В качестве исходных источников информации были использованы широкоизвестные европейские базы данных EDMO (European Directory Marine Organizations, http://www.seadatanet.org/edmo) — профильная адресная база данных обо всех организациях, участвующих в морских исследованиях; EDMERP (Directory Marine and Environmental Research Projects) — европейский каталог научных проектов по широкому спектру дисциплин; EDMED (European Directory Marine and Environmental Datasets) — европейский каталог морских и экологических данных.

Выбор именно этих баз данных обусловлен несколькими факторами:  
а) информация, полученная из этих каталогов, востребована пользователями;  
б) имеется разработанная технология, позволяющая постоянно актуализировать информацию;  
в) надежность информации обеспечивается соглашениями между владельцами данных, предоставляющих информацию, и европейской комиссией, финансировавшей создание этих баз.

При проектировании базы данных было принято решение содержать на портале локальные копии баз данных EDMO, EDMERP, EDMED. При этом в выгрузку включались только данные, имеющие отношение к Черноморскому региону. Выбор в пользу локальных копий был сделан в целях снижения нагрузки на основные сервера, поддерживающие европейские базы данных, и для уменьшения времени обработки запросов. Такая схема организации доступа к информации хороша тем, что с минимальными затратами мы получаем доступ к уже разработанным базам данных с соответствующими сервисами, отвечающими за ввод новой информации, актуализацию данных, контроль качества содержания.

На следующем этапе была выполнена разработка сервиса, отвечающего за обмен информации между порталом проекта UBSS и серверами баз данных EDMO, EDMED, EDMERP. На языке Java было разработано специальное приложение, которое посылает запрос с сервера UBSS на сервера европейских каталогов информации. В свою очередь, на серверах европейских каталогов активирован специальный веб‑сервис, позволяющий принимать запросы через сеть Интернет и выдавать результаты обработки запросов. После обработки запроса серверы европейских каталогов высылают отфильтрованную информацию приложению портала в виде набора XML‑файлов (Extensible Markup Language). Приложение, использующее технологию XSLT (eXtensible Stylesheet Language Transformations), интегрирует полученную информацию в локальную, управляемую СУБД MySQL, базу данных. Такая технология может быть использована для доступа к любым базам данных. Таким образом, на портале UBSS была размещена наиболее актуальная информация европейских баз. Сам процесс обмена информацией между серверами можно автоматизировать, определив расписание, по которому он будет запускаться, либо по анализу специальных маркеров, определяющих обновление информации в европейских каталогах. На рис. 1 отображена принципиальная схема организации доступа к данным европейских каталогов через портал UBSS.

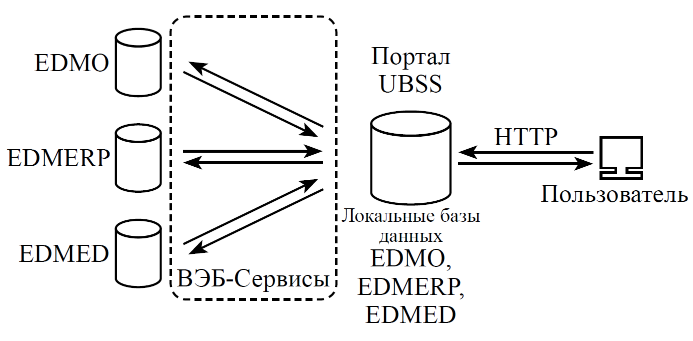


Рис. 1. Принципиальная схема доступа к европейским базам данных через портал UBSS

Последним этапом разработки сервисов баз данных стало создание пользовательских инструментов выборки, фильтрации и сортировки данных. Используя разработанные инструменты, пользователь может запросить, например, только украинские организации из базы данных EDMO. Другим примером такого сервиса может быть запрос пользователя о проектах, которые проходили в определенный интервал времени по определенной тематике из базы данных EDMERP. Или же это может быть запрос пользователя о данных по температуре, которые были получены в определенный период времени на заданном участке черноморского региона.

Отдельным блоком стоит специальный инструмент поиска, разработанный для поиска информации по всем существующим базам данных — «cross database search». Этот инструмент был разработан для облегчения поиска информации пользователем портала UBSS. Результат выполнения работы этого инструмента представляется в табличном виде с указанием конкретной базы данных, где была найдена информация.

В отличие от EDMERP, базы данных EDMO и EDMED имеют пространственную привязку, поэтому дополнительно стояла задача показать результаты запросов пользователя не только в табличном виде, но и в виде отдельных слоев в ГИС-приложении.

Разработка сервисов, обеспечивающих работу с базами данных, тесно взаимосвязана с разработкой пользовательского интерфейса портала проекта UBSS.

Пользовательский интерфейс. Целью разработки пользовательского интерфейса портала UBSS являлось создание интерактивного веб‑интерфейса, отвечающего за взаимодействие между порталом и пользователем в части использования сервисов баз данных и ГИС-приложения.

Основными критериями при создании веб‑интерфейса являлись:

∙ интуитивная простота интерфейса, легкость в использовании, четкая структура и логичность расположения управляющих элементов;

∙ минимальное использование графических элементов в целях повышения скорости загрузки приложения в окне браузера;

∙ все элементы оформления и содержательная информация должны быть выдержаны в единой цветовой гамме, позволяющей пользователю выделить содержательную часть и отдельно элементы управления;

пользователь не должен устанавливать никакого дополнительного программного обеспечения для работы с порталом UBSS, используя стандартное программное обеспечение для просмотра страниц в Интернете.

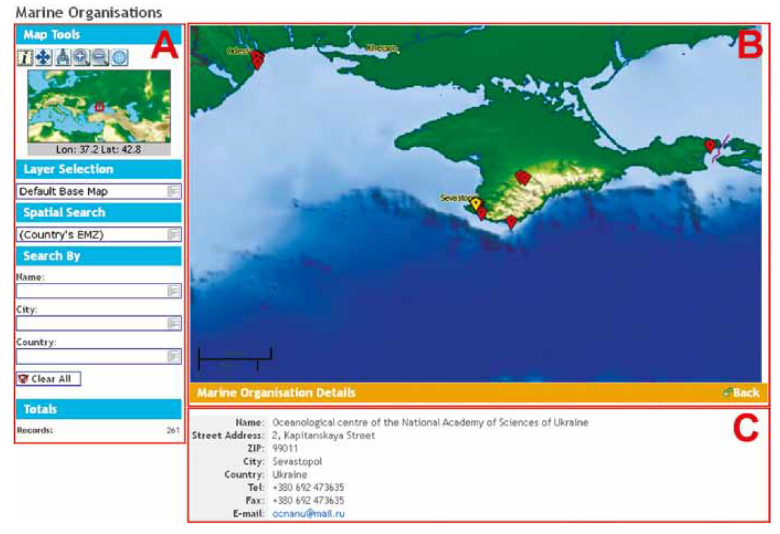


Рис. 2. Пользовательский интерфейс приложения: A — окно блока управления элементов; B — окно ГИС‑приложения; C — окно результата табличного вида

Можно выделить три основных блока в компоновке разработанного пользовательского интерфейса (рис. 2). Первый блок, находящийся в левой стороне страницы, представляет собой набор управляющих элементов, отвечающих за функциональность ГИС‑приложения и поля базы данных, по которым пользователь может проводить выборку и фильтрацию данных. Элементы, отвечающие за функциональность ГИС‑приложения, позволяют осуществлять выбор слоя базовой карты, функции уменьшения/увеличения масштаба, идентификацию объекта, функции пространственной выборки объектов и прочие элементы. Также в этом блоке компоновки находится навигационная карта.

Вторым блоком компоновки пользовательского интерфейса является окно ГИС‑приложения, в котором отображаются выбранный пользователем базовый слой и тематический слой, являющийся результатом обработки фильтрации баз данных. Результатом обработки запросов к базе EDMO являются точечные объекты, представленные на базовой карте специальным символом. Результат обработки запросов к базе EDMED представляет собой совокупность прямоугольников, определяющих пространственный охват данных. Так как база EDMERP не имеет объектов с пространственной привязкой, при работе с этой базой данных окно ГИС-приложения не отображается.

Третьим блоком компоновки интерфейса является окно табличного представления результатов запросов к базе данных. В этом окне, расположенным справа под окном ГИС‑приложения, результаты запросов к базам данных представлены в табличном, простом для восприятия информации, виде. Пользователю предоставляется возможность осуществлять сортировку данных, переход между страницами результата выборки, получать полную информацию о выбранном из таблицы объекте. Выбранные в окне табличного представления записи выделяются в виде специальных значков тематического слоя в окне ГИС-приложения, и наоборот. Таким образом, обеспечивается полная взаимосвязь между базой данных и ГИС‑приложением, когда любые действия пользователя с базой данных, визуализируются ГИС‑приложением.

Интерфейс отдельного раздела портала «Каталог карт» несколько отличается от общего интерфейса портала. Это связано с тем, что цель данного каталога не поиск информации, а представление уже готовых картографических материалов, хранящихся на доверенных серверах, посредством сервисов WMS (Web Map Service) и WFS (Web Feature Service). Окно табличного представления информации состоит из списка серверов и их описания. При выборе пользователем требуемого сервера окно ГИС‑приложения заполняется списком доступных картографических материалов, представленных в виде отдельных слоев ресурса. Есть возможность сделать видимым и активным не только отдельный слой, но и несколько слоев. Это позволяет не только визуализировать информацию, но и совершать простейшую операцию оверлея. Например, на карту температуры, представленную в виде отдельного слоя, можно наложить карту течений, также представленную в виде отдельного слоя.

Организационно пользовательский интерфейс реализован как совокупность сервлетов — приложений, выполняемых на серверной стороне и обрабатывающих запросы к локальным базам данных, и клиентской части, представленной в виде набора динамически создаваемых страниц, визуализирующих результаты обработки запросов. Создание страниц происходит на основе языков DHTML (Dynamic HyperText MarkUp Language) и JavaScript.

Для проверки стабильности внешнего вида интерфейса в разных средах было проведено тестирование в наиболее распространенных веб‑браузерах — Microsoft Internet Explorer, Mozilla FireFox, Google Chrome. Необходимость подобного тестирования обусловлена вероятностью неодинакового отображения веб‑страниц вследствие неоднозначной интерпретации исходного HTML‑кода. Результаты тестирования показали стабильность интерфейса во всех перечисленных средах.

ГИС‑приложение. С точки зрения пользователя, ГИС‑приложение является незаменимым инструментом для интерактивной работы с пространственными данными в сети интернет. Разработка полнофункционального Веб‑ГИС‑приложения представляет собой комплексную задачу, при решении которой необходимо учитывать обширный набор технических спецификаций, требований, владеть программированием, в особенности сетевым. По определению, Веб‑ГИС — это геоинформационная система, ориентированная на ее использование в компьютерных сетях типа интернет или интранет (в их веб‑службах), обычно реализованная на ГИС‑сервере [5].

ГИС‑приложение, которое было разработано в рамках проекта UBSS, использует бесплатный картографический сервер MMS (Minnesota Map Server). Другим фактором, определившим выбор именно этого картографического сервера, стала существенная поддержка MMS геоинформационным сообществом, включая OGC (Open Geospatial Consortium). Дополнительно решалась задача интеграции JavaScript-библиотеки Open Layers в инструментарий картографического сервера. Данная библиотека позволяет очень быстро и легко создать веб‑интерфейс для отображения картографических материалов, представленных в различных форматах и расположенных на различных серверах.

Вследствие того, что основная задача ГИС‑приложения портала UBSS заключалась в визуализации пространственных данных по результатам работы пользователя с базами данных, особое внимание уделялось именно разработке разных способов и функциональных возможностей инструментов представления пространственной информации. В ГИС‑приложении были реализованы основные функции навигации: сдвиг, масштабирование по охвату, масштабирование по объекту. Инструменты скомпонованы в интуитивно понятную панель.

Следующий набор функций ГИС‑приложения представляют инструменты для работы со слоями. Пользователю предоставляется возможность выбора базовых слоев. Это может быть специально разработанный для проекта набор слоев данных, либо наборы слоев, полученных с открытых картографических серверов, таких как Metacarta (labs.metacarta.com) или Demis (www2.demis.nl). Результаты запросов к базам данных EDMO, EDMED автоматически отображаются в виде отдельных активных тематических слоев: полигональных для базы EDMED и точечных для базы EDMO объектов. При работе с разделом «Каталог карт» пользователю предоставляется больше функциональных возможностей работы со слоями — управление видимостью, активация, показ легенды активного слоя, управление которых реализованы посредством пиктограмм, представляющих свойства слоя.

Параллельно решалась задача показа результатов работы с базами данных. Если задача визуализации точечных объектов представляется достаточно простой, то визуализация системы полигональных перекрывающихся объектов, представляющих собой графическое представление результатов обработки запросов к базе данных EDMED, требовала оригинального решения. Пользователь видит только совокупность полигональных объектов, поэтому ему трудно определить, для какого именно объекта он хочет получить дополнительную информацию. С другой стороны, также трудно определить, сколько и какие другие данные попадают в зону интереса. Поэтому было решено, что насыщенность цвета находится в функциональной зависимости от количества нижележащих объектов. Чем насыщенней цвет, тем больше объектов перекрывают или накладываются друг на друга. Такое решение позволило существенно облегчить пользователю восприятие графического представления результатов обработки запросов к базе EDMED.

Отдельным блоком стояла задача реализация спецификаций WMS (Web Map Service) и WFS (Web Feature Service). На сервере портала UBSS реализована следующая архитектура. Портал UBSS выступает в качестве WMS клиента, то есть он может получать с других серверов картографическую информацию через WMS‑сервис. Это реализовано в виде доступа к базовым слоям портала с серверов Metacarta и Demis, которые выступают в качестве WMS‑серверов. Визуализация картографической информации в разделе «Каталог карт» также осуществляется через WMS‑сервис.

С другой стороны, пользователь или другие информационные сетевые ресурсы могут использовать ряд тематических слоев, которые хранятся на сервере портала. В этом случае портал UBSS выступает в качестве WMS‑сервера. Аналогичная ситуация и с реализацией спецификаций WFS. Интеграция WMS и WFS‑сервисов стала возможной благодаря тому, что картографический сервер MMS поддерживает реализацию этих спецификаций (рис. 3). На представленной иллюстрации в качестве базовых выступают слои, загруженные через WMS‑сервис с портала Demis, а в качестве тематических слоев представлены картографические материалы с сайта Института геофизики (Грузия).

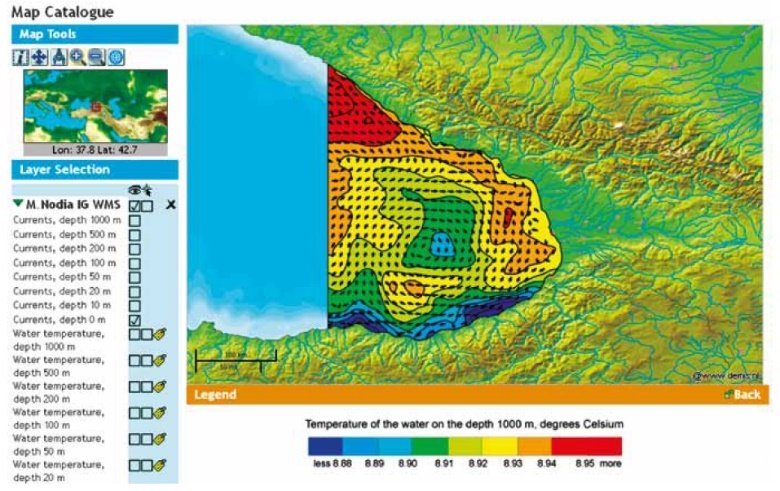


Рис. 3. Пример использования WMS‑сервиса

Также в ГИС‑приложении UBSS-портала доступны такие стандартные функции как проведение измерений, вычисление координат курсора и текущего масштаба.

Заключение. Веб‑технологии обеспечивают удобный обмен картографическими источниками и пространственными данными, открытый доступ к разнообразным ГИС‑приложениям для любого пользователя, находящегося в любой точке земного шара в режиме реального времени, и используются при организации доступа к пространственным данным геопорталов. Востребованность и постоянно растущая доступность географически привязанной информации (наборов готовых данных, изображений и т. д.) на сегодняшний день определяют направление развития картографии.

Проект UBSS является примером успешного применения веб‑технологий и сервисов, ориентированных на решение задачи доставки информации к потребителю посредством сетевых технологий. Аналогичная технологическая схема обмена информацией может быть применена при создании специализированных тематических баз данных и быть продуктивной при решении широкого круга образовательных, научных и практических задач. Поддержка встроенного картографического сервиса позволяет не только значительно облегчить восприятие пространственной информации. Посредством реализации спецификаций WMS и WFS осуществляется обмен картографической информацией, таким образом, происходит создание единого геоинформационного пространства.

Выполнение рекомендаций OGC позволило без дополнительных технологических затрат интегрироваться в Европейскую инфраструктуру пространственных данных, облегчить поиск информации и доступ к данным.

# Исследование алгоритмов автоматизированной генерализации линейных объектов

Аннотация. Для получения оптимальных результатов при создании любых карт требуется разработка некоторых правил и критериев по выбору масштаба карты, а также проведению генерализации. Масштаб карты обычно выбирается традиционно — это ручной процесс, но он требует от картографа глубоких знаний об объекте картографирования, что делает данный процесс субъективным. Поэтому при создании самой карты и осуществлении генерализации ее содержания, а также уменьшении ручного труда, возникает необходимость автоматизации процесса. Рассмотрены альтернативные решения путей генерализации линейных объектов на основе ГИС‑технологий.

Введение. Понятию автоматизированной генерализации, пришедшего из обычной картографии, стали уделять все большее внимание как и в географических информационных системах (ГИС), так и в традиционных областях картографирования. Для сокращения производственного времени в картосоставлении и обеспечения объективности создаваемых картографических образов требуется исследование процессов и алгоритмов автоматизации генерализации карт. Настоящее исследование, посвященное генерализации дорожной сети для территории г. Дехрадун (Индия), проведено с использованием ArcGIS.

При картографировании все географические объекты и явления подвергаются масштабированию. Для получения оптимальных результатов создатели карт разрабатывают ряд правил и критериев, но они изменяются в процессе создания карты, оставаясь темой многих исследований и поныне.

Установление оптимального масштаба карты обычно производится традиционно — это ручной процесс, он требует от картографа глубоких знаний об объекте картографирования, опыта. Это делает процесс субъективным, так как можно создать сколько угодно карт по масштабу с заранее установленными требованиям. В этом случае полезно пользоваться автоматической генерализацией, которая основана на картографических правилах, работающих с установленными цензами и нормами и дающих стабильный результат.

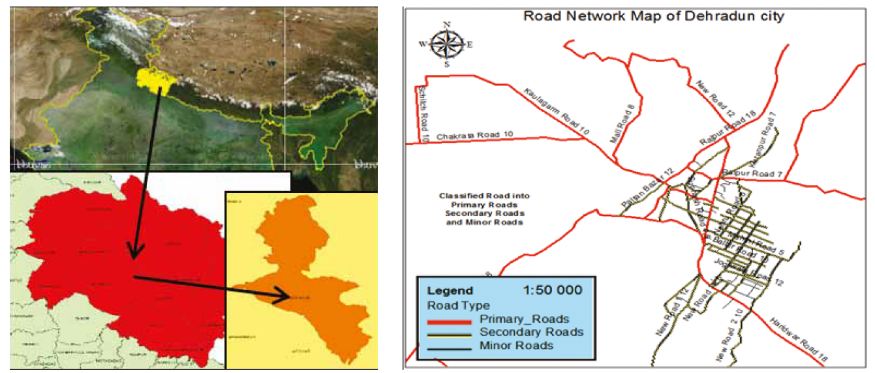


Рис. 1. Карта изучаемой территории

Можно выделить три главных элемента карты — дорожную сеть, объекты гидрографии и населенные пункты. Наши исследования направлены на генерализацию дорожной сети. В качестве объекта исследования были взяты земли г. Дехрадун, штата Уттаркханда, Индия. Город имеет разные типы дорог: национальные (главные), небольшие второстепенные, шириной около 18 м, и прочие дороги, шириной меньше трех метров. В качестве исходной информации была использована карта-путеводитель г. Дехрадун, для оценки результатов генерализации дорог — топографическая карта Уттаранчал масштаба 1:50 000. 53 Д/З (рис. 1). В статье изложены результаты сравнительного анализа программного обеспечения генерализации дорог, ее операторы и алгоритмы, которые применяются во многих странах мира, а также их преимущества и недостатки.

Цели и задачи. Целью исследования была разработка инструмента автоматической генерализации дорожной сети и составление генерализованной карты на район исследования, которое было проведено в несколько этапов:

1) подготовка карты дорожной сети на основе карты-путеводителя г. Дехрадун масштаба 1:20 000;

2) уточнение и корректировка контуров содержания карты по данным спутниковой съемки CaroSat‑1 (панхроматические снимки, ПАН) и IRSP6 LISSIVMX (многоспектральные снимки) и объединение данных;

3) изучение и оценка существующих операторов и алгоритмов ГИС генерализации дорог;

4) разработка программного обеспечения для осуществления генерализации дорожной сети и составления электронной карты дорожной сети;

5) оценка качества генерализованной карты дорожной сети.

Методика. Для достижения цели исследования разработана методика (рис. 2), суть которой состоит в следующем. Сначала были изучены алгоритмы генерализации дорожной сети. Далее был произведен сбор данных и оцифрована топографическая карта на основе карты-путеводителя г. Дехрадун масштаба 1:20 000 и обновлена по спутниковым данным (IKONAS). Сбор данных разделен на два этапа: получение имеющихся топографических карт; спутниковых снимков и других данных для исследования; сбор информации наземного контроля.

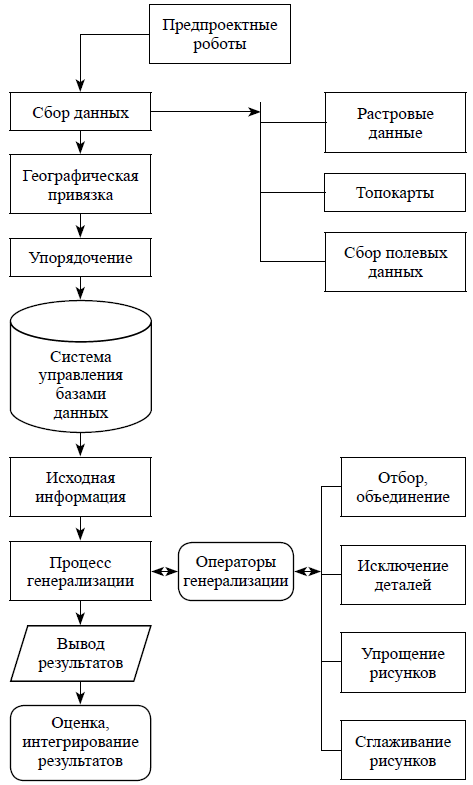


Рис. 2. Схема методологии исследования

На основе оцифрованных данных была создана база данных ГИС, которая в дальнейшем будет служить входным источником для выбора операторов программного обеспечения генерализации дорог. Для организации базы геоданных были использованы топографические карты, спутниковые изображения и материалы полевых исследований. Методология состояла из трех этапов: сбор данных; создание базы геоданных; разработка алгоритмов для генерализации дорожной сети и использование их для получения разномасштабных карт.

Программное обеспечение генерализации было разработано при помощи языка программирования «Питон» и графического интерфейса пользователя GUI, созданного по программе PyQt. Разработанная программа генерализации дорог имеет четыре оператора — выбор, устранение, упрощение и сглаживание. Результаты были оценены на основе качественных и количественных показателей путем сопоставления полученных данных с топографической картой части Уттаракханда в масштабе 1:50 000 (номер листа 53J/3).

В ходе исследований векторизация топографической карты и анализ информации были произведены с использованием программного обеспечения ArcGIS. Обработка изображения и его слияние с картой осуществлялись с использованием программы ERDASImagine. Алгоритмы были разработаны программой Python 2.7.3 и графическим интерфейсом (GUI) с использованием QtDesigner 4.8.4 (рис. 3).

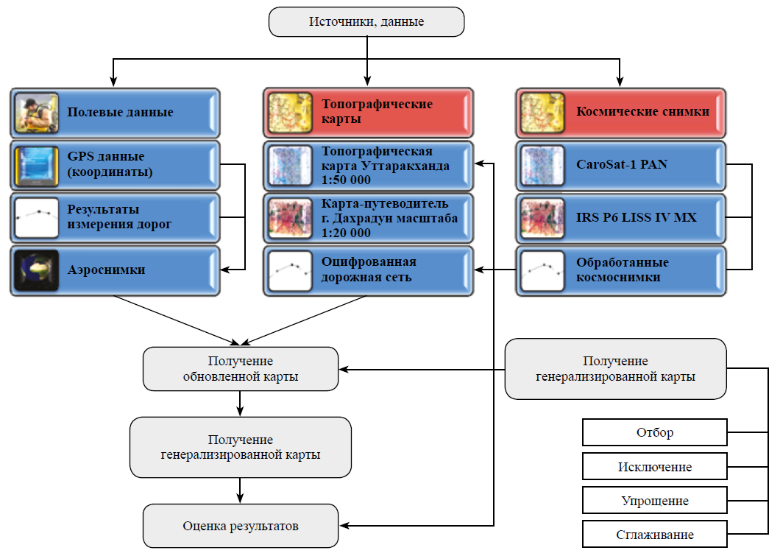


Рис. 3. Блок‑схема исследования

После того как процесс сбора данных был завершен, дороги были классифицированы (упорядочены) в соответствии с их характеристиками, которые помогают пользователю генерализовать данные с помощью разработанного программного обеспечения. Основываясь на исследования, можно выделить следующие важные характеристики для классификации: ширина дороги; дорожное покрытие; открытые области; области поселение; равнины; гористый и крутой ландшафт.

Исследование было выполнено по характеристике «Расположение и покрытие дорог», т. е. по области расположения дороги — равнинам и покрытию — асфальтом.

Необходимо подчеркнуть, что удачная классификация дорог делает выбор более легким и точным. Этот шаг идентифицирует объекты, которые помещены в группы в соответствии с аналогичными свойствами. Это также уменьшает сложность и улучшает организацию карты, поэтому дороги были упорядочены в классы с шириной в пределах от 3 до 18 м: главные (основные), второстепенные и прочие дороги. Только определенные дорожные классы были отобраны для включения в целевой масштаб. Выбор выполнен на основании признаков дорог — ширина, тип (основной, второстепенный и прочие) и длина.

Картографируемые объекты, которые не имеют отношения к назначению карты, например, сегмент дороги — короче определенной длины или небольшие сегменты дороги, которые могут ухудшить содержание итоговой карты и не являются значимыми для представления на карте, — могут быть устранены. В данном случае, в качестве целевого масштаба 1:50 000 все дороги, имеющие длину менее 100 м, не были представлены в итоговой карте. С другой стороны, используя атрибут данных дорог, можно в дальнейшем устранить детали, полученные на мелкомасштабных картах.

Для упрощения линии используется алгоритм Дугласа‑Пекера, более оптимальный для этих целей. Его позиционная точность является самой высокой среди других алгоритмов упрощения линии, и отобранная область исследования также относительно небольшая, как и продолжительность обработки, которая также очень непримечательная (рис. 4).

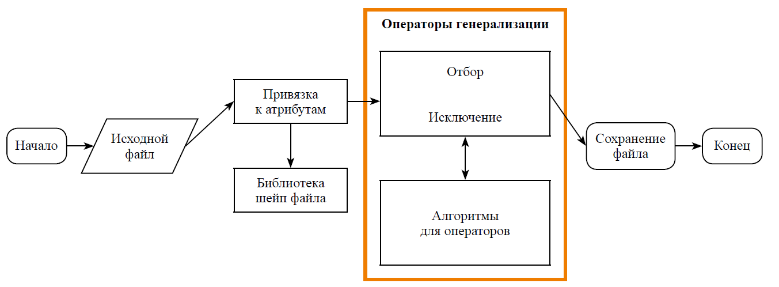


Рис. 4. Блок‑схема операторов программы генерализации

К сокращению планирования времени, уменьшению загрузки памяти компьютера, более быстрому преобразованию вектора к растру и более быстрой векторной обработке приводит сглаживание. Операция по сглаживанию линии позволяет уменьшить количество острых углов, улучшить эстетику изображения. Здесь использовался алгоритм сглаживания кривой Безье, который дает сглаживание линии после алгоритма упрощения линии, что придает линиям более эстетичный вид (рис. 5). Алгоритм кривой Безье широко используется во многих коммерческих программных обеспечениях, особенно часто в ArcGIS ESRI.

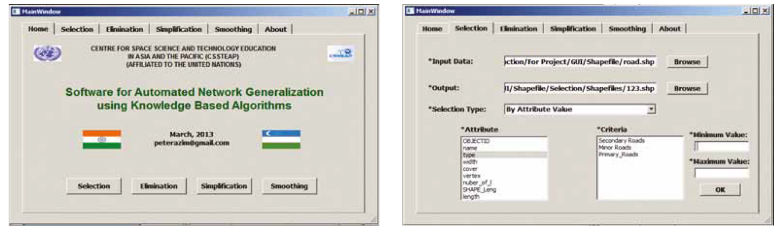


Рис. 5. Интерфейс инструмента генерализации

Как было упомянуто выше, инструмент генерализации дорожной сети состоит из двух операторов — «Выбора», «Устранения» — и из двух алгоритмов «Упрощение линии», «Сглаживание линии». Они были разработаны в качестве заключения исследования и служат для обсуждения операторов программного обеспечения и точности, которую они имеют.

GUI (Графический интерфейс пользователя) был разработан, чтобы достигнуть целей одновременно. В GUI имеется шесть команд — «Домой», «Выбор», «Устранение», «Упрощение», «Сглаживая» и «О программе» (рис. 6).



Рис. 6. Ошибки после алгоритма упрощения линий Дугласа-Пекера с допуском 0.00024

Операторы «Выбор» и «Устранение» выбирают линейные объекты согласно таким признакам дорог, как класс, ширина и длина. Для масштаба 1:50 000 дороги длиной меньше чем 200 м считаются второстепенными и в конечном счете были удалены.

Для упрощения линейных особенностей считается, что алгоритм упрощения линии Дугласа−Пекера дает лучшие результаты в позиционной точности, а также при искажении форм. Некоторые исследователи считают этот алгоритм трудоемким, однако, в наборе данных, которые используются в нашем случае, достигается упрощение дорог в течение незначимого времени. При эксперименте алгоритма с несколькими допусками лучший результат получен для карты масштаба 1:50 000, и он был взят в качестве допуска упрощения дорожной сети для текущего масштаба карты.

Накладывание целевой карты и упрощенных результатов карты было сравнено, и в большинстве случаев карта, полученная из алгоритма Дугласа–Пекера, соответствовала топографической карте, однако, в некоторых местах несходство было замечено (рис. 7).

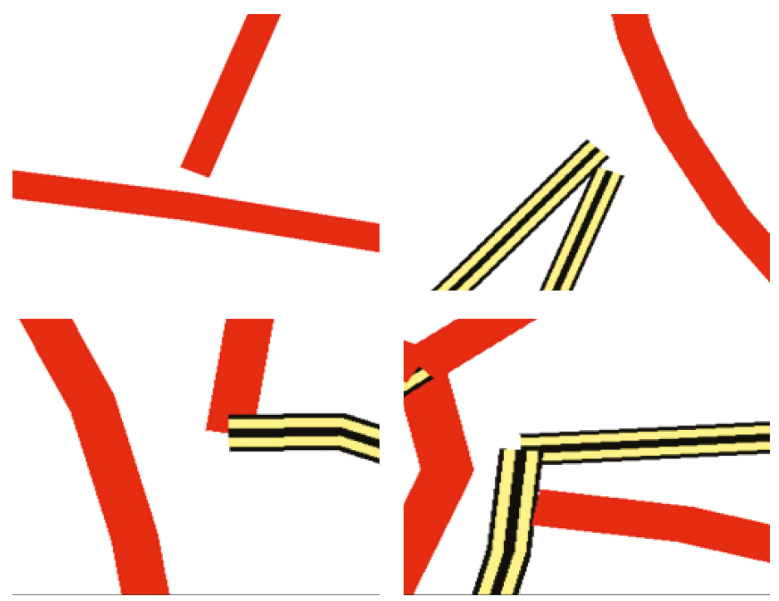


Рис. 7. Ошибки после алгоритма сглаживания Безье

Оператор сглаживания дает сглаженные линии, так как после алгоритма упрощения дороги выглядят прямыми линиями. На карте они должны быть гладкими, что и делает карту более читабельной и приятной в понимании.

После алгоритма сглаживания Безье, судя по эстетическому виду и читаемости карты, результаты были более чем достаточными для заданной целевой карты. Тем не менее, некоторые позиционные ошибки были замечены (рис. 8).

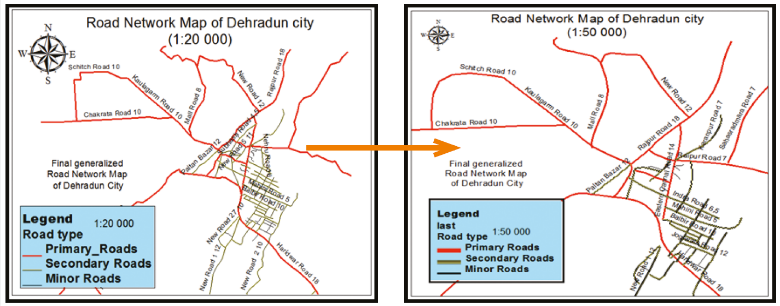


Рис. 8. Сравнение оцифрованной карты дорожной сети масштаба 1:20 000 (а) с генеральной картой масштаба 1:50 000 (б)

В итоге, после сравнения оцифрованной карты дорожной сети с конечной генерализированной картой были между собой сравнены (рис. 9) стало видно, что конечная генерализированная карта выглядит более читаемой по сравнению с цифровой картой. Генерализированную карту также сравнили с отсканированной «топографической картой Уттаранчала 1:50 000, 53 J/3», (таблица). Обнаруженные различия между ними были проверены.

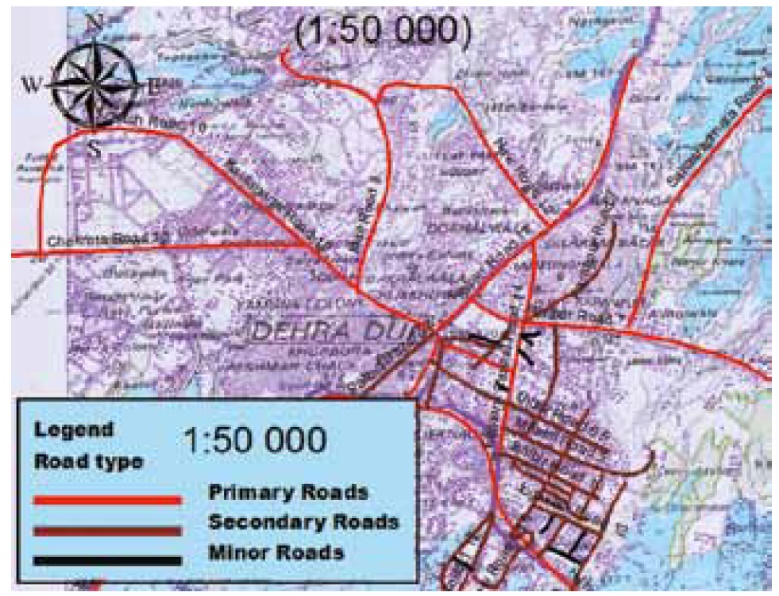


Рис. 9. Наложение генерализации карты на топографическую карту Уттаранчала

|  |  |  |  |
| --- | --- | --- | --- |
| Дороги | Топографическая карта | Итоговая карта | Различия |
| Главные  Второстепенные  Прочие | 15  23  5 | 18  28  9 | 3  5  4 |

Анализ таблицы показывает, что «Главные дороги» и «Второстепенные дороги», главным образом, соответствуют на картах, однако, есть некоторые незначительные позиционные ошибки. Топографическая карта имеет 15 «Главных дорог» и 23 «Второстепенные дороги», а в генерализированной карте их, соответственно, 18 и 28. С другой стороны, больше различий произошло с «Прочими дорогами», 5 и 9 соответственно. Эти несоответствия могут быть объяснены устарелостью топографической карты, поскольку классификация дорог была выполнена на основании полевых данных.

Выводы. Таким образом, можно заключить, что в существующих инструментах генерализации не хватает интерактивной системы генерализации. Кроме того, были изучены несколько алгоритмов генерализации линейных объектов, и в итоге алгоритм упрощения линейных объектов Дугласа−Пекера презентовал самую высокую позиционную точность.

Разработанный автоматизированный инструмент генерализации дорожной сети состоит из четырех операторов: выбор; исключение; упрощение; сглаживание. Созданный графический интерфейс пользователя для инструмента генерализации подключен к операторам. Результаты показывают, что большинство генерализированных дорог, особенно главные и второстепенные, совпадают с дорогами топографической карты, несколько различий встречаются в прочих дорогах. Эти различия могут быть объяснены использованием устаревшей топографической карты. Генерализированная карта была наложена на топографическую карту масштаба 1:50 000, где можно наблюдать некоторые геометрические и позиционные погрешности. Это связано с параметрами использованных операторов упрощения и сглаживания.

В заключение можно отметить, что дистанционное зондирование и географические информационные технологии в сочетании с картографическими знаниями не только могут обеспечить инструментом генерализацию дорожной сети, но и способствовать автоматизированию процесса генерализации любых видов карт быстро и точно. Этот инструмент может использоваться во многих организациях картографии и быть удобным для навигационных систем, поскольку они требуют быстрого преобразования карт различных масштабов.