



7. Панюкова, Т.А. Маршруты с локальными ограничениями / Т.А. Панюкова // Вестник Южно-Уральского государственного университета. Серия: Математическое моделирование и программирование. 2010. Вып. 5. № 16(192). С. 58–67.

8. Панюкова, Т.А. Маршруты с локальными ограничениями: алгоритмы и программная реализация / Т.А. Панюкова, И.О. Алферов // Прикладная информатика. 2013. № 1(43). С. 80–90.

9. Szeider S. Finding Paths in Graphs Avoiding Forbidden Transitions // Discrete Applied Mathematics, 2003. № 126. P. 261–273.

10. Кристофидес, Н. Теория графов / Н. Кристофидес – М.: Мир, 1978. 432 с.

Д.Д. Мальчиков, Н.А. Остроглазов

ПЛАГИН ВИЗУАЛИЗАЦИИ ДОСТОПРИМЕЧАТЕЛЬНОСТЕЙ ГОРОДА НА ЭЛЕКТРОННОЙ КАРТЕ

(Самарский университет)

Для реализации плагина «Достопримечательность» необходима функциональность геоинформационной системы, в которой в каждый слой добавляются на карту новые географические объекты [1]. Таким образом, чем больше геообъектов разных типов, тем больше тематических слоев загружено в систему (рисунок 1).

Для создания плагина необходимо выбрать предметную область, затем разработать ER-модель базы данных [2], которая определяет отношения между объектами плагина. Рассмотрим предметную область «Достопримечательность» [3, 4]. Основными объектами плагина будут культурные и исторические достопримечательности, которые можно посетить клиентам. Эти объекты обладают семантическими и географическими параметрами. Дополним базу данных таблицами, которые дополняют тематический слой и делают его более информативным, такие как «Адрес», «Организация» и другие.

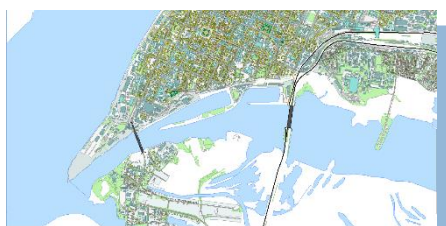
Основная сущность плагина «Достопримечательность» имеет параметры:

- идентификатор (*id*) – который позволяет уникализировать достопримечательность и, если понадобится, то выбрать нужный геообъект по данному идентификатору из базы данных;
- имя (*name*) – название достопримечательности;
- адрес (*adress_id*) – местоположение геообъекта, в данном случае это идентификатор сущности «Адрес», которая имеет множество реализующих её параметров, например «Название улицы», «Номер дома», «Перекрёсток» и другие;
- геометрия (*geometry_id*) – геометрия геообъекта, которая реализует зону покрытия достопримечательности, например, небольшая скульптура в парке, так и целая площадь;



- описание (*restriction_id*) – описание семантики геообъекта достопримечательности – это текст, в котором описываются характеристики объекта;
- тип достопримечательности (*sighttype_id*) – тип геообъекта, добавляемый пользователем, скульптура, музей, площадь или другие;
- организация (*organization_id*) – физическое или юридическое лицо, занимающееся обслуживанием этого объекта.

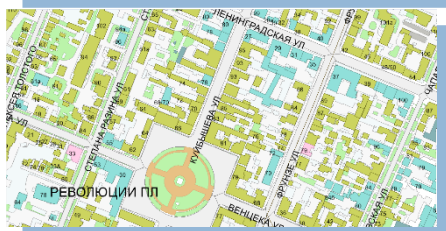
ER-модель базы данных показана на рисунке 2.



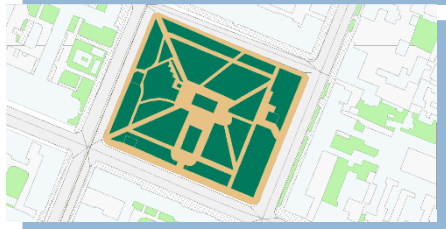
Слой гидрографии



Слой зданий и строений



Слой подписей



Слой парков и скверов

Рисунок 1 – Тематические слои «ITSGIS»

Основная функциональность плагина заключается в создании, удалении и изменении геообъектов достопримечательностей, построения туристических маршрутов, моделирование движения транспортных средств (туристических автобусов) [5]. Добавлены дополнительные функции: фильтрация объектов и получение списка созданных геообъектов.

Для работы с объектами внутри плагина создадим классы одноименные с таблицами базы данных, такие как *Sight*, *SightType*, *Restriction*, добавим конструктор классов и параметры, которые описывают объект. Часть конструктора класса представлена на рисунке 3.

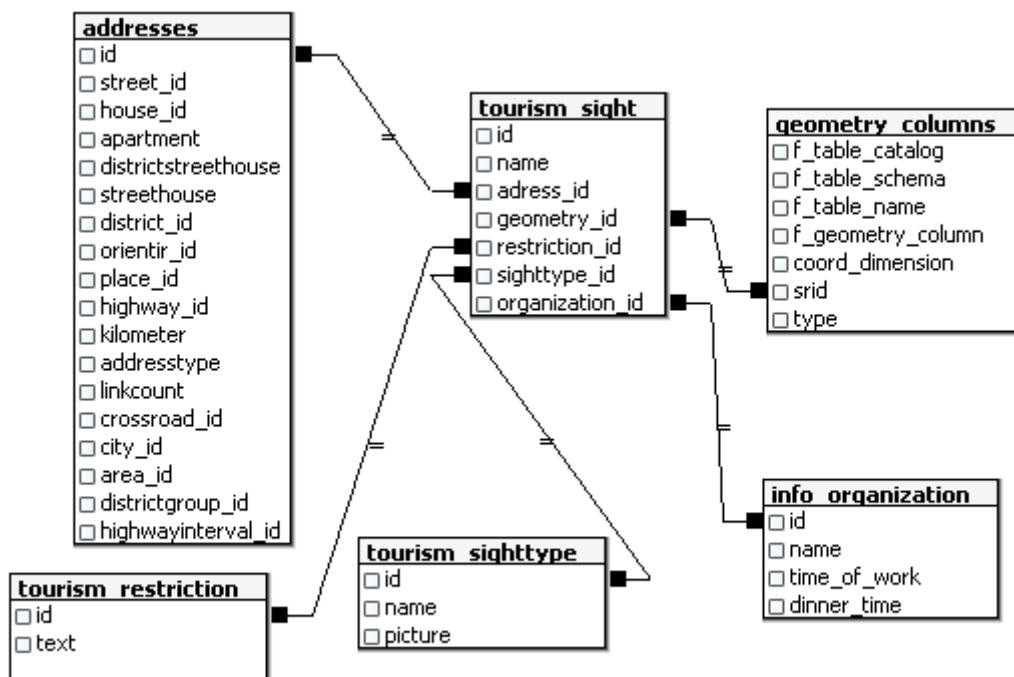


Рисунок 2 – ER-модель базы данных

```
/// <param name="type">Тип достопримечательности</param>
public Sight(string name, SightType type, Address adr, FeatureObject feat)
    : this()
{
    Name = name;
    SightType = type;
    Address = adr;
    Feature = feat;
}
#endregion

#region DataProperties

/// <summary>
/// Название достопримечательности.
/// </summary>
public string Name { get; set; }

/// <summary>
/// Тип достопримечательности.
/// </summary>
public SightType SightType { get; set; }

/// <summary>
```

Рисунок 3 – Часть конструктора класса Sight

Связь между классами плагина и базой данных обеспечивает ORM – технология программирования, связывающая разработку базы данных с концепциями объектно-ориентированного проектирования [6], создавая «виртуальную» объектную базу данных [7]. В качестве ORM используем NHibernate. Создадим Mapping файл формата XML, с помощью которого соединим отношения классов плагина с таблицами в базе данных (рисунок 4).



Визуальный вывод достопримечательностей, как объектов, происходит на соответствующий тематический слой. Для этого необходимо создать класс, реализующий визуализацию объекта, его геометрию и размеры. В этом случае это будут иконки нужного типа, в виде небольших PNG или JPEG изображений со своими координатами [8]. Результатом работы класса визуализации является слой с графическими объектами, за счет того, что слой не полностью заполнен графическими объектами, можно увидеть предыдущий слой, поверх которого визуализируется текущий (рисунок 5).

```
<?xml version="1.0" encoding="utf-8"?>

<hibernate-mapping xmlns="urn:hibernate-mapping-2.2">
  <class lazy="false" name="ITS.Core.Tourism.Domain.Sight, ITS.Core.Tourism" table="tourism_sight">
    <id name="ID" column="id" type="long" unsaved-value="0">
      <generator class="hilo" />
    </id>
    <property column="name" name="Name" type="string" not-null="true" />
    <many-to-one name="Address" column="address_id" class="ITS.Core.Domain.Districts.Address, ITS.Core" cascade="none"/>
    <many-to-one name="Organization" column="organization_id" class="ITS.Core.Domain.Organizations.Organization, ITS.Core" cascade="none"/>
    <many-to-one name="Feature" column="geometry_id" class="ITS.Core.Domain.FeatureObjects.FeatureObject, ITS.Core" cascade="all-delete-orphan"/>
    <many-to-one name="Restriction" column="restriction_id" class="ITS.Core.Tourism.Domain.Restriction, ITS.Core.Tourism" cascade="none"/>
    <many-to-one name="SightType" column="sighttype_id" class="ITS.Core.Tourism.Domain.SightType, ITS.Core.Tourism" cascade="none"/>
  </class>
</hibernate-mapping>
```

Рисунок 4 – Mapping классов и таблиц

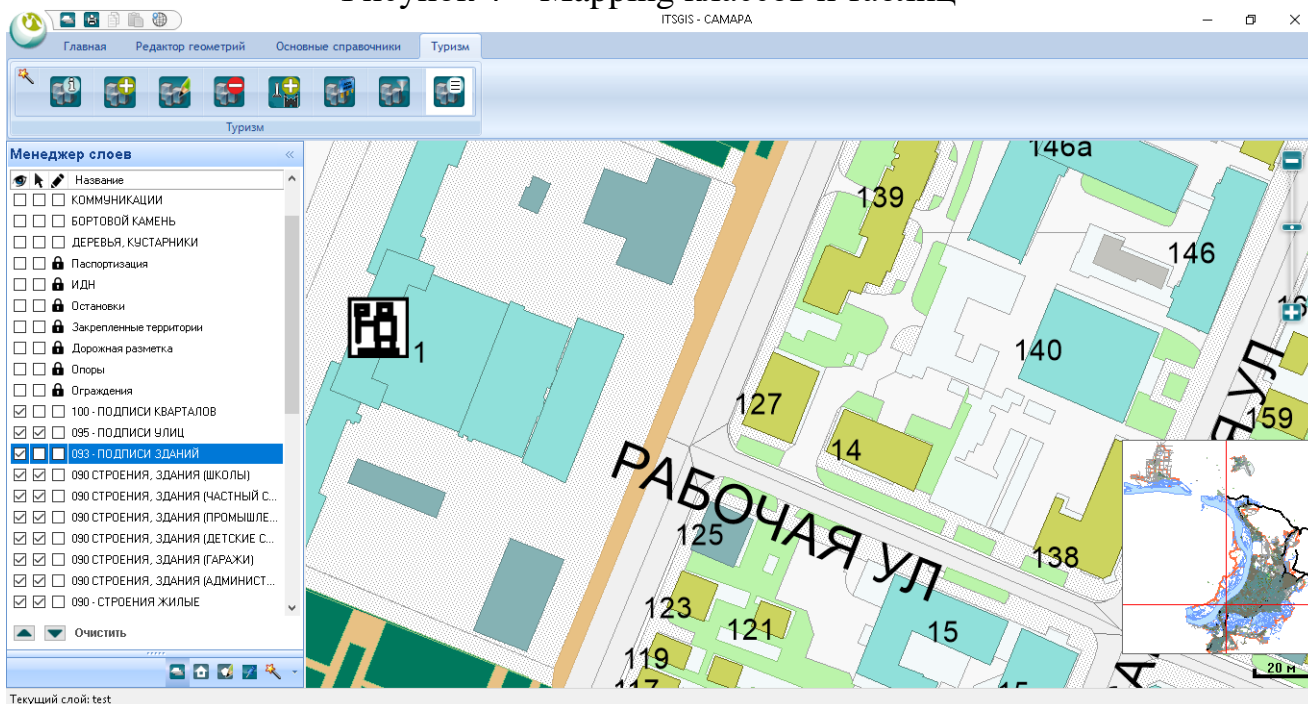


Рисунок 5 – Отображение слоев «ITSGIS»

В результате разработки плагина расширяется функционал «ITSGIS», добавляются новые функции, позволяющие создавать объекты тематического слоя «Достопримечательности», изменять, удалять и выводить достопримечательности на электронную карту, фильтровать их по признакам.

Литература

1. Интеллектуальная транспортная геоинформационная система ITSGIS. Плагины / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Са-



винов, В.А. Ключников, Д.А. Алтухов, Н.А. Остроглазов, А.Н. Имамутдинов. – Самара : Интелтранс, 2016. – Т.2. – 217 с.

2. Коннолли, Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли, К. Бегг. – 3-е изд. – М.: Вильямс, 2003. – 1436 с.

3. Достопримечательность [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/Достопримечательность>.

4. Рцинская, И.И. Путеводитель как инструмент конструирования региональных достопримечательностей (вторая половина XIX – начало XX в.) // Вестник Московского университета. Серия 19. Лингвистика и межкультурная коммуникация. – 2011. – № 1. – С. 74-93.

5. Михеева Т.И. Моделирование движения в интеллектуальной транспортной системе / Вестник Самарского государственного аэрокосмического университета – Самара: СГАУ, 2004. – С. 118-126.

6. Михеева Т.И., Михеенков И.Е. Программная таксономия – основа для создания гипермедийных обучающих систем // Информационные технологии, 1998. – №8. –С. 40-43.

7. ORM [Электронный ресурс] – URL: <https://ru.wikipedia.org/wiki/ORM>.

8. Ноубл, Дж. Рецепты программирования / Дж. Ноубл, Т. Андерсон, Г. Брэйтуэйт, М. Казарио, Р. Третола, Flex 4. – БХВ-Петербург, 2011. – С. 548. – 720 с.

С.В. Михеев

СИНТЕЗ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ПРИ УПРАВЛЕНИИ ТРАНСПОРТНОЙ ИНФРАСТРУКТУРОЙ НА ОСНОВЕ ПАТТЕРНОВ

(Самарский национальный исследовательский университет
имени академика С.П. Королева,
Группа компаний «ИнтелТранС»)

Введение

Для моделей систем поддержки принятия решений при управлении транспортной инфраструктурой характерно использование сложных динамически реконфигурируемых структур объектов и их ассоциаций. Методология синтеза систем управления транспортной инфраструктурой основана на развиваемом подходе к системному анализу и компьютерному моделированию сложно-организованных систем. При разработке использованы оригинальные приемы объектно-ориентированного проектирования на основе теории паттернов. Конструирование и реконструирование таких паттернов требует инструментов, обеспечивающих эти процессы универсальными средствами создания и динамической модификации объектов.