



ИНТЕЛТРАНС
Г Р У П П А К О М П А Н И Й

IT & TRANSPORT

ИТ & ТРАНСПОРТ

Сетевое издание

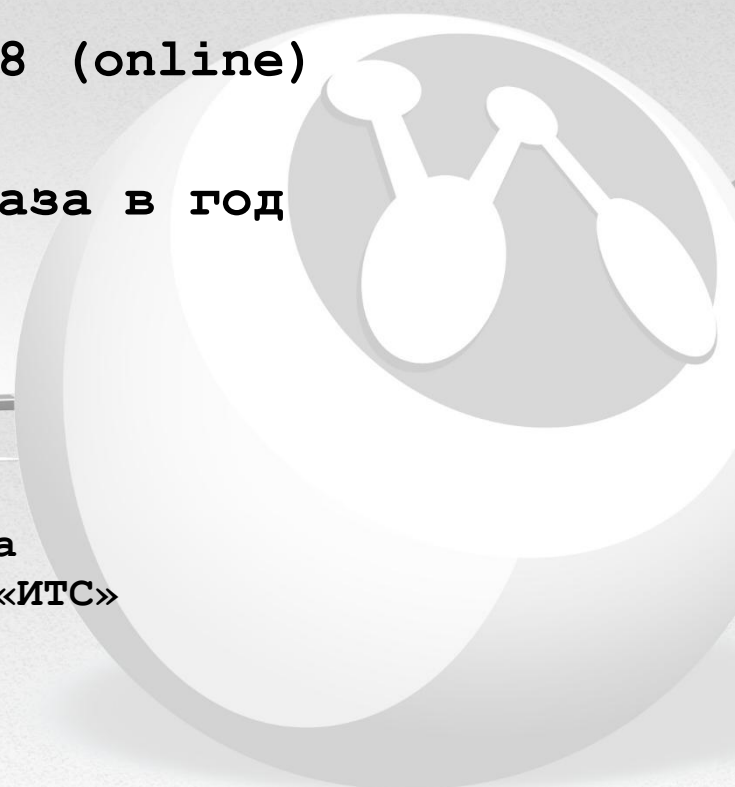
№ 2 (26) – 2024

ISSN 2949–5288 (online)

Выходит: 4 раза в год

Самара
ООО «НПЦ «ИТС»

2024





ИНТЕЛТРАНС
Г Р У П П А К О М П А Н И Й

И Т & Т Р А Н С П О Р Т

И Т & Т Р А Н С П О Р Т

Направления :

**1.2.2. МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ЧИСЛЕННЫЕ
МЕТОДЫ И КОМПЛЕКСЫ ПРОГРАММ (технические науки)**

**2.3.1. СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА
ИНФОРМАЦИИ (технические науки)**

**2.3.3. АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ
ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ (технические науки)**

**2.3.5. МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ВЫЧИСЛИТЕЛЬНЫХ МАШИН, КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ
СЕТЕЙ (технические науки)**

**Главный редактор Михеева Т. И.
доктор технических наук, профессор,
заслуженный деятель науки и образования РФ**

**Самара
ООО «НПЦ «ИТС»
2024**

IT & Transport / ИТ & Транспорт : под ред. профессора Т. И. Михеевой. – Самара, ООО «НПЦ «ИТС», 2024, №2 (26). – 66 с. ил.

ISSN: 2949-5288

В сборнике представлены научные статьи, содержащие результаты исследований в следующих предметных областях: системный анализ, управление и обработка информации, интеллектуальные транспортные системы, геоинформационные системы, информационные технологии, базы данных и знаний, системы искусственного интеллекта, цифровая обработка изображений, управление транспортными процессами, развитие транспортной инфраструктуры, автоматизация и управление технологическими процессами и производствами, математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей, математическое и компьютерное моделирование.

Предназначен для научно-технических работников, инженеров, аспирантов и студентов, обучающихся по специальностям, связанным с информационными технологиями, транспортными системами и процессами.

Авторская позиция и стилистические особенности публикаций полностью сохранены.

Главный редактор – Т. И. Михеева доктор технических наук, профессор,
заслуженный деятель науки и образования РФ

Редакционная коллегия:

Бурдин А.В.	доктор технических наук, доцент
Гераськин М.И.	доктор экономических наук, профессор
Жанказиев С.В.	доктор технических наук, профессор
Заболотнов Ю.М.	доктор технических наук, профессор
Зырянов В.В.	доктор технических наук, профессор
Иващенко А.В.	доктор технических наук, профессор
Минниханов Р.Н.	доктор технических наук, профессор
Прохоров С.А.	доктор технических наук, профессор
Филиппова А.С.	доктор технических наук, профессор
Хайтбаев В.А.	доктор экономических наук, профессор
Зеленко Л.С.	кандидат технических наук, доцент
Золотовицкий А.В.	кандидат технических наук, доцент
Михеев С.В.	кандидат технических наук, доцент
Сапрыкин О.Н.	кандидат технических наук, доцент
Сапрыкина О.В.	кандидат технических наук, доцент
Федосеев А.А.	кандидат технических наук, доцент
Шопин А.Г.	кандидат технических наук

Редакция: Чекина Е.В.

ISBN 978-5-6049535-5-6©

Сетевое издание «ИТ & Транспорт» зарегистрировано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций; регистрационный номер: серия ЭЛ № ФС 77–85276 от 30.05.2023

e-mail: itsspc@yandex.ru

© Адрес страницы сайта: <http://itstransport.ru>

© ООО «НПЦ «ИТС», 2024

УДК 004.02

*Михеева Т.И., Смолев А.М.***ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ СИСТЕМЫ «ITSGIS»
В ЗАДАЧАХ ОПТИМИЗАЦИИ ИЗМЕРЕНИЯ ИНТЕНСИВНОСТИ***Самара, Самарский университет имени академика С.П. Королёва
ИнтелТранс*

В данной статье приводится описание типичных задач по измерению интенсивности в геоинформационной системе ITSGIS. Эти задачи эффективно решаются посредством взаимодействия с объектами, составляющими улично-дорожную сеть (УДС) города. Объекты УДС привязываются к улицам карты. Дается информация о паттернах проектирования, позволяющих оптимизировать процесс расчета интенсивности за счет многократного использования решений подзадач, входящих в этот расчет. Раскрываются методы и средства, которые были применены при реализации полученных решений в системе ITSGIS.

Ключевые слова: проектирование, улично-дорожная сеть, интенсивность, геоинформационная система, информационный эксперт, состояние, адаптер.

Введение

Паттерны проектирования интеллектуальной транспортной геоинформационной системы «ITSGIS» являются основным компонентом в ее глобальной архитектуре. В данной статье рассматриваются общеметодологические паттерны проектирования, применяемые при разработке программного обеспечения и используемые для решения задач оптимизации процесса функционирования транспортной сети. Паттерн в нотации визуальных языков моделирования, таких как UML, является призмой, через которую можно смотреть на решаемую задачу. Также они позволяют производить миграцию знаний о методах решения из одной области знаний в другие. В данной статье рассматривается, каким образом общеметодологические паттерны проектирования, применяемые при разработке программного обеспечения, используются для решения задач оптимизации процесса функционирования транспортной сети.

В мировой практике управления транспортной инфраструктурой в крупных городах активно используются технологии интеллектуальных транс-

портных геоинформационных систем «ITSGIS». Система представляет собой комплекс интегрированных средств управления дорожно-транспортной инфраструктурой, применяемых для решения задач организации дорожного движения на основе современных информационных технологий, а также организации информационных потоков, включающих в себя данные о функционировании транспортной инфраструктуры в реальном режиме времени. Многоуровневая, сложноорганизованная «ITSGIS» представляет собой гибридную систему, состоящую из множества разнородных систем, объединенных для достижения единой цели, сложным образом взаимодействующих друг с другом – управляющих, классифицирующих, прогнозирующих, экспертных, принимающих решения или поддерживающих эти процессы.

Структурой, с которой постоянно сталкиваются проектировщики объектно-ориентированных приложений, являются паттерны проектирования. Используемый при разработке информационных систем любой паттерн проектирования представляет собой формализованное описание часто встречающейся задачи проектирования, удачное решение данной задачи, а также рекомендации по применению этого решения в различных ситуациях. Основан на использовании объектно-ориентированного подхода к проектированию сложных систем. Объект сочетает данные и процедуры для их обработки, называемые многообразными. Посылка запроса – единственный способ заставить объект выполнить операцию. Выполнение – единственный способ изменить внутреннее состояние объекта.

При декомпозиции системы на классы объектов приходится учитывать множество факторов: инкапсуляцию, глубину детализации, наличие зависимостей, гибкость, производительность, развитие, повторное использование. Паттерны проектирования помогают выявить не самые очевидные абстракции и объекты, которые могут их использовать. Следует подчеркнуть, что важным начальным этапом при работе с паттернами является адекватное моделирование рассматриваемой предметной области. Это является необходимым как для получения должным образом формализованной постановки задачи, так и для выбора подходящих паттернов проектирования.

Паттерн проектирования представляет собой именованное описание проблемы и ее решения, кроме того, содержит рекомендации по применению в различных ситуациях, описание достоинств и недостатков. Паттерны проектирования позволяют разными способами решать многие задачи, с которыми постоянно сталкиваются проектировщики объектно-ориентированных приложений. Любой паттерн проектирования, используемый при разработке информационных систем, представляет собой формализованное описание часто встречающейся задачи проектирования, удачное решение данной задачи, а также рекомендации по применению этого решения в различных ситуациях.

Интеллектуальная транспортная геоинформационная система «ITSGIS»

Синергетический эффект при проектировании «ITSGIS» проявляется в форме организационно обусловленного перехода от имманентности к синергии за счет последовательно расширяемой системной интеграции. Интегрированные «ITSGIS», как системы управления дорожно-транспортной инфраструктурой урбанизированной территории, призваны повысить уровень организации дорожного движения: улучшить характеристики улично-дорожной сети, усовершенствовать дислокацию технических средств организации дорожного движения, оптимизировать процесс управления транспортными потоками на всех фазах движения, уменьшая транспортные задержки, повышая безопасность движения.

Проектирование объектно-ориентированных программ – трудная задача, требующая одновременного решения множества проблем. Необходимо подобрать подходящие объекты, отнести их к различным классам, соблюдая разумную степень детализации, определить интерфейсы классов и иерархию наследования и установить существенные отношения между классами. Дизайн должен, с одной стороны, соответствовать решаемой задаче, с другой – быть общим, чтобы удалось учесть все требования, которые могут возникнуть в будущем. Необходимо избежать или, по крайней мере, свести к минимуму необходимость перепроектирования.

Во многих объектно-ориентированных системах можно встретить шаблоны, состоящие из классов и взаимодействующих объектов, с помощью которых решаются конкретные задачи проектирования, существующие во многих системах. Обобщение и классификация таких задач и наиболее удачных путей их решения привело к появлению паттернов проектирования.

Под паттерном проектирования будем понимать описание взаимодействия объектов и классов, адаптированных для решения общей задачи проектирования в конкретном контексте. Каждый паттерн описывает некую повторяющуюся проблему и ключ к ее разгадке, причем таким образом, что этим ключом можно пользоваться при решении самых разнообразных задач. Паттерн проектирования именуется, абстрагирует и идентифицирует ключевые аспекты структуры общего решения, которые и позволяют применить его для создания повторно используемого дизайна.

Паттерн «Абстрактная фабрика» в «ITSGIS» использовался для решения задачи создания семейства взаимосвязанных или взаимозависимых объектов без спецификации их конкретных классов. Решение данной задачи заключается в создании абстрактного (родового) класса, в котором объявлен интерфейс для создания конкретных классов. Примером использования паттерна

является класс «Измерение_Интенсивности» движения транспортных потоков, специфицируемый многими параметрами. Объектно-ориентированная модель класса будет определяться структурой наследования и агрегации (рисунок 1).

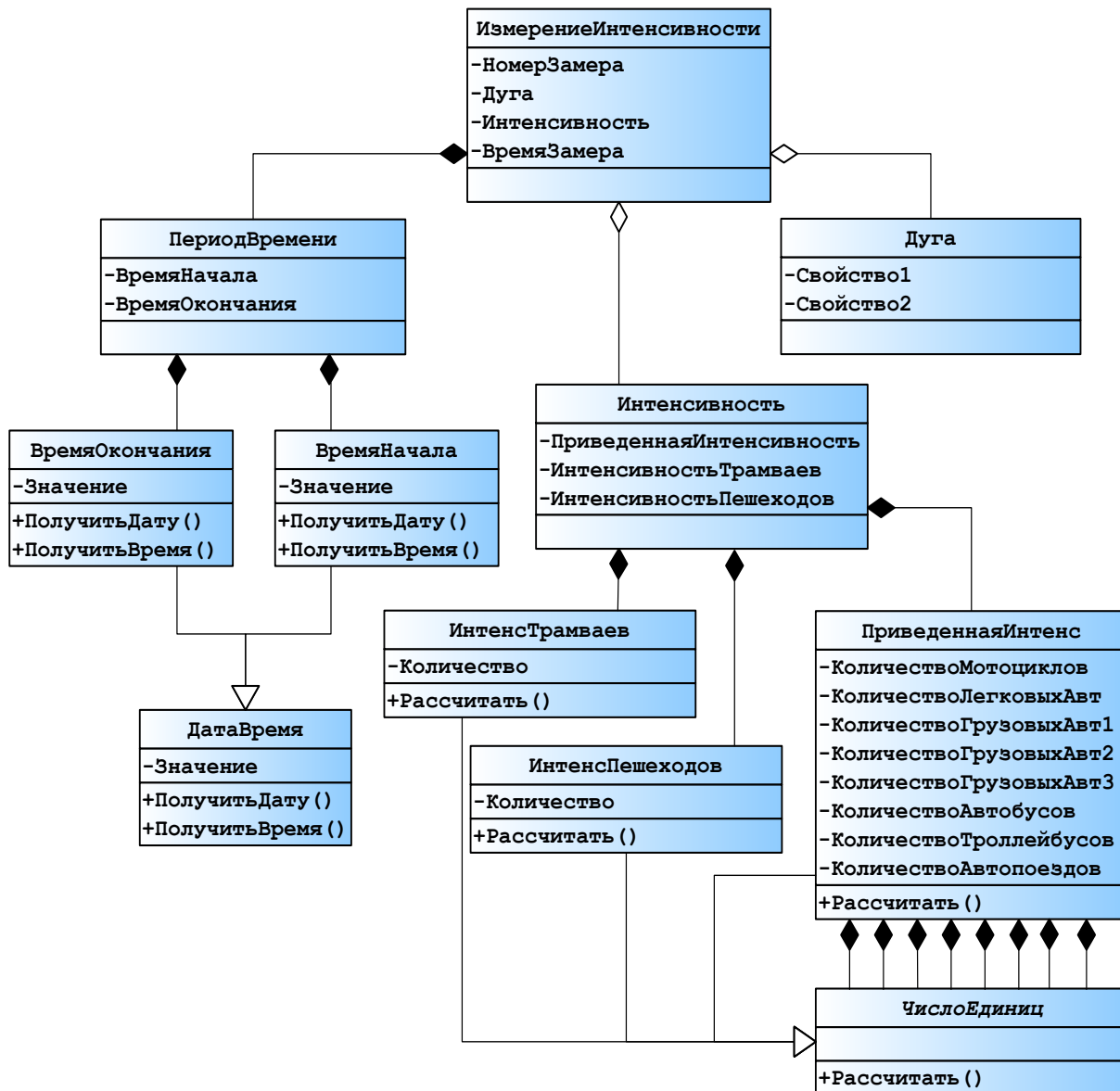


Рисунок 1. Диаграмма паттерна «Абстрактная фабрика»

при решении задачи измерения интенсивности

Построенная модель характеризуется наличием трех родовых классов: дуга графа улично-дорожной сети, на которой производится измерение интенсивности движения, количество единиц транспортных средств определенного вида, время измерения интенсивности, включающее дату, время начала и окончания измерения. Поскольку разные измерения могут конструироваться на одних и тех же объектах классов «Период_Времени», а объекты этих классов, в свою очередь, на одних и тех же объектах класса «Дата_Время»,

конструирование осуществляется на основе паттерна с использованием разделяемых страт, определяющих общие части для различных результатов измерений «расположенных» на дуге графа.

Натурные испытания и анализ результатов моделирования. Получение первичных данных осуществлялось на входных участках управляемого перекрестка. Длина контролируемого участка определена равной 100 м. В качестве сечения, в котором фиксировались движущиеся транспортных средств, покидающие перекресток, выбрана «стоп-линия», пересекаемая отбывающим автомобилем.

Для оценки управляющих режимов на каждом из подходов регистрировались следующие измеряемые величины:

- моменты времени прибытия и отбытия транспортных средств, начала и конца возобновления и запрещения движения;
- интенсивность движения, определяемая количеством транспортных средств за один светофорный цикл (рисунок 2);
- тип прибывающих транспортных средств;
- длина очереди, измеряемая количеством транспортных средств;
- направление движения транспортных средств, совершающих поворотное движение;
- транспортных средств, меняющие полосу движения в зоне контроля.

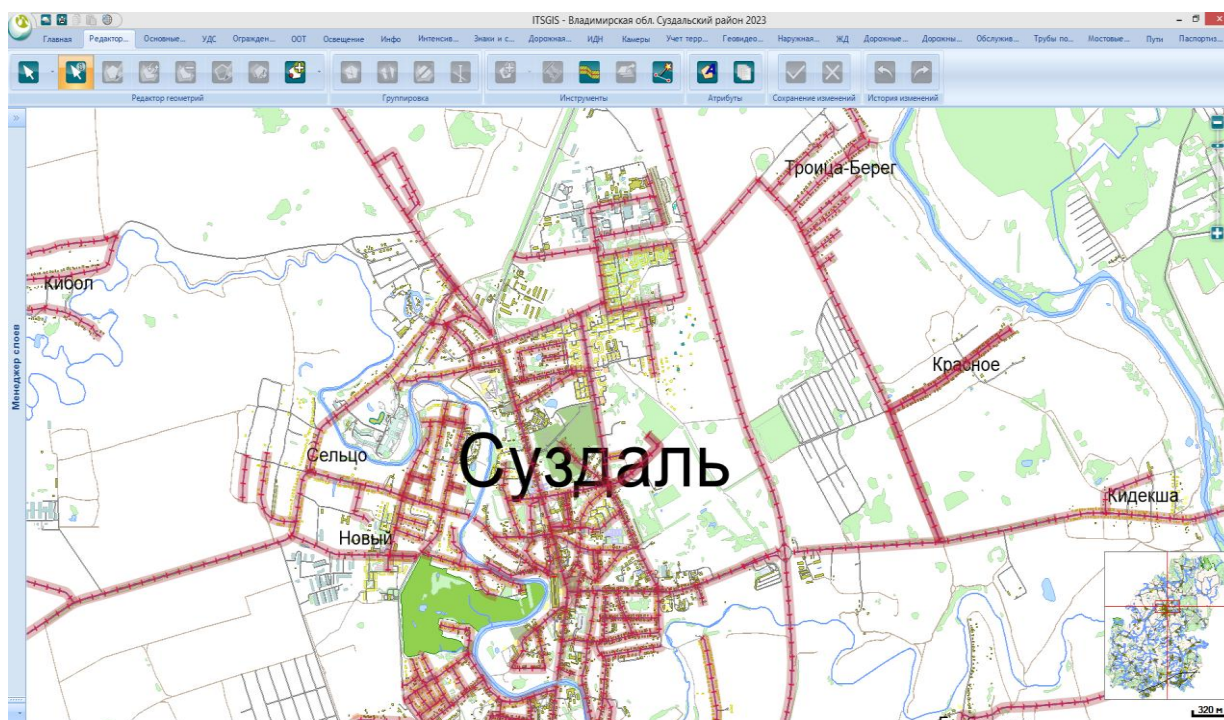


Рисунок 2. Участки улично-дорожной сети,
выгруженные в паспорта

Обработка результатов измерений интенсивности начинается с запуска паттерна работы со снимками и формирования снимка (рисунок 3).

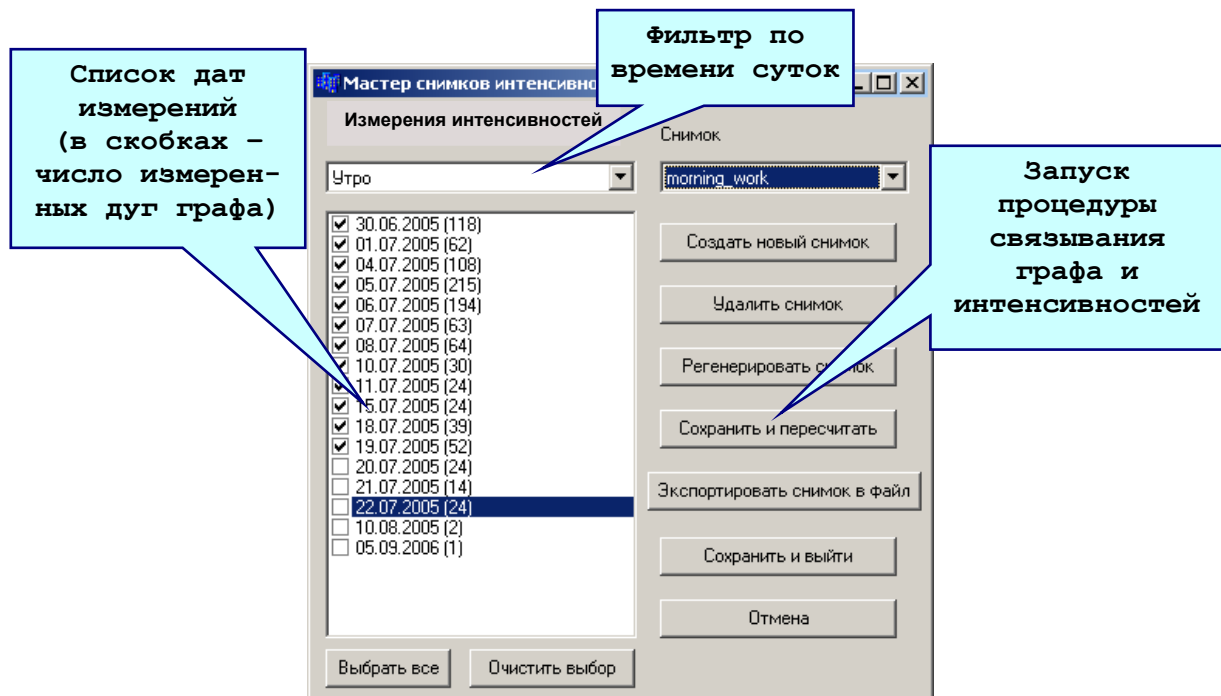


Рисунок 3. Экранная форма модуля обработки снимков

В процессе эксперимента выявлены характеристики на перекрестке и оценена адекватность имитационной модели:

- вид распределений интервалов времени прибывающих и отбывающих транспортных средств;
- интенсивность потока насыщения для каждого из направлений движения;
- линейность прибытия и отбытия транспортных средств в течение цикла;
- время задержки на автомобиль;
- число остановок на автомобиль;
- стандартное отклонение задержки на автомобиль;
- интенсивность потока;
- оптимальную последовательность переключения управляющих сигналов на каждом подходе.

Все измеряемые величины фиксировались наблюдателями, осуществляющими необходимые записи. При проведении натурного эксперимента были зарегистрированы моменты начала красного и зеленого сигналов, длина очереди в начале зеленого сигнала, момент пересечения «стоп-линии» транспортных средств, стоящих последними и очереди.

Паттерн «Информационный эксперт» подсистемы принятия решения

Подсистема принятия решения принимает в качестве исходных данных статический объект – граф улично-дорожной сети. При этом решение принимается на основании графа принятия решения. Т.к. исходные данные не модифицируемы и в то же время граф принятия решения должен без дополнительных модификаций отрабатывать все изменения, производимые в классе «Граф сети», то наиболее оптимальным был выбор в части реализации подсистемы принятия решения паттерна «Информационный эксперт» графа сети.

Данный паттерн может применяться, когда должна аккумулироваться, рассчитываться и храниться для дальнейшей обработки необходимая информация. В качестве решения выступает метод решения, при котором некоему классу, обладающему необходимой информацией, назначается обязанность аккумуляции, расчета, обработки оперативных данных системы. Паттерн допускает мультиклассовую реализацию, при которой «Информационный эксперт» реализуется на базе нескольких классов, связанных по данным.

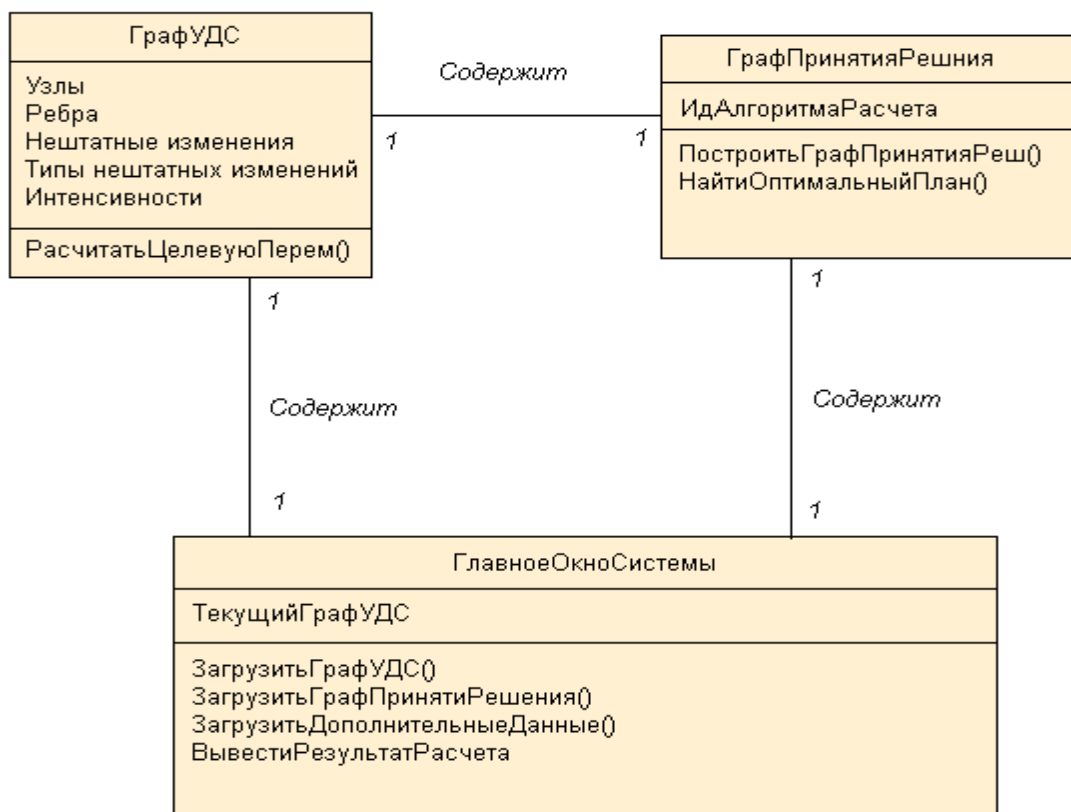


Рисунок 4. Диаграмма реализации паттерна «Информационный эксперт»

В рассматриваемой системе применен метод реализации посредством единственного класса. Выбран метод хранения необходимой информации в виде агрегации. Если говорить в терминах реализованных классов, то класс *TGraph*, содержащий информацию о состоянии улично-дорожной сети агре-

гируется классом *TSolveGraph* – графом принятия решения. Он хранит информацию об улично-дорожной сети (УДС), обрабатывает ее (поиск оптимального плана восстановления сети), и выдает результат в виде рекомендации по последовательности устранения нештатных изменений.

Класса *TSolveGraph* имеет в качестве двух своих главных методов методы *Построить_Граф_Принятия_Реш()* и *Найти_Оптимальный_План()*, которые являются главными расчетными методами данного класса.

Атрибут *Ид_Алгоритма_Расчета* – параметр, идентифицирующий метод поиска целевой переменной.

Паттерн «Состояние» множественной обработки данных

Данный паттерн варьирует поведение объекта в зависимости от его внутреннего состояния. Множественность обработки данных обозначает возможность применения различных алгоритмов при поиске значения переменной, характеризующей оптимальность улично-дорожной сети с точки зрения обеспечения пропускной способности потребителям сети.

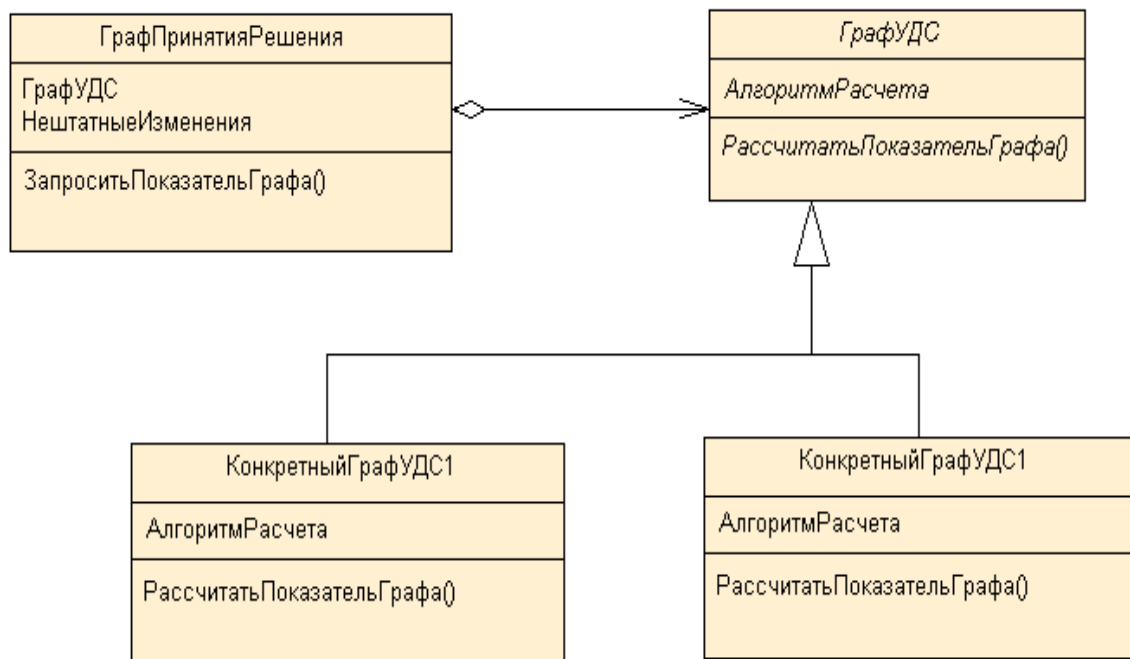


Рисунок 5. Диаграмма реализации паттерна «Состояние»

Объект, который может принимать различные состояния – *Граф_УДС*, входящий по принципу агрегации в класс *Граф_Принятия_Решения*. При этом в части множественной обработки данных состояние *Граф_УДС* характеризуется с помощью единственной переменной *FAlgString* типа *String*. Эта переменная может принимать строковое значение произвольной длины, удовлетворяющее грамматике:

<АлгИд>:<ДинБиблиотека>-<ТочкаВхода>||<СемантическоеОписание>

АлгИд – идентификатор алгоритма расчета показателя графа (целочисленная величина);

ДинБиблиотека – название динамической библиотеки, реализующей данный алгоритм расчета (строковое значение);

ТочкаВхода – название процедуры в динамической библиотеке, производящее расчет (строковое значение);

СемантическоеОписание – краткое естественно-языковое описание алгоритма расчета (строковое значение).

При этом *UML* структура реализации описанного паттерна имеет вид, изображенный на рисунке 5.

Объекты конкретные состояния графа *УДС* входят в качестве одного из атрибутов в объект *Граф_Принятия_Решения* и отличаются только *Алгоритмом_Расчета*. В зависимости от этого соответственно изменяется и поведение объекта при вызове функции *Рассчитать_Показатель_Графа*.

Паттерн «Адаптер» взаимодействия системы с внешними источниками данных

Данный паттерн применяется при разработке систем, в которых необходимо обеспечить взаимодействие несовместимых интерфейсов или создать единый устойчивый интерфейс для нескольких компонентов с разными интерфейсами. В контексте данной системы паттерн «Адаптер» применен при проектировании и реализации подсистемы загрузки данных.

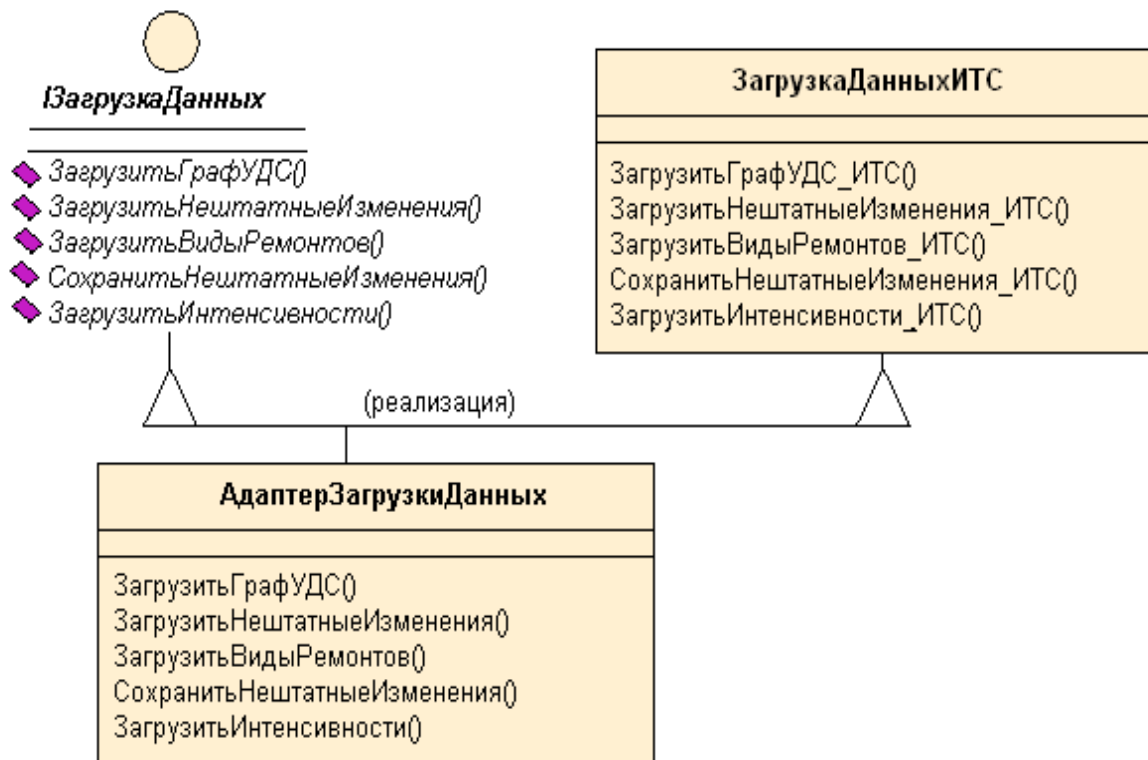


Рисунок 6. Диаграмма реализации паттерна «Адаптер»

Исходные данные для данной системы находятся в двух источниках данных:

- Геоинформационная база данных – данные о топологии улично-дорожной сети представлены в формате «ITSGIS» либо в виде реляционной модели базы данных.
- Данные об интенсивностях – находятся в реляционном представлении СУБД *InterBase/FireBird*.
- Данные о нештатных изменениях – хранятся в текстовом файле определенной структуры либо в реляционном представлении СУБД *InterBase/Firebird*.
- Данные о видах ремонтных работ – хранятся в текстовом файле определенной структуры.

Именно преобразованием всех описанных внешних данных к внутренним интерфейсам системы занимается подсистема загрузки данных, реализованная по принципу паттерна-адаптера.

Топология паттернов проектирования

Под топологией паттернов проектирования разрабатываемой системы «ITSGIS» подразумевается набор применяемых паттернов и их взаимное расположение на методологическом уровне решения проблемы.

Например, при разработке автоматизированной системы исследования транспортной нагрузки при возникновении нештатной ситуации необходимо сделать попытку посмотреть на все внутренние модели данных и процессы системы с единой точки зрения, обеспечивающей максимальную масштабируемость и простоту взаимодействия.

В рамках проектируемой системы управления транспортной инфраструктурой, объектом которой являются потоки и топология улично-дорожной сети, в целом необходимо смотреть на процесс разработки через призму паттернов «Адаптер» и «Информационный эксперт». Паттерн «Информационный эксперт» предоставляет новые функции по обработке данных. Паттерн «Состояние» применяются для расширения имеющейся функциональности и являются гибкой альтернативой порождению подклассов. Из данного утверждения логичным является вывод о том, что класс взаимодействия подсистем необходимо реализовать с использованием паттерна «Информационный эксперт» и паттерн «Состояние».

Паттерн «Адаптер», в свою очередь, преобразует интерфейс класса в некоторый другой интерфейс, ожидаемый клиентами. Обеспечивает совместную работу классов, которая была бы невозможна без данного паттерна из-за несовместимости интерфейсов. Логично использование данного паттерна при взаимодействии с геоданными и информационным хранилищем системы. Это

необходимо по причине того, что указанные хранилища должны быть готовы к использованию другими системами, работающими с ними, либо требовать минимальной разработки промежуточного слоя. Механизм работы с базой данных проектируется с использованием паттерна «DataSet» и «Адаптер». Паттерн «Адаптер» обеспечивает преобразование данных к интерфейсу, обрабатываемому системой. В рамках рассматриваемой системы он преобразует набор таблиц реляционной базы данных к внутреннему представлению моделируемых объектов.

В виде паттерна «Адаптер» реализуется подсистема загрузки данных о состоянии улично-дорожной сети, о ее нештатных изменениях, о функциональных возможностях по их устранению. Подсистема загрузки данных, в свою очередь, взаимодействует со всей системой в целом, используя паттерн «State» (состояние). Данный вид паттернов определяет семейство алгоритмов, инкапсулируя их и позволяя подставлять один вместо другого. Можно менять алгоритм независимо от клиента, который им пользуется. Подсистема проецирования исходных данных должна быть независима от клиента и допускать как изменение формата данных, так и алгоритма их преобразования. Паттерн «State» является ключевым при проектировании всей системы в целом. По данной методологии организуется взаимодействие с алгоритмами поиска решения, отображения графов принятия решения и других прикладных подсистем системы исследования транспортной нагрузки при возникновении нештатных ситуаций.

Паттерн поиска оптимального плана восстановления сети

В задачах поиска оптимального плана восстановления сети резонно рассматривать следующий уровень проектирования, более близкий к уровню реализации, т.к. именно он представляет наибольший интерес. Здесь рассматривается паттерн поведения «Информационный эксперт» (*Information Expert*). Паттерн «Информационный эксперт» представляет операцию, которую надо выполнить над элементами объекта. Позволяет определить новую операцию, не меняя классы элементов, к которым он применяется. При его применении совместно с паттерном «Стратегия» (*Strategy*) возникает особый симбиоз, обладающий повышенной гибкостью, позволяющий изменять «на лету» алгоритм работы управляющего модуля системы. Это должно позволить в случае необходимости без особых затруднений расширять и модифицировать обработку данных и изменять приоритеты вычислительных алгоритмов.

Реализация паттернов проектирования системы

Проектирование любой системы начинается с поиска уже найденных ранее решений или идей. Найденные подходы к решению некоторых классов оформляются в идее паттернов. Паттерны описывают, как можно реализовать иерархию классов и их взаимодействие с тем, чтобы масштабирование системы в условиях поставленной задачи было наименее затратным как с точки зрения трудозатрат, так и с точки зрения экономического эффекта. При разработке рассматриваемой системы использовался ряд стандартных паттернов проектирования, таких как паттерн «Адаптер», паттерн «Информационный эксперт», паттерн «Состояние».

Следует отметить, что перечисленные способы хранения внешних данных имеют место исключительно в контексте разработанной системы по причине того, что именно применение паттерна «Адаптер» делает требования к хранилищу гибкими и модифицируемыми. Например, хранение геоданных УДС может обеспечиваться средствами «ITSGIS» ArcInfo, InGEO, RuGIS при наличии соответствующих подсистем загрузки данных, реализующих интерфейс взаимодействия, ожидаемый системой обработки данных.

Заключение

Применение паттернов при проектировании и реализации информационных систем, решающих транспортные, сетевые и потоковые задачи, позволяют использовать разработанные решения в сопряжении с различными геоинформационными системами, обеспечивая агрегацию геоданных и дополняя (визуализируя) информационную картину, позволяя минимизировать неопределенность по данным.

Следует отметить, что при работе с таким гибким объектом, как транспортная сеть и при моделировании процессов, происходящих на ней, нельзя ограничиваться рассмотренными здесь паттернами. В каждой конкретной задаче необходимо проводить анализ структуры данных, алгоритмов расчета и возможных схем их взаимодействия для того, чтобы оптимально выбрать подходящие паттерны.

Список использованных источников

1. Воронина, Е.Е. Фотовидеофиксация как средство профилактики дорожно-транспортных происшествий / Е.Е. Воронина // Вестник НЦБЖД. – 2019. – № 2(40). – С. 73-79.
2. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влассидес // СПб.: Питер, 2001. – 368 с.

3. Минниханов, Р.Н. Применение данных ИТС систем для моделирования эффективной транспортной среды в Республике Татарстан / Р.Н. Минниханов, С.С. Степанова, В.А. Решетов, А.З. Махмутова // Безопасность дорожного движения. – 2021. – № 3. – С. 8-12.
4. Михеева, Т.И. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с.
5. Михеева, Т.И. Структурно-параметрический синтез интеллектуальных транспортных систем / Т.И. Михеева. – Самара : Самар. науч. центр РАН, 2008. – 380 с.
6. Михеева, Т.И. Системный анализ объектов транспортной инфраструктуры в геоинформационной среде / Т.И. Михеева // Программные продукты и системы. – 2018. – № 1. – С. 12-18.
7. Головнин, О.К. Паттерны поддержки принятия решений по дислокации технических средств организации дорожного движения / Т.И. Михеева, О.К. Головнин // Перспективные информационные технологии (ПИТ 2013) : труды Международной научно-технической конференции. – Самара : Издательство Самарского научного центра РАН, 2013. – С. 267–272.
8. Сапрыкин, О.Н. Построение архитектуры аналитического инструментария интеллектуальной транспортной системы на основе паттернов проектирования / Т.И. Михеева, О.Н. Сапрыкин, О.В. Сапрыкина // Вестник Самарского государственного технического университета. Серия: Технические науки. – 2010. – № 2 (27). – С. 27–35.
9. Михеев, С.В. Модели наследования в системе управления дорожным движением / Т.И. Михеева, С.В. Михеев // Информационные технологии. – 2001. – № 7. – С. 50–54.
10. Головнин, О.К. Универсальное программное средство для геоинформационных систем сенсорных инфокиосков / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Региональная научно-практическая конференция, посвященная 50-летию первого полета человека в космос : тезисы докладов. – Самара : Издательство Самарского государственного аэрокосмического университета, 2011. – С. 235–236. – ISBN 978-5-7883-0839-5.
11. Головнин, О.К. Применение паттерна «Рабочий поток» в геоинформационной составляющей интеллектуальной транспортной системы [Электронный ресурс] / О.К. Головнин // Математика. Компьютер. Образование : тезисы докладов двадцатой международной конференции. – Пущино, 2013. – Режим доступа : <http://www.mce.su/rus/archive/abstracts/mce20/sect101361/doc174037>.

12. Golovnin, O.K. Interactive content management in the system “Infomatriks” / T.I. Mikheeva, S.V. Mikheev, O.K. Golovnin, I.G. Bogdanova // Proceedings of the 14th international workshop on computer science and information technologies CSIT’2012, Ufa – Hamburg – Norwegian Fjords, September 20–26, 2012. – P. 285–289.
13. Смолев, А.М. Методы моделирования адресного плана в рамках геоинформационной системы / А.М. Смолев, Т.И. Михеева, А.В. Золотовицкий // IT & Transport / ИТ & Транспорт : Сборник научных статей. – Самара : ИнтелТранС, 2023. – Т. 24. – С. 4-13.
14. Михеева, Т.И. Информационная технология автоматической дислокации геообъектов транспортной инфраструктуры на улично-дорожной сети / Т.И. Михеева, А.В. Сидоров, О.К. Головнин // Перспективные информационные технологии : труды Международной научно-технической конференции, Самара, 04–06 декабря 2013 года / Самарский государственный аэрокосмический университет им. академика С.П. Королева. – Самара : Самарский научный центр РАН, 2013. – С. 236-241.
15. Михеева, Т.И. Инструментальная среда для проектирования объектов интеллектуальной транспортной системы / Т.И. Михеева // Вестник Самарского гос. техн. ун-та. Серия «Технические науки» №40. Самара: СамГТУ, – 2006. – С.96-103.

Mikheeva T.I., Smolev A.M.

DESIGN PATTERNS OF THE "ITSGIS" SYSTEM FOR ADDRESSING INTENSITY MEASUREMENT OPTIMIZATION PROBLEMS

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

This article describes typical tasks of measuring intensity in the ITSGIS geographic information system. These tasks are efficiently solved by interacting with objects that make up the city's street network. Street network objects are linked to the street geometries of the map. Information is given on design patterns that allow optimizing the intensity calculation process by reusing the solutions to the subtasks included in this calculation. The methods and tools that were used to implement the solutions within the ITSGIS system are disclosed.

Keywords: design, street network, intensity, geographic information system, information expert, state, adapter.

УДК 004.02

Смолев А.М., Михеева Т.И.

ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АДРЕСНОГО ПЛАНА ГЕОИНФОРМАЦИОННОЙ СИСТЕМЫ НА ПАТТЕРНОВОЙ ОСНОВЕ

*Самара, Самарский университет имени академика С.П. Королёва
ИнтелТранс*

Данная статья посвящена описанию паттернов проектирования, примененных при разработке программной системы обработки адресного плана территории и ее интеграции с геоинформационной системой общего назначения «ITSGIS». Наряду с общесистемными подробно охарактеризованы специализированные паттерны, применяемые в сфере геоинформатики и для разработки геоинформационных систем.

Ключевые слова: паттерн, проектирование, логическая модель данных, модульная архитектура, геоинформационная система, визуализация, адресный план, инструмент.

Введение

При создании любой достаточно сложной программной либо информационной системы ее разработчики неизбежно будут прибегать к применению паттернов проектирования как средству упрощения такой разработки [1]. Упрощение обеспечивается основным свойством паттернов проектирования: они позволяют неоднократно использовать одни и те же средства для решения разных по сущности, но схожих по своей структуре задач по построению архитектуры системы. За счет этого достигается упрощение взаимодействия различных разработчиков и ускорение разработки отдельных систем [2].

Для решения задачи построения системы управления адресным планом (АдП) применялся ряд методов, основанных на теории паттернов. Эти методы можно подразделить на два вида: общесистемные паттерны, применимые при решении задач в любой предметной области, и специализированные – используемые только в конкретной области. В нашем случае специализированные паттерны направлены на способствование нахождению решений задач в сфере геоинформатики [3].

Перечислим общесистемные паттерны, предназначенные для решения общих задач проектирования программной системы и использованные в син-

тезируемой системе управления АдП: проектирования интеллектуальной транспортной геоинформационной системы «ITSGIS»:

- «Встраиваемый модуль» (*Plug-in*);
- «Инверсия управления» (*Inversion of Control*);
- «Внедрение зависимостей» (*Dependency Injection*);
- «Перехватчик» (*Interceptor*);
- «Команда» (*Command*);
- «Издатель-подписчик» (*Publish-subscribe*);
- «Модель-Вид-Презентатор» (*MVP*) [4].

Теперь приведем список паттернов, примененных при реализации доступа к данным адресного плана реляционной СУБД и описанных в работе М. Фаулера [5]:

- «Модель предметной области» (*Domain Model*);
- «Объектно-реляционное отображение» (*Object-Relational Mapping*);
- «Шлюз» (*Gateway*);
- «Единица работы» (*Unit of Work*);
- «Объект доступа к данным» (*Data Access Object*);
- «Объект-запрос» (*Query Object*).
- «Объект передачи данных» (*Data Transfer Object*).

Многие из перечисленных паттернов уже применялись при интеграции других элементов в программную систему «ITSGIS» [6]. Благодаря применению паттерна проектирования «Встраиваемый модуль» удалось добиться повторного использования ряда решений, предложенных в этих элементах системы.

Среди специализированных паттернов, использованных при проектировании геоинформационной системы адресного плана, можно выделить паттерны атрибутной типизации, визуализации пространственных данных, структурирования данных и инструментальные. Оставшаяся часть статьи посвящена подробному описанию этих паттернов.

Паттерны атрибутной типизации

Семантический атрибут. Паттерн «Семантический атрибут» задает смысловую (сущностную) характеристику пространственного объекта.

Например, «год постройки здания» $year_{\tilde{B}} \in N = \{1, 2, \dots, CY\}$ является семантическим атрибутом объектов класса $\tilde{B} \in AP$. Этот атрибут применяется при решении задачи выбора участков застроенной территории под снос либо комплексную реновацию.

Пространственный атрибут. В свою очередь, паттерн «Пространственный атрибут» задает расположение объекта относительно других либо частей одного и того же объекта. В архитектуре ГИС данный паттерн делится на три подвида: *объект-точка*, *объект-линия* и *объект-полигон*. Они отличаются размерностью атрибутивных данных. Пространственное расположение точки *coord* задается при применении паттерна «объект-точка» как $coord \in Z^{1 \times 2}$, где $Z^{1 \times 2}$ – векторное пространство размерностью 2. Для паттерна «объект-линия» расположение задается уже как $coord \in Z^{2 \times k}$, т.е. *coord* представляет собой матрицу с 2 столбцами и *k* строками. Наконец, для «объекта-полигона» $coord \in Z^{2 \times k \times m}$, т.е. *coord* состоит из *m* матриц размером $2 \times k$. Первая матрица задает внешний контур полигона, причем первая ее строка равна последней; матрицы 2..*m* задают границы отверстий полигона.

Адресный атрибут. Паттерн «Адресный атрибут» задает расположение объекта относительно существующего адресного объекта для ориентирования человеком в условиях отсутствия графической карты. В объектно-ориентированной ГИС этот атрибут задается паттерном как ссылка на адресную запись $A_O AP$ адресного объекта *O*, расположенную в отдельной модели. Например, камера фотовидеофиксации, регулирующая движение на перекрестке ул. *Аэродромная* × ул. *Авроры*, получит адресный атрибут со ссылкой именно на этот перекресток. В некоторых случаях атрибут может быть двойным – светофор, расположенный на пересечении первой и второй улицы, но регулирующий движение по первой, может получить главную привязку (первая улица) и второстепенную (вторая).

Паттерны визуализации

Атрибут визуализации. Паттерн «Атрибут визуализации» описывает способ базового представления подписи объекта адресного плана на электронной карте. Если данный атрибут указан, то отображение подписи происходит в соответствии с описанным методом для объектов класса $Algorithm \Rightarrow A_CustomRenderer$, если атрибут не указан, то в соответствии с общепринятым представлением $Algorithm \Rightarrow A_BaseRenderer$ (рисунк 1).

Например, для задач, требующих визуализации подписи улицы, применяется $A_CustomRenderer \Rightarrow A_StreetCaptionRenderer$, устанавливающий положение подписи с применением специализированного алгоритма расстановки, основываясь на трассировке улиц и масштабе карты. Этот алгоритм должен расставлять подписи улиц прямо вдоль их направления и избегая пересечений подписей. Алгоритмы $A_CustomRenderer \Rightarrow A_TerritoryCaptionRenderer$ и $A_CustomRenderer \Rightarrow A_HouseCaptionRenderer$ устанавливают подпись тер-

ритории (здания) в центре либо во внутренней точке объекта, если текущий масштаб карты находится в диапазоне отображения соответствующего слоя.

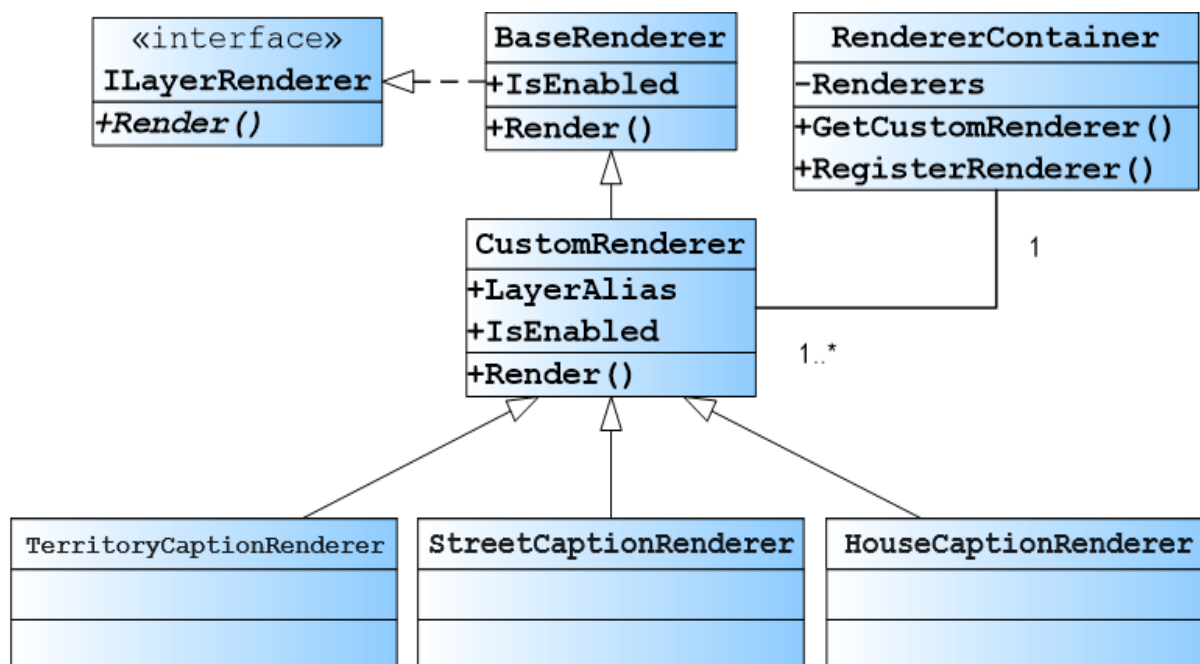


Рисунок 1. Диаграмма реализации паттерна «Атрибут визуализации»

Сбалансированная цветовая схема. Паттерн «Сбалансированная цветовая схема» используется для отображения и идентификации типов зданий на карте. Чем реже встречается некоторый тип зданий, тем более ярким и заметным цветом здания такого типа помечаются на карте. Например, медицинские учреждения выделяются красным цветом. Здания часто встречающихся типов (например, жилые) помечаются нейтральными цветами – бежевым либо близким к белому. Использование данного паттерна улучшает читаемость карты и заостряет внимание пользователя на действительно важных объектах.

Диапазон масштаба отображения. Паттерн «Диапазон масштаба отображения» определяет минимальный $SC_{\min}$ и максимальный $SC_{\max}$ масштабы карты, на котором будут отображаться:

- сами объекты адресации (O);
- подписи соответствующих объектов адресации (A_o).

Под минимальным масштабом карты подразумевается самый мелкий. Для объектов одного типа должно выполняться условие: $SC_{\min}(O) \leq SC_{\min}(A_o)$, т.е. минимальный масштаб отображения объекта не должен превышать минимальный масштаб его подписи.

Шаблон именованного адресного объекта. Данный паттерн применяется при нанесении подписи территории или улицы на интерактивную карту. Шаблон может содержать как обычный текст, так и специальные теги, обо-

значающие необходимость подстановки в это место значения некоторого атрибута адресного объекта.

В системе должно быть реализована подстановка трех видов атрибутов, обозначаемых тегами `[name]`, `[type]` и `[abbr]`. Тег `[name]` обозначает название объекта, `[type]` – его тип (напр. городской округ) и `[abbr]` – сокращение этого типа (г.о.) Дополнительно каждый тег – допустим, `[type]` – имеет три разновидности, обозначающие регистр символов вставляемого значения: `[type]` – текст вставляется в виде, записанном в БД, `[Type]` – первая буква заглавная, остальные – строчные, `[TYPE]` – все символы заглавные.

Пример подстановки значений для улицы Гагарина: `[Type] [name]` – «Улица Гагарина», `[abbr] [name]` – «ул. Гагарина». Различные способы наименования могут применяться в подходящих для этого сценариях – полный тип может использоваться для заполнения отчетов, сокращенный – для визуализации карты. Кроме того, применяемый шаблон зависит от самого названия адресного объекта.

Дерево сворачиваемых элементов. Этот паттерн используется для отображения списка территорий, составляющих полную структуру АдП ГИС. Такой способ отображения позволяет достаточно быстрым и наиболее удобным образом находить нужную территорию при ее выборе (рисунок 2). Этот паттерн может сочетаться с еще одним – отображением текущей цепочки раскрытых территорий в верхней части дерева.

Выбор уличного адресного объекта. Паттерн используется для присваивания адресному атрибуту произвольного геообъекта некоторого адреса улицы либо перекрестка. Этот паттерн смешанный, поскольку несет в себе качества как паттерна визуализации, так и паттерна структурирования данных.

Визуальная часть паттерна (рисунок 3) состоит в пользовательском элементе интерфейса, реализующего данную стратегию. Она реализована в виде графы с двумя рядами, содержащими флажок и выпадающий список со списком близлежащих улиц. Флажок вверху определяет, необходимо ли привязывать объект к адресу. Флажок внизу определяет тип адресного объекта, к которому необходимо привязать модифицируемый геообъект – улицу либо перекресток. При выборе улицы активен один выпадающий список, при выборе перекрестка – два.

Поля с выпадающими списками содержат набор улиц, ближайших к геообъекту. Улицы сортируются в порядке удаления. Если улица еще не была присвоена объекту, поля по умолчанию заполняются самыми близкими к объекту улицами.

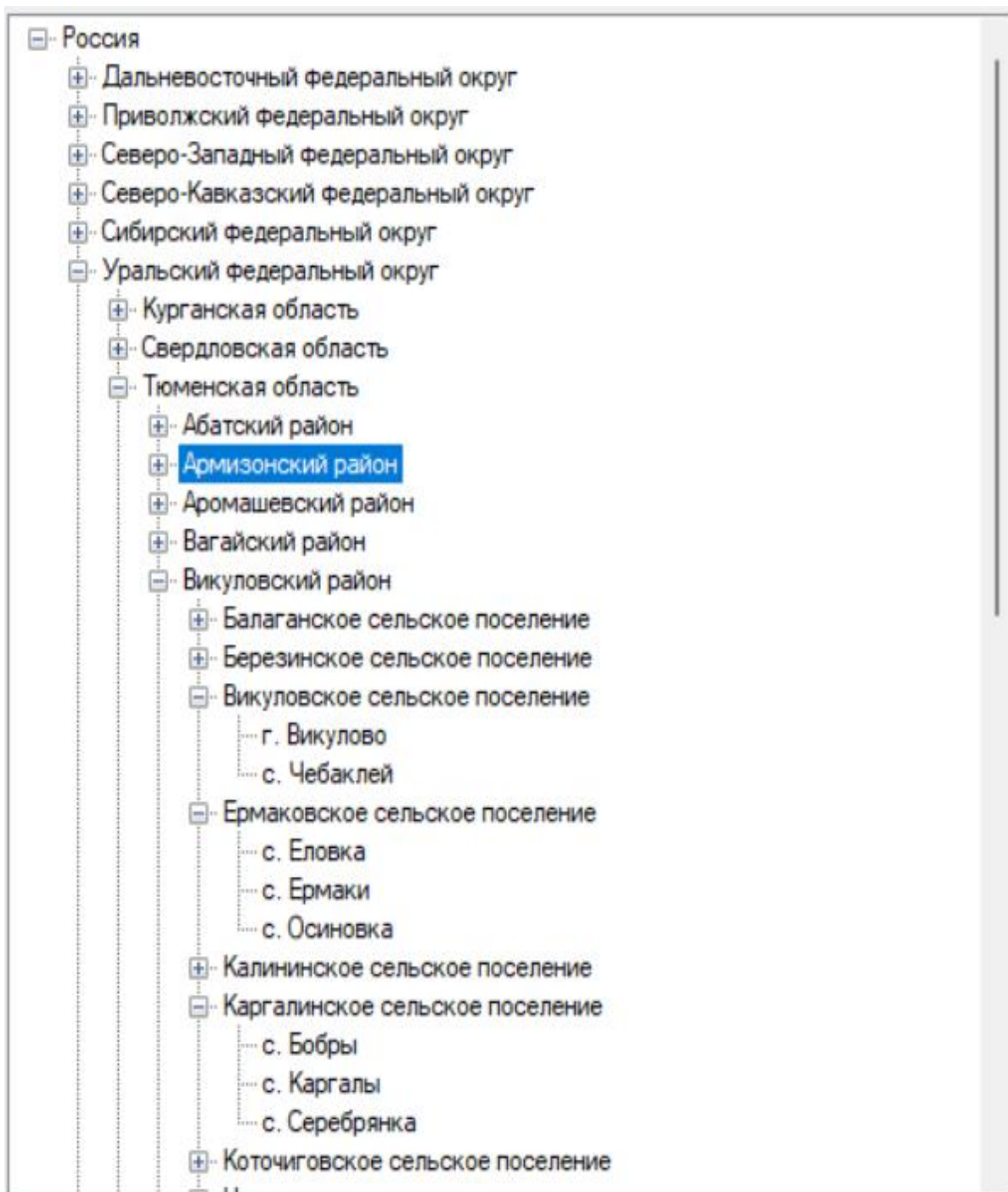


Рисунок 2. Паттерн «Дерево сворачиваемых элементов»

Улица	☒	г. Сызрань, ул. Комсомольская	▼
Пересеч.	☒	г. Сызрань, ул. Володарского	▼

Рисунок 3. Паттерн «Выбор уличного адресного объекта»

Структурная часть паттерна заключается в способе хранения этой атрибутивной информации в базе данных как двух необязательных записей об улицах в таблицах соответствующих типов геообъектов. Для сохранения быст-

родействия системы доступа к данным отдельная таблица со списком привязок улиц не создается.

Зависимое от точности данных отображение. Для визуализации пространственных и адресных данных на разном масштабе в случаях, когда использования только паттерна «Диапазон масштаба отображения» недостаточно, применим паттерн «Зависимое от точности данных отображение». Этот паттерн задает различные способы отображения одного и того же объекта в зависимости от масштаба. При его укрупнении представление объекта становится более точным благодаря тому, что этот объект занимает больше места на экране. Например, на мелких масштабах дорога может отображаться простой линией, а на крупных – полигоном с шириной, соответствующей ширине существующей или проектируемой дороги.

Паттерны структурирования данных

Генератор атрибутов. Паттерн «Генератор атрибутов» предназначен для формирования набора атрибутов, требуемых какому-либо применяемому методу в среде синтезируемой системы.

Например, существуют различные способы задания адреса территории A_T , используемые в нормативно-правовых или отраслевых методических документах: начиная с корневой $\langle \{A'_{T_i}\}_{i=1}^{n-1}, A'_{T_n} \rangle$, с самой вложенной

$\langle A'_{T_n}, \{A'_{T_i}\}_{i=n-1}^1 \rangle$, с полным именованном типа (Самарская область), с со-

кращением типа (Самарская обл.) и т.п. Для формирования полного адреса территории, описывающего ее расположение в соответствии с применяемым способом, используется реализация паттерна «Генератор атрибутов». Для каждого способа описания состава формируется набор типов территорий $Type^T$ и их сокращений $Tabbr^T$. Такой способ позволяет единообразно использовать в системе данные об адресе территории, полученные и обработанные различными способами.

Атрибутный фильтр. Паттерн «Атрибутный фильтр» предназначен для отбора атрибутов, требуемых для выполнения какой-либо операции. Для реализации паттерна описываются ограничения на значения атрибута: «входит в набор», «содержит», «содержит без учета регистра», «равно», «больше», «меньше», «не» (отрицание). На один атрибут может быть наложена комбинация ограничений.

Например, для формирования списка зданий, в которых в соответствии с нормативными актами должны располагаться лифты, требуется получить информацию о том, этажность каких зданий превышает 5. Для этого формиру-

ется атрибутный фильтр, позволяющий получить информацию о существующих зданиях и их этажности.

Другим примером может выступить задача определения перекрестков с установленными камерами фотовидеофиксации. Метод поиска перекрестков *Algorithm* $\Rightarrow$ *A_MonitoredIntersections* запрашивает атрибуты камер, описывающие их пространственную и адресную привязку.

Загрузка по требованию. Подсистема управления адресным планом обеспечивает работу других модулей ГИС, что приводит к многослойной сущности модели данных. Например, при просмотре атрибутов объектов класса «Дорожный_знак» также могут быть рассмотрены атрибуты объекта класса «Улица», содержащего сведения об адресе расположения дорожного знака на электронной карте. Паттерн «Загрузка по требованию» упрощает работу с атрибутами объекта класса «Дорожный_знак», так как он исключает необходимость в вызове дополнительных запросов на получение данных о его дислокации. Для этого паттерном описываются способы отказа от загрузки сопряженных объектов, когда в этом нет необходимости. Существует несколько способов реализации паттерна «Загрузка по требованию» [7].

- «Маркированный атрибут» использует маркер, помечающий состояние значения атрибута (рисунок 4). Информация об атрибутах загружается из БД только в том случае, если клиент ГИС не получил ее ранее.



Рисунок 4. Паттерн «Загрузка по требованию»

- «Виртуальный объект» представляет собой объект-заглушку с таким же интерфейсом, как и у настоящего объекта. При создании его атрибуты остаются пустыми, а заполняются они при первом их запросе.

- «Объект-контейнер» отличается от «виртуального объекта» тем, что для загрузки информации используется не сам атрибут, а отдельный метод, что требует проверки значения исходного атрибута для определения необходимости загрузки данных из БД.
- «Объект-призрак» загружает данные для всех атрибутов при первом обращении к любому атрибуту объекта. Используется в случаях, когда загрузка одного атрибута практически гарантирует необходимость загрузки всех атрибутов.

Общая коллекция. Синтезируемая система управления в процессе функционирования вызывает методы получения данных, требующихся различным подсистемам. Иногда разные подсистемы требуют для своей работы одну и ту же информацию. Чтобы избежать повторные вызовы методов получения уже имеющихся данных от обособленных подсистем, применяется паттерн «Общая коллекция». Эта коллекция содержит информацию обо всех объектах определенного типа, которые ранее запрашивались любым из модулей системы. Если впоследствии те же данные потребуются другому модулю, система возьмет их именно из этой коллекции. В случае изменения и удаления каких-то данных эти изменения также отражаются и в коллекциях. Данный паттерн может применяться в сочетании с другим – «Таймером» – в случае, если некоторые объекты не вызывались в течение продолжительного времени, и при этом нам требуется освободить память ОЗУ. Тогда такие объекты удаляются из коллекции и при последующем обращении они снова будут загружаться из БД.

Инструментальные паттерны

Пул потоков. Для выполнения множества длительных запросов применяется паттерн «Пул потоков». Особенно он полезен в случаях, когда нам требуется запрашивать множество данных из СУБД, а сами запросы могут быть параллелизованы. Классическим примером применения паттерна является загрузка цифровой интерактивной карты. В данном случае при загрузке видимых участков карты главный поток приложения создает поток загрузки для каждого такого участка. Общее число действующих потоков в пуле может быть значительным (от 15), однако каждый поток входит в режим ожидания после вызова метода удаленной загрузки данных, освобождая ресурсы ЭВМ для выполнения других потоков, и возобновляет работу только после ответа сервера. Пример выполнения запроса однотипных данных с удаленного сервера и с применением пула потоков представлен на рисунке 5.

Отмена выполнения. В случаях, когда в определенный момент действие над объектом не может быть произведено, применяется паттерн «Отмена выполнения».

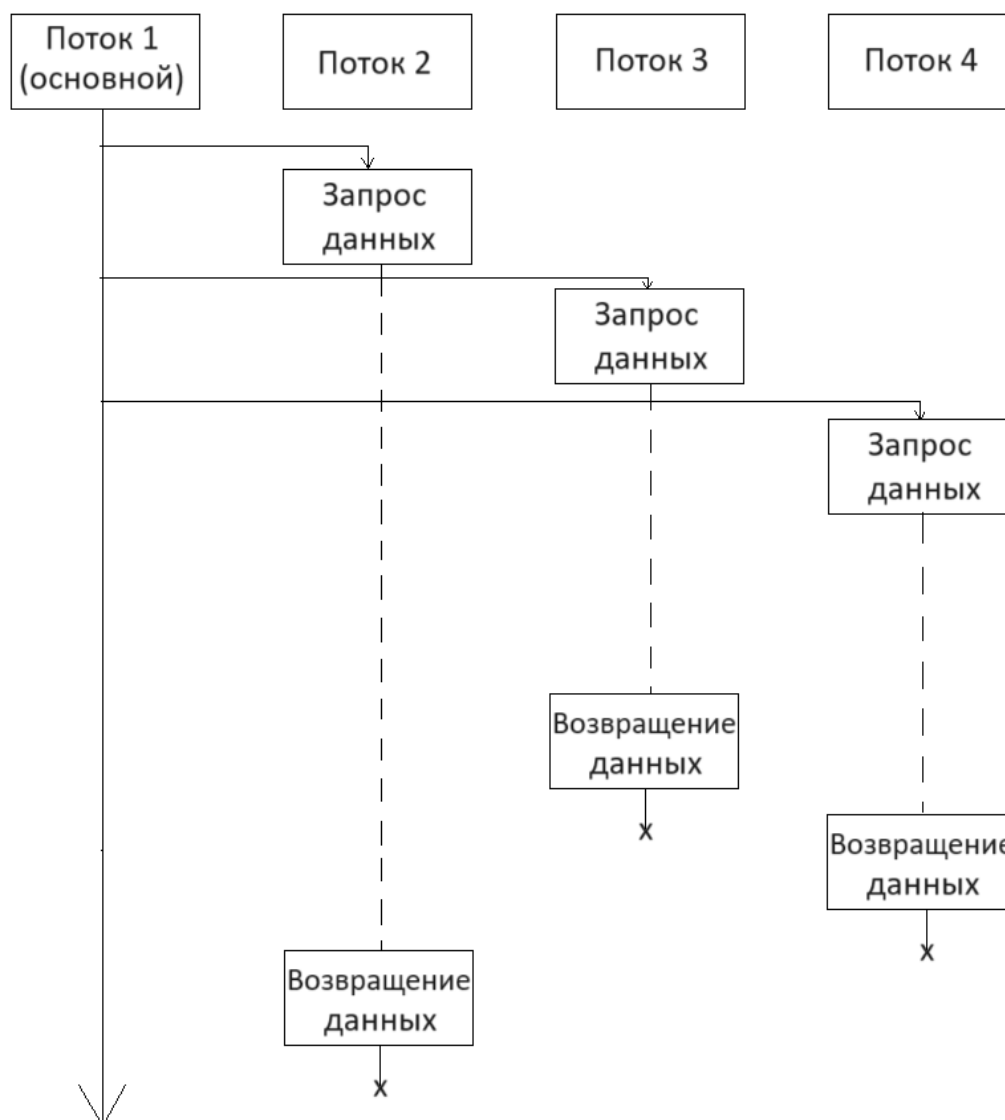


Рисунок 5. Диаграмма выполнения многопоточного запроса данных

Он представляет собой создание дополнительного атрибута объекта, который прямо либо косвенно содержит информацию о том, может ли быть выполнено то или иное действие. Самый простой случай – пустое значение атрибута, необходимого для выполнения действия: например, пустое значение атрибута «Шаблон именования типа» приводит к невозможности выполнения алгоритма $Algorithm \Rightarrow A_BuildNameFromTemplate$, заданного соответствующим паттерном. Также доступность алгоритма может зависеть от конкретного значения атрибута – например, если атрибуту «Тип территории» присвоено любое значение, отличное от «Населенный пункт», такой территории нельзя установить тип населенного пункта.

Интервал времени. Паттерн используется при задании промежутка от одного момента времени до другого. Момент времени может быть точным (заданы и дата, и время) либо относительным (задано только время, либо заданы время и день недели). Для реализации паттерна используются два объекта. В зависимости от типов объектов существует две разновидности реализации:

- «начало-конец», когда используются два объекта типа «момент времени» T_1 и T_2 , задающих начальную и конечную границу промежутка;
- «начало-период»: используется объект «момент времени» T для задания начала отрезка и «продолжительность» D для определения величины этого отрезка – конец промежутка находится как $T + D$ (или D после T).

Список использованных источников

1. Мордвинов, В.А. Использование паттернов для проектирования информационных систем / В.А. Мордвинов // ИТНОУ: Информационные технологии в науке, образовании и управлении. – 2018. – № 1(5). – С. 79–85.
2. Хузятов, Ш.Ш. Проектирование автоматизированных систем управления на основе методики паттерна / Ш.Ш. Хузятов, Р.А. Валиев // Научно-технический вестник Поволжья. – 2015. – № 2. – С. 215–218.
3. Михеева, Т.И. Структурно-параметрический синтез распределенной геоинформационной системы решения задач сетцентрического управления транспортными процессами на основе паттернов / Т.И. Михеева, О.К. Головнин // Системная инженерия и информационные технологии. – 2019. – Т. 1, № 1(1). – С. 22–31.
4. Fowler, M. Patterns of Enterprise Application Architecture / M. Fowler, D. Rice, M. Foemmel [и др.]. – Addison Wesley, 2002. – 560 с.
5. Турушев, Т.К. Архитектурные паттерны проектирования системы – MVC, MVP, MVVM / Т.К. Турушев // Инновационный потенциал развития науки в современном мире: достижения и инновации : Сборник научных статей по материалам VI Международной научно-практической конференции, Уфа, 22 октября 2021 года. – Уфа: Общество с ограниченной ответственностью «Научно-издательский центр «Вестник науки», 2021. – С. 16–21.
6. Михеева, Т.И. Паттерны проектирования сложноорганизованных систем / Т.И. Михеева, С.В. Михеев, О.К. Головнин, О.Н. Сапрыкин. – Самара : ИнтелТранс, 2015. – 216 с.

7. Головнин, О.К. Организация загрузки данных на основе паттернов в интеллектуальной транспортной системе / Т.И. Михеева, О.К. Головнин, С.В. Михеев // Перспективные информационные технологии в научных исследованиях, проектировании и обучении (ПИТ-2012) : труды научн.-техн. конф. с международным участием и элементами научной школы для молодежи, посвященной 40-летию кафедры информационных систем и технологий СГАУ. – Самара : Изд-во СНЦ РАН, 2012. – С. 224–228.
8. Михеева, Т.И. Паттерны поддержки принятия решений по дислокации технических средств организации дорожного движения / Т.И. Михеева, О.К. Головнин // Перспективные информационные технологии (ПИТ-2013) // Труды межд. научно-техн. конф. – Самара: Изд-во Самарск. науч. центра РАН, 2013. – С.267-273.
9. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с. – ISBN 978-5-9906857-4-1.
10. Mikheeva, T. Recognition of Urban Transport Infrastructure Objects Via Hyperspectral Images / O. Saprykin, A. Fedoseev, T. Mikheeva // In Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS-2016). Rome, Italy: SCITEPRESS, 2016. – P. 203-208.
11. Михеева, Т.И. Интеллектуальная транспортная геоинформационная система ITSGIS / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Современные проблемы безопасности жизнедеятельности: интеллектуальные транспортные системы : материалы IV Международной научно-практической конференции (Казань, 25–26 февраля 2016 г.). – Казань : ГБУ «Научный центр безопасности жизнедеятельности», 2016. – С. 362–368. – ISBN 978-5-85247-837-5.

Smolev A.M., Mikheeva T.I.

**PATTERN-BASED DESIGN OF THE ADDRESS PLAN
MODULE WITHIN A GEOGRAPHIC INFORMATION SYSTEM**

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

This article is devoted to the description of design patterns used in the software system for processing a territorial address plan as well as its integration with the general-purpose geoinformation system, "ITSGIS". Along with general system patterns, specialized patterns that are used in the field of geoinformatics and for geographic information system development are described in detail.

Keywords: pattern, design, logical data model, modular architecture, geographic information system, visualization, address plan, tool.

УДК 004.02

Смолев А.М., Михеева Т.И.

РЕАЛИЗАЦИЯ ПРОГРАММНОГО ИНТЕРФЕЙСА «ITSGIS» ДЛЯ ПРИМЕНЕНИЯ НА ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЕ

*Самара, Самарский университет имени академика С.П. Королёва
ИнтелТранс*

В данной статье приводится описание реализации программного интерфейса (API) геоинформационной системы «ITSGIS», предназначенного для загрузки информации об объектах различных видов на электронной карте. Обосновывается необходимость реализации данного программного интерфейса. Указывается, что для обеспечения взаимодействия компонентов «ITSGIS» с вновь разрабатываемыми модулями требуется устранить или снизить зависимость от конкретных инструментов разработки ПО. Раскрываются методы и средства, которые были применены при интеграции программного интерфейса в геоинформационную систему.

Ключевые слова: программный интерфейс, реализация, транспортная инфраструктура, модульная архитектура, компьютерная сеть, геоинформационная система, интеграция.

Введение

Многообразие технических и программных средств, применяемых при решении типичных для транспортной сферы задач управления, таких как мониторинг дорожного движения и проектирование автомобильных дорог, объясняется их многокомпонентностью и повышенной сложностью. Так, при управлении дорожным движением могут использоваться как дорожные технические средства – знаки, светофоры, освещение, шлагбаумы, камеры фотофиксации и т.д., так и программные – системы мониторинга и прогнозирования ситуации, макро- и микро моделирования дорожного движения, геоинформационные системы (ГИС) и пр. Соответственно, крайне высокую

значимость в управлении транспортным хозяйством имеют средства коммуникации, позволяющие программным и техническим средствам взаимодействовать друг с другом [1].

Одной из систем, предназначенных для управления дорожным движением, является интеллектуальная транспортная геоинформационная система «ITSGIS» (рисунок 1).

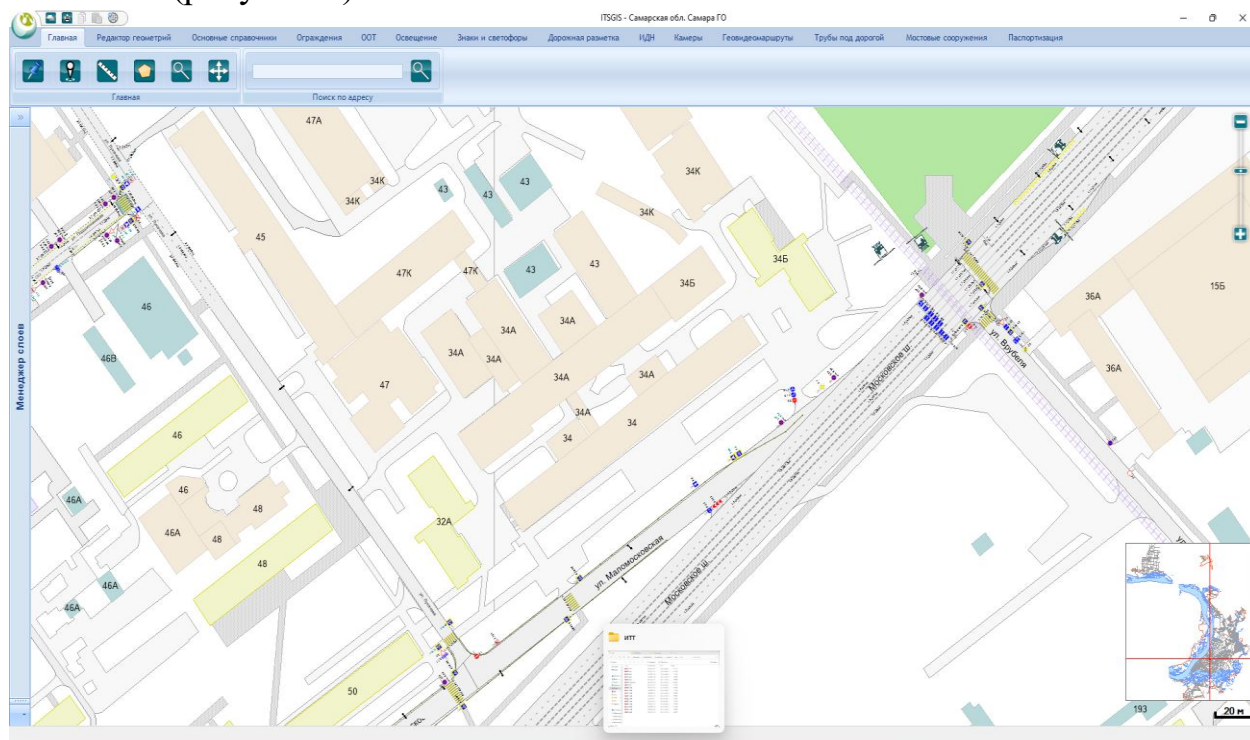


Рисунок 1. Графический интерфейс системы «ITSGIS»

Эта система обладает комплексной внутренней структурой и сочетает в себе компоненты, характерные для ГИС общего назначения, с инструментами, предназначенными для моделирования и проектирования транспортной инфраструктуры и позволяющими осуществлять организацию дорожного движения (ОДД) на объектах произвольного масштаба, начиная от окрестности некоторого здания и заканчивая целыми регионами страны [2]. В состав «ITSGIS» входит база данных (БД), которая и содержит информацию о транспортной инфраструктуре — технических средствах ОДД [3]. Помимо пространственной информации об этих объектах, в БД содержится подробная техническая характеристика по каждому из них, что позволяет пользователю однозначно идентифицировать их, а также получить необходимую для проведения моделирования транспортного процесса и прогнозирования ситуации информацию.

Система «ITSGIS» обладает трехзвенной архитектурой, в которую входят сервер БД, сервер приложений и клиентская часть. Эти компоненты связываются между собой посредством TCP-соединения. В свою очередь, клиент и сервер приложений обладают модульной архитектурой, позволяющей ди-

намически встраивать в систему специализированные плагины для обеспечения работы с геообъектами отдельного типа либо же дополнения системы функциями мониторинга, либо расчетного моделирования. При этом ядро приложения, отвечающее за реализацию функций и средств ГИС, обладает монолитной архитектурой. Особенностью реализации системы является то, что плагины, встраиваемые в нее, могут реализовываться только с помощью определенного стека технологий, который включает в себя следующие инструменты [4]:

- платформа разработки программного обеспечения (ПО) – *.Net Framework* версии 4.0;
- пространственное расширение платформы разработки – *NetTopologySuite* версии 1.13.1;
- система управления базами данных (СУБД) – *PostgreSQL* 9.1;
- пространственное расширение СУБД – *PostGIS* 1.5;
- контейнер инверсии зависимостей – *Unity Container*.

Хотя использование данных средств в большинстве сценариев применения «ITSGIS» является достаточным, в связи с их постепенным устареванием могут возникать и исключения. Кроме того, транспортные системы, разрабатываемые на данный момент, могут использовать и другие технологии, более типичные для разработки Интернет-приложений, такие как язык программирования Python либо программная платформа *Node.js*, а также базироваться на другой архитектуре, например, состоять из микросервисов. В связи с этим актуальной является задача разработки программного интерфейса (API) «ITSGIS» для упрощения взаимодействия с разнообразными приложениями, а также для создания возможности реализации плагинов «ITSGIS» на более новом либо альтернативном стеке технологий.

Программный интерфейс

Прикладным программным интерфейсом является стандарт или спецификация, которая описывает способы взаимодействия двух программных либо вычислительных систем между собой. Система, предоставляющая доступ к себе по некоторому API, должна реализовывать соответствующий интерфейс стандарт. Основным назначением API является повышение удобства разработчиков ПО при построении связей между уже существующей и новой, самим разрабатываемой программами. Этот фактор особенно важен в случаях, когда программисту не известны конкретные алгоритмы работы чужой системы, и единственным, исключая API и официальную документацию, способом получить информацию о системе является тестирование методом «черного ящика». Корректно документированный программный интерфейс

позволяет определить рамки, в которых могут действовать сторонние разработчики ПО.

На сегодняшний день существует три основные категории протоколов *API*, используемых при взаимодействии ПО по сети Интернет: *SOAP*, *RPC* и *REST* [5].

1. *RPC (remote procedure call)* – удаленный вызов процедур. Используется для программного вызова функций или методов объектов программы, расположенной на удаленном компьютере. Данный вид *API* позволяет передавать параметры функции и по прошествии времени получить результат выполнения. Представляет собой один из способов межпроцессного взаимодействия.

2. *SOAP (simple object access protocol)* – объектно-ориентированный протокол доступа к информации, предоставляемый веб-приложениями. Основной отличительной чертой *SOAP* является применение языка разметки *XML* для передачи данных между узлами сети. *SOAP* поддерживает широкий спектр коммуникационных протоколов, используемых в Интернете, таких как *HTTP*, *SMTP* и *TCP*. *SOAP* также расширяем и не зависит от стиля, что позволяет разработчикам писать программные интерфейсы различными способами и легко добавлять функции.

3. *REST (Representational State Transfer)* – протокол, базирующийся на использовании стандартных запросов *HTTP* для обработки информации и разделяющий внешний и внутренний интерфейсы приложения [5]. Операции по запросу данных должны осуществляться только на основе *URI*, т.е. адреса, уникально идентифицирующего ресурс. За счет этого обеспечивается возможность использования *API* приложения любыми программами, а не только изначально заточенными под этот интерфейс. Вне запросов данные или статус выполнения не сохраняются. В качестве типа передаваемых документов может использоваться любая (*XML*, *JSON* или *HTML*).

Способом построения системы, при котором программные интерфейсы играют главенствующую роль в связывании отдельных ее компонентов и их координации, является микросервисная архитектура. В ней *API* определяет способы, которыми один сервис может взаимодействовать с другим, включая запросы данных, выполнение операций и обмен информацией. Поскольку благодаря *API* обеспечивается стандартизация взаимодействия между сервисами, процесс разработки, тестирования и масштабирования системы с микросервисной архитектурой становится более простым и контролируемым [6]. Характерным элементом современных сложных программных комплексов является связывание ее частей посредством сети Интернет, за счет чего эти комплексы часто естественным образом приобретают сложную многокомпо-

нентную структуру, проще всего реализуемую в виде мелких служб – микросервисов.

Транспортная инфраструктура является одной из сфер применения микросервисной архитектуры при построении сложных программных систем. Так, в Москве на данной основе было предложено создание интеллектуальной системы управления городской рельсовой транспортной системой, способной обрабатывать информацию со множества датчиков, анализируемую с помощью алгоритмов искусственного интеллекта и применением технологии *Big Data*. Эта система должна строиться с использованием полносвязной сети *MPLS*, обеспечивающей коммутацию по меткам, а к подключаемым устройствам относятся камеры, датчики и системы контроля, устанавливаемые на трамваях, вагонах метро, станциях, участках путей и т.д. Как на внешнем уровне, т.е. между датчиками и сервером больших данных, так и на внутреннем (между микросервисами и клиентами) для взаимодействия устройств используются *API* различного рода (рисунок 2) [7].

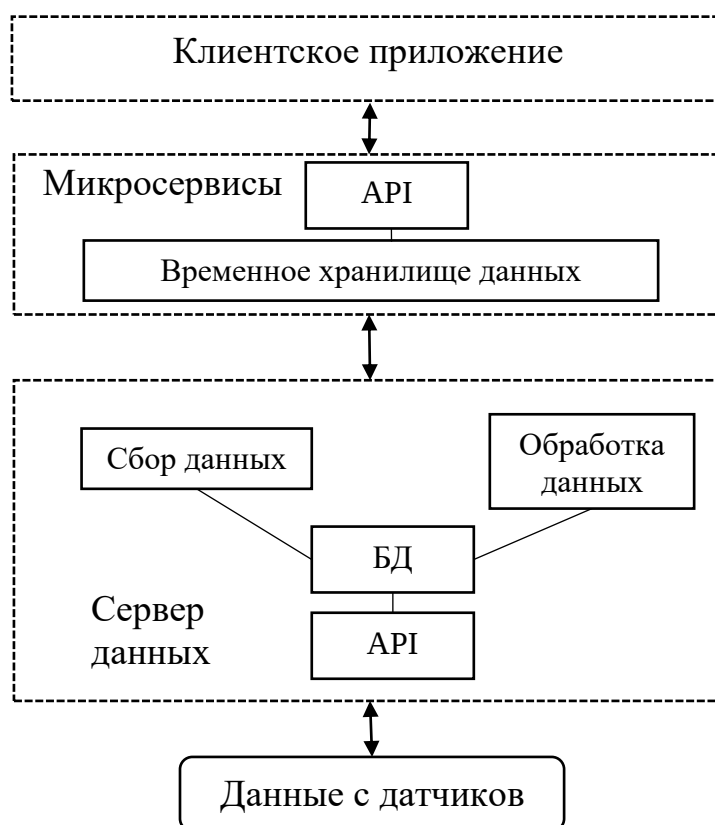


Рисунок 2. Архитектура системы управления рельсовым транспортом Москвы

Также микросервисная архитектура с взаимодействием посредством программных интерфейсов используется и при проектировании интеллекту-

альных транспортных систем, что объясняется гибкостью использования технологий и повышенной надежностью ее компонентов [8].

Требования к реализуемому API

Основным назначением программного интерфейса системы «ITSGIS» является предоставление возможность получения пространственных и атрибутивных данных, содержащихся в БД системы, посредством отвечающих требованиям *REST* запросов *HTTP* по локальной сети либо по сети Интернет.

Функциональные требования к API системы:

- предоставление данных в формате *JSON* подробной информации об объектах следующих типов:
 - ✓ адресный план,
 - ✓ подписи объектов,
 - ✓ проезжая часть автодорог,
 - ✓ дорожные знаки,
 - ✓ светофоры,
 - ✓ дорожная разметка,
 - ✓ опоры освещения,
 - ✓ ограждения,
 - ✓ остановки,
 - ✓ камеры фотовидеофиксации;
- обеспечение фильтрации объектов по координатам в задаваемых пользователем границах;
- поддержка преобразования координат, задаваемых в глобальных, локальных и местных системах координат, включая *WGS84*, проекцию Гаусса-Крюгера, СК-42, ГСК-2011.

Технические требования:

- обеспечение совместной работы с сервером БД «ITSGIS» на одном вычислительном устройстве;
- взаимодействие с СУБД *PostgreSQL* и географическим расширением *PostGIS*;
- поддержка подключения к системе «ITSGIS» произвольных компонентов, потенциально расширяющих транспортные и геоинформационные возможности системы;
- доступность для устройств с любым аппаратным и программным обеспечением, включая стационарные компьютеры с ОС *Windows*, *Linux*, *FreeBSD*, мобильные устройства, умные датчики и т.п.;
- функционирование как в локальной сети, так и сети Интернет;

- максимизация работоспособности приложения (период простоя, *downtime* не более 0,1%).

Реализация API «ITSGIS»

Для разработки программного интерфейса были выбраны следующие инструменты:

- платформа разработки – *.Net Core 3.1*, поддерживающая, с одной стороны, новейшие среды разработки, а с другой – обеспечивающая взаимодействие с сервером приложений ИТСГИС и СУБД *PostgreSQL 9.1*;
- интерфейс взаимодействия с БД – *Npgsql 6* с пространственным расширением *Npgsql.NetTopologySuite*, выбранные по тем же соображениям;
- расширение, поддерживающее работу с системами координат – *ProjNET4GeoAPI*;
- платформа разработки API – *ASP.Net Core* с поддержкой *MVC*;
- библиотека для поддержки работы с *Json* – *NewtonsoftJson*.

Разработанное приложение состоит из контроллеров *MVC*, принимающих и обрабатывающих *GET*-запросы со стороны произвольных приложений. Эти запросы должны содержать один основной и 4–6 дополнительных параметров. Алгоритм работы контроллеров API представлен на рисунке 3.

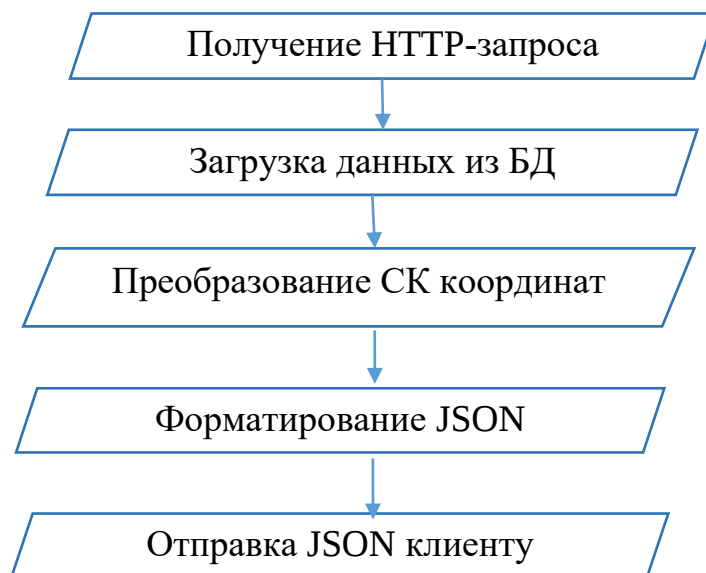


Рисунок 3. Схема обработки данных контроллером API

Основной параметр определяет тип выгружаемых объектов. Среди типов представлены как чисто геометрические объекты (подписи, здания, геометрии проезжих частей и т.д.), так и геообъекты, обладающие атрибутивной

информацией (знаки, разметка, опоры освещения и др.) Для получения геометрических объектов используется одно значение параметра – “*Layer*”, для получения атрибутивной информации используются отдельные значения для каждого типа – “*Signs*”, “*Roadmarking*”, “*Lighting*”.

Запрос должен содержать 4 обязательных параметра, задающих координаты границ выгружаемых данных об объектах. Параметры задаются в местной системе координат территории – города или муниципального района, где должна проводиться работа. Пятым параметром является название этой территории, при отсутствии параметра берется информация из конфигурации приложения. Шестой параметр определен только для геообъектов с атрибутивной информацией и задает статус выгружаемых объектов: установлен (*Set*), требуется (*Required*), демонтировать (*Dismantle*) либо любой статус (*All*). При отсутствии параметра берется значение по умолчанию – *All*.

При выполнении запроса приложением производится загрузка данных из БД и их преобразование в формат *JSON*, причем координаты получаемых данных преобразуются из местной СК с единицей измерения метр в глобальную СК со значениями в градусах. Пример схемы хранящихся в БД исходных данных для типа геообъекта «Опора освещения» [9] в нотации *IDEF1X* представлен на рисунке 4, а пример содержимого документа *JSON* с отображением этих данных приведено ниже.

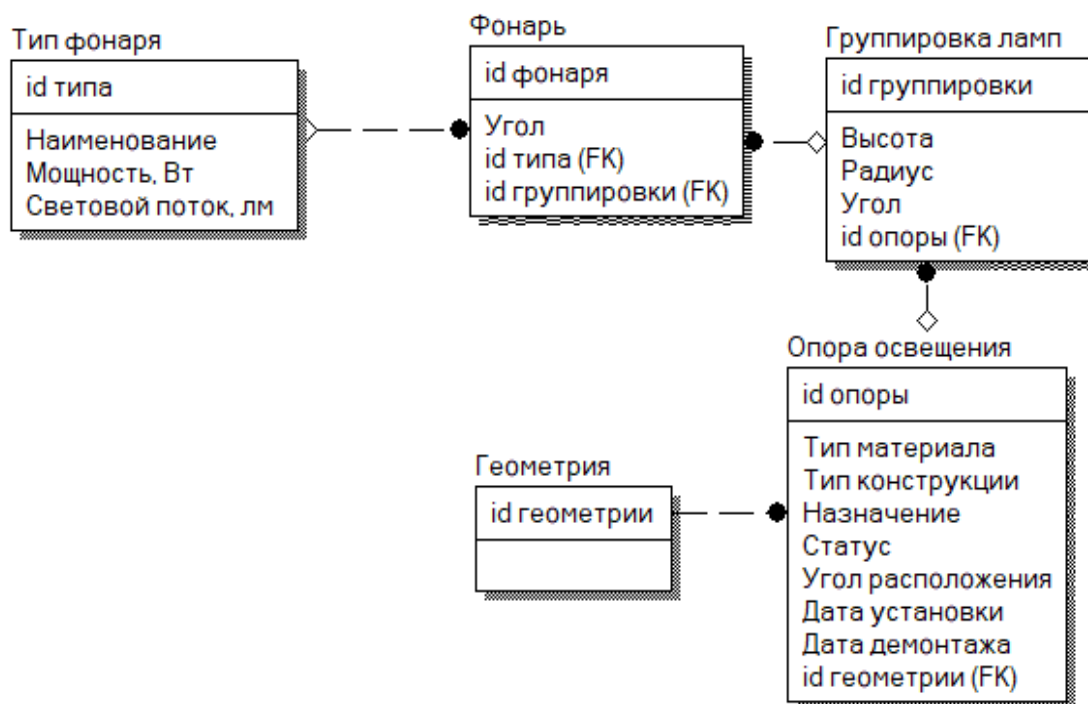


Рисунок 4. Модель данных объекта «Опора освещения»

```
[
  {
    "id": 55083717,
    "angle": 139.763641690726,
    "lamp_type_id": 1,
    "group_id": 55050902,
    "id1": 1,
    "type_name": "ДПЛ 125",
    "type_power": 125,
    "type_flow": 7500,
    "id2": 55050902,
    "height": 6.0,
    "radius": 2.0,
    "angle1": 10.0,
    "post_id": 55018266,
    "id3": 55018266,
    "post_material": "Metal",
    "post_type": "Console",
    "angle2": 0.0,
    "feature_id": 54952953,
    "purpose": "Lamp",
    "status": "Set",
    "installdate": null,
    "lifetime": null,
    "row_id": null,
    "street_id": null,
    "crossstreet_id": null,
    "geometry": {
      "Coordinates": [
        {
          "X": 73.39702601308532,
          "Y": 61.25452145114551,
          "Z": "NaN",
          "M": "NaN"
        }
      ],
      "NumGeometries": 1,
      "IsSimple": true,
      "IsValid": true,
      "Area": 0.0,
      "Length": 0.0,
      "<...>"
    },
    "version": 3
  },
],
```

```
{
  "id": 363792482,
  "angle": -42.8252558853899,
  "lamp_type_id": 2,
  "group_id": 55050902,
  "id1": 2,
  "type_name": "ДПЛ 250",
  "type_power": 250,
  "type_flow": 13500,
  "id2": 55050902,
  "height": 6.0,
  "radius": 2.0,
  "angle1": 10.0,
  "post_id": 55018266,
  "id3": 55018266,
  "post_material": "Metal",
  "post_type": "Console",
  "angle2": 0.0,
  "feature_id": 54952953,
  "purpose": "Lamp",
  "status": "Set",
  "installdate": null,
  "lifetime": null,
  "row_id": null,
  "street_id": null,
  "crossstreet_id": null,
  "geometry": {
    "Coordinates": [
      {
        "X": 73.39702601308532,
        "Y": 61.25452145114551,
        "Z": "NaN",
        "M": "NaN"
      }
    ],
    "NumGeometries": 1,
    "IsSimple": true,
    "IsValid": true,
    "Area": 0.0,
    "Length": 0.0,
  },
  "version": 3
}
```

Заключение

Повышение разнообразия средств разработки современных программных комплексов, включая транспортные и геоинформационные, требует от существующих сложных систем со сформировавшейся кодовой базой постоянного обеспечения взаимодействия с этими комплексами. Наиболее удобным способом такого обеспечения является интеграция программного интерфейса (*API*) в существующую инфраструктуру системы. Программный интерфейс должен обладать возможностью передачи данных от СУБД к произвольным клиентским приложениям, а также как можно меньше зависеть от платформы, на которой он разрабатывался. С учетом этих соображений для интеллектуальной геоинформационной системы «ITSGIS» был успешно разработан интерфейс, обеспечивающий загрузку пространственных и атрибутивных данных о геометриях карты и технических средствах ОДД на любое устройство, подключенное к сети расположения *API*.

Список использованных источников

1. Малыгин, И.Г. Сети, информация и знания – основные драйверы Четвертой индустриальной революции (INDUSTRIE 4.0) / И.Г. Малыгин, В.И. Комашинский, М.Ю. Аванесов [и др.] // Информация и космос. – 2016. – № 1. – С. 14-25.
2. Смолев, А.М. Методы моделирования адресного плана в рамках геоинформационной системы / А.М. Смолев, Т.И. Михеева, А.В. Золотовицкий // IT & Transport / ИТ & Транспорт : Сборник научных статей. – Самара : ИнтелТранС, 2023. – Т. 24. – С. 4-13.
3. Головнин, О.К. Архитектура системы передачи данных ГИС ITSGIS / О.К. Головнин // Новые информационные технологии в научных исследованиях: материалы XX Юбилейной Всероссийской научно-технической конференции студентов, молодых ученых и специалистов (Рязань, 17–19 ноября 2015 г.). – Рязань : РГРТУ, 2015. – С. 278–280.
4. Golovnin, O.K. Territorial Management using ITSGIS / O.K. Golovnin, T.I. Mikheeva, V.A. Klyuchnikov // Intelligent Technologies for Information Processing and Management (ITIPM'2014) : Proceedings of the 2nd International Conference, Ufa, 10–12 ноября 2014 года. Vol. 1. – Ufa: ГОУ ВПО «Уфимский государственный авиационный технический университет», 2014. – С. 211-214.
5. Мартынюк, С.Н. Использование автоматизированных систем видеонаблюдения в обеспечении безопасности дорожного движения / С.Н. Мартынюк, В.Б. Косовский // Ученые записки Крымского феде-

- рального университета имени В.И. Вернадского. Юридические науки. – 2022. – Т. 8(74), № 2. – С. 105-110.
6. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с. – ISBN 978-5-9906857-4-1.
 7. Джалалов, М.Э. Применение шаблонов проектирования для управления API в микросервисной архитектуре / М.Э. Джалалов // Экономика и качество систем связи. – 2024. – № 1(31). – С. 128-136.
 8. Mikheeva, T. Recognition of Urban Transport Infrastructure Objects Via Hyperspectral Images / O. Saprykin, A. Fedoseev, T. Mikheeva // In Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS-2016). Rome, Italy: SCITEPRESS, 2016. – P. 203-208.
 9. Алексеев, В.М. Построение архитектуры интеллектуальной системы управления городской рельсовой транспортной системой / В.М. Алексеев, Л.А. Баранов, М.А. Кулагин, В.Г. Сидоренко // Мир транспорта. – 2021. – Т. 19, № 1(92). – С. 18-46.
 10. Гончаров, Е.Ю. Микросервисная архитектура в контексте разработки интеллектуальных транспортных систем / Е.Ю. Гончаров // Сборник лучших докладов студенческой научно-технической конференции, посвященной 100-летию Отечественной гражданской авиации : Сборник материалов конференции, Московский государственный технический университет гражданской авиации, 13 апреля 2023 года. – Москва: ИД Академии Н.Е. Жуковского, 2023. – С. 77-81.
 11. Михеева, Т.И. Дислокация объектов кабельной сети на электронной карте города / Т.И. Михеева, Р.А. Кирясов, А.А. Осьмушин // Перспективные информационные технологии : труды Международной научно-технической конференции, Самара, 04–06 декабря 2013 года / Самарский государственный аэрокосмический университет им. академика С.П. Королева. – Самара: Самарский научный центр РАН, 2013. – С. 278-281.

Smolev A.M., Mikheeva T.I.

AN IMPLEMENTATION OF THE ITSGIS PROGRAMMING INTERFACE FOR UTILIZATION AT TRANSPORT INFRASTRUCTURE

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

This article describes the implementation of the ITSGIS geographic information system application programming interface (API), designed for loading information regarding entities of various types on an electronic map. The necessity of implementing this plugin is substantiated. It is indicated that to ensure the interaction of the "ITSGIS" components with newly developed modules, it is necessary to eliminate or reduce the dependence on specific software development tools. The methods and tools that were used to integrate the API into the geographic information system are disclosed.

Keywords: API, implementation, transport infrastructure, modular architecture, computer network, geographic information system, integration.

УДК 004.02

Михеева Т.И., Клепиков Н.М.
**УЛУЧШЕННОЕ ОБНАРУЖЕНИЕ ОБЪЕКТОВ
С ИСПОЛЬЗОВАНИЕМ ДРОНОВ И ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА**

*Самарский университет имени академика С.П. Королёва
ИнтелТранс*

В статье раскрывается алгоритм действий для повышения точности распознавания объектов на ортофотоизображениях с помощью методов глубокого обучения. Отмечаются причины, по которым для решения этой задачи желательно использовать наклонные изображения. Также описывается метод ортотрансформации, т.е. совмещения пиксельных координат на распознанных изображениях с глобальными координатами для автоматического переноса данных на электронную карту.

Ключевые слова: обнаружение объектов, искусственный интеллект, ортоизображение, глубокое обучение, электронная карта.

Введение

Одной из задач, с которыми превосходно справляется искусственный интеллект, является обнаружение объектов. Этот метод компьютерного зре-

ния позволяет нам идентифицировать и находить объекты на изображениях и видео. Это бесценный инструмент для картографирования нашего мира, создания цифровых двойников и получения количественного понимания нашей окружающей среды. Это, в свою очередь, способствует принятию решений на основе данных для умных городов, городского планирования, управления ресурсами, защиты окружающей среды и реагирования на стихийные бедствия.

Ортоизображения часто используются для обнаружения объектов, но они не всегда являются лучшими по следующим причинам:

- ограниченная детализация: ортоизображения могут отображать такие объекты, как пожарные гидранты и столбы, в виде точек или кругов. Однако косые изображения, снятые с дронов или камер уличного обзора, предоставляют более подробную информацию об объектах, что облегчает их обнаружение.



Рисунок 1. Формы гидранта на прямом и наклонном изображениях

Ортофотоизображения (слева) ограничивают детали объекта, например, столбы и гидранты отображаются в виде точек или кругов, а наклонные изображения (справа) дают больше деталей:

- слепые пятна: кроны деревьев на ортоизображениях могут полностью скрывать объекты. На левом изображении показано расположение объекта, который мы не можем видеть из-за крон деревьев. На правом изображении показан тот же объект на снимке, сделанном с улицы.



Рисунок 2. Видимость гидранта на ортоизображении и обычном снимке

Навесы деревьев скрывают объект на ортоизображении (слева), а на изображении с видом на улицу он виден (справа):

- Ограниченные модели и наборы данных. Наборы обучающих данных с открытым исходным кодом, такие как набор данных СОСО (*Common Objects in Context*) и предварительно обученные модели, преимущественно доступны для неортоизображений. Они не так хорошо работают на ортоизображениях.



Рисунок 3. Набор тренировочных графических данных СОСО

Практически все изображения в наборе данных СОСО наклонены.

Одним из решений этих проблем является использование наклонных изображений с дронов или камер уличного обзора. Мы можем обнаруживать

объекты в пространстве пикселей наклонных изображений и преобразовывать обнаруженные объекты в пространство карты. Пространство карты использует систему координат на основе карты, а пространство пикселей ссылается на необработанное пространство изображения без поворотов и искажений. Когда модель с эталонной системой пиксельного пространства используется с ортотрансформированной коллекцией изображений, содержащей изображения с информацией о камере, сначала делается вывод в пиксельном пространстве, а затем выведенная геометрия преобразуется в пространство карты с использованием информации о кадре и камере.

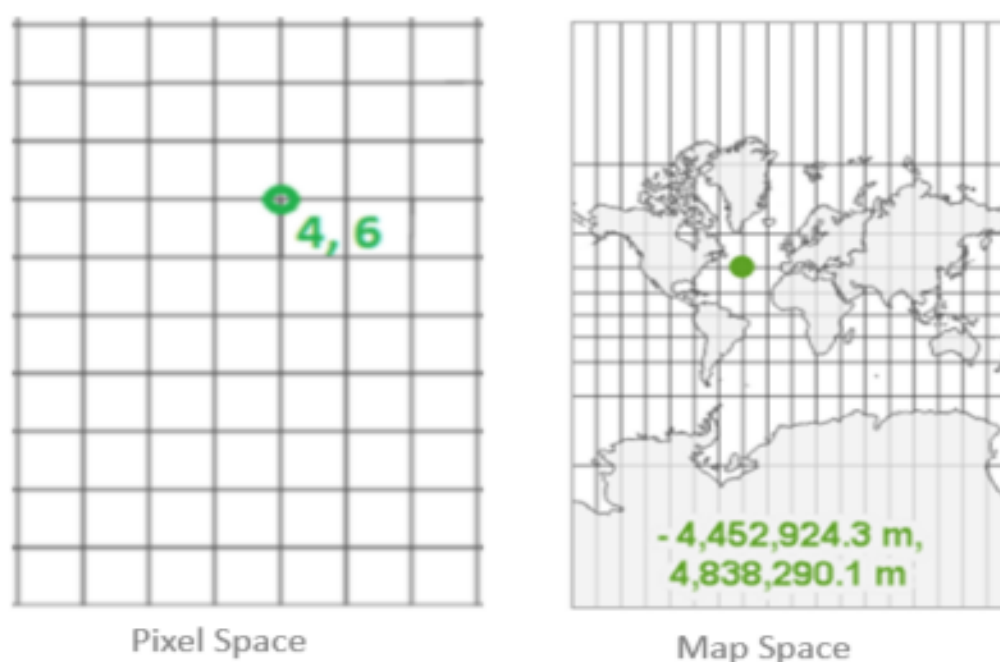


Рисунок 4. Соотнесение экранных (пиксельных) и глобальных координат

Расположение объекта в пространстве пикселей и пространстве карты.

В этом блоге мы будем использовать изображения, сделанные дронами, из нового коммерческого комплекса под названием *Packing House District*, расположенного в Редлендсе, штат Калифорния. Мы построим модель для обнаружения припаркованных автомобилей. Потенциально это может помочь лицам, принимающим решения, понять трафик в этом новом проекте. Наша цель – предложить шаблон рабочего процесса и продемонстрировать его применение на примере.

Для нашего рабочего процесса нам нужны следующие два элемента:

1. Модель, обученная работе с изображениями дронов. Модель должна использовать пиксельное пространство в качестве системы координат.
2. Коллекция ортотрансформированных изображений, содержащая изображения дронов с информацией о камере,

включая фокусное расстояние и другие характеристики датчика. Точность *GPS* важна, укажите ее, если таковая имеется.

Теперь давайте посмотрим на шаги.

Шаг 1. Построение модели

Постройте модель в три этапа:

1. Маркировка изображений: маркировка с помощью панели «Метки объектов для глубокого обучения» с параметрами «Коллекция изображений» и «Пространство в пикселях».
2. Экспорт чипов изображений: затем используйте инструмент «Экспорт обучающих данных для глубокого обучения» с эталонной системой, установленной на «Пространство пикселей», чтобы экспортировать чипы изображений.
3. Модель обучения : наконец, используйте инструмент *Train Deep Learning Model*. Выходная модель будет иметь «*ImageSpaceUsed*»: «*PIXEL_SPACE*» в файле определения модели *Esri (emd)*.

Шаг 2. Создание коллекции ортотрансформированных изображений.

Создайте коллекцию ортотрансформированных изображений в три шага:

1. Создать рабочую область: используйте мастер создания рабочей области ортокартографии.
2. Блокировка: затем используйте инструменты в группах «Настройка» и «Уточнение», чтобы вычислить ориентацию каждого изображения. Если возможно, добавьте опорные точки и связующие точки.
3. Ограничьте дублирование изображений: используйте инструмент «Вычислить элементы-кандидаты», чтобы определить изображения-кандидаты, которые лучше всего представляют область мозаики.

Шаг 3. Обнаружение объектов.

Используйте инструмент «Обнаружение объектов с помощью глубокого обучения» с моделью и коллекцией изображений в качестве входных данных, а для параметра «Режим обработки» установлено значение «Только элементы-кандидаты обработки» и запустите инструмент. Этот режим обработки гарантирует, что обнаружения сначала происходят в необработанном пространстве пикселей каждого изображения, а затем преобразуются в пространство карты коллекции изображений.

Geoprocessing

←

Detect Objects Using Deep Learning

+

Parameters

Environments

?

Input Raster

Image Collection

Output Detected Objects

ImageCollectio_DetectObjects

Model Definition

\\framecamera_workflow\\Model_Percking_house_oblique_cars_e20

Processing Mode

Process candidate items only

Arguments

Name	Value
padding	56
threshold	0.5
nms_overlap	0.1
batch_size	64
exclude_pad_detections	True
test_time_augmentation	False

☒ Non Maximum Suppression

Confidence Score Field

Confidence

Class Value Field

Class

Max Overlap Ratio

0.3

Рисунок 5. Обнаружение объектов с помощью инструмента глубокого обучения

Для прямого сравнения пиксельного пространства и обнаружения пространства пикселей на карте мы запустили инструмент как на одном изображении дрона (пространство пикселей), так и на коллекции изображений (пространство пикселей на карте).



Рисунок 6. Обнаружение автомобилей сбоку

Обнаружения в пиксельном пространстве одного из необработанных изображений дрона.



Рисунок 7. Обнаружение автомобилей сверху

Обнаружения в пространстве карты коллекции изображений.

Заключение

В статье было рассказано о преимуществах наклонных изображений для улучшения обнаружения объектов. Вы изучили шаги, используя набор изображений дрона; однако вы можете применить эти шаги и концепции к изоб-

ражениям, снятым камерами *Streetview*. Кроме того, рабочий процесс можно адаптировать для задач классификации пикселей.

Список использованных источников

1. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с. – ISBN 978-5-9906857-4-1.
2. Клепиков, Н.М. Методы распознавания дорожных знаков в среде ITSGIS / Н.М. Клепиков, Т.И. Михеева // Перспективные информационные технологии (ПИТ 2021). – Самара, 2021. – С. 470-476.
3. Каллан, Р. Основные концепции нейронных сетей / Р. Каллан. – М.: Издательский дом «Вильямс», 2001. – 287 с.
4. Хайкин, С. Нейронные сети: полный курс / С. Хайкин. – М.: Издательский дом «Вильямс», 2006. – 1104 с.
5. Mikheeva, T. Recognition of Urban Transport Infrastructure Objects Via Hyperspectral Images / O. Saprykin, A. Fedoseev, T. Mikheeva // In Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS-2016). Rome, Italy: SCITEPRESS, 2016. – P. 203-208.
6. Михеев, С.В. Диагностика состояния транспортной инфраструктуры с использованием нейронных сетей / С.В. Михеев, А.В. Сидоров, А.А. Осьмушин // Современные проблемы науки и образования. – 2013. – №6. – С. 215-216.
7. Палагута, К.А. Экспериментальное исследование влияния внешней засветки на работу систем распознавания дорожных знаков / К.А. Палагута, Н.А. Игнатович // Инновационные, информационные и коммуникационные технологии. – 2019. – №1. – С.406-409.
8. Тадеусевич, Р. Элементарное введение в технологию нейронных сетей с примерами программ / Р. Тадеусевич. – М.: Горячая линия-Телеком, 2011. – 408 с.
9. Клепиков, Н.М. Исследование алгоритмов распознавания дорожных знаков / Н.М. Клепиков, Т.И. Михеева // IT & Transport. – 2021. – №14. – С. 46-57.
10. Михеева, Т.И. Интеллектуальная транспортная геоинформационная система ITSGIS / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Современные проблемы безопасности жизнедеятельности: интеллектуальные транспортные системы : материалы IV Международной научно-практической конференции (Казань, 25–26 февраля 2016 г.). – Казань : ГБУ «Научный центр безопасности жизнедеятельности», 2016. – С. 362–368. – ISBN 978-5-85247-837-5.

11. Михеева, Т.И. Метод синтеза системы зонального сетевидного управления транспортными процессами / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Известия Самарского научного центра Российской академии наук. – 2016. – Т. 18, № 4 (4). – С. 799–807.

Mikheeva T.I., Klepikov N.M.

ADVANCED OBJECT DETECTION USING DRONES AND ARTIFICIAL INTELLIGENCE

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

The article describes an algorithm of actions for increasing the accuracy of object recognition in orthographic images using deep learning methods. The reasons why it is desirable to use oblique images to solve this problem are noted. Also described in this article is the orthographic transformation method, i.e. combining pixel coordinates in recognized images with global coordinates for automatic data transfer to an electronic map.

Keywords: object detection, artificial intelligence, orthoimage, deep learning, electronic map

УДК 004.02

Михеев С.В.

СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ МЕТОДИЧЕСКИЕ Р-МОДЕЛИ НА ОСНОВЕ ПАТТЕРНОВ

*Самарский университет имени академика С.П. Королёва
ИнтелТранс*

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. Вводятся основные определения паттерного проектирования P -моделей, специфицируются виды и структуры паттернов, определяется организация конструирования объектов на основе P -моделей. В целом вводимые в главе аксиоматика и систематизация базируются на использовании оригинальных определений и составляет концепцию стратифицированного паттерного конструирования объектов и ассоциаций.

Ключевые слова: обнаружение объектов, паттерн, логическая модель данных, ортоизображение, модульная архитектура, геоинформационная система, проектирование, адресный план.

Введение

В имплементативной основы проектирования поддержки процессов принятия решения при управлении транспортной инфраструктурой паттерновая модель. Синтез системы поддержки принятия решения при управлении транспортной инфраструктурой производится на фундаментальной методологической основе, таксономических и агрегатных паттернов свойств. Синтез универсальных структур на основе паттернов рассмотрен на формальной декларации объектов, инвариантной к их предметной ориентации. Наряду с синтезом методических P -моделей описаны паттерны формализации и систематизации релевантных информационно-логических и функциональных аспектов транспортной инфраструктуры. Имплементативные P -модели содержат результативности паттерного проектирования системы поддержки принятия решения при управлении транспортной инфраструктурой процессов моделирования, управления, экспериментальных исследований. Синергетические P -модели выступают, как часть системного анализа общих принципов функционирования сложноорганизованной системы, в качестве которой рассматривается система поддержки принятия решения.

Стратифицированная паттерновая модель системы поддержки

Паттерн – образец, шаблонная модель, формализованное описание часто встречающейся задачи проектирования, эффективное, в заданном контексте, типовое решение проектной (программной) проблемы. Определим паттерновую модель, как P -модель.

Стратифицированная P -модель декларирует статический регламент структурированного конструирования классов объектов и межклассовых отношений.

Стратифицированная P -модель декларирует страты, как объекты, получающие онтологический смысл и описывающие элементы нелинейно-динамической системы, в качестве которой выбрана система поддержки принятия решений управления транспортной инфраструктурой. Межклассовые отношения, специфицированные P -моделью, не могут быть изменены в динамике процесса имитации. Статичность регламента не означает статичности объектов, конструируемых по этому регламенту, они могут динамически реконструироваться в соответствии со структурой межклассовых отношений в модели.

Визуально P -модель представим совокупностью вложенных сферических страт, определяемых триадой $\langle \hat{S}, (\Psi_1, \Psi_2, \dots, \Psi_n), \Omega \rangle$, где $\hat{S}$ – множество страт пространства предметной области, $\Psi_1, \Psi_2, \dots, \Psi_n$ – множество типов отношений между стратами пространства, Ω – множество отображений.

Сферическая страта представляет собой классы объектов и, в свою очередь, может быть декомпозирована для получения необходимого уровня детализации. Декомпозиция представляется вырезкой конуса из общего пространства предметной области (рисунок 1).

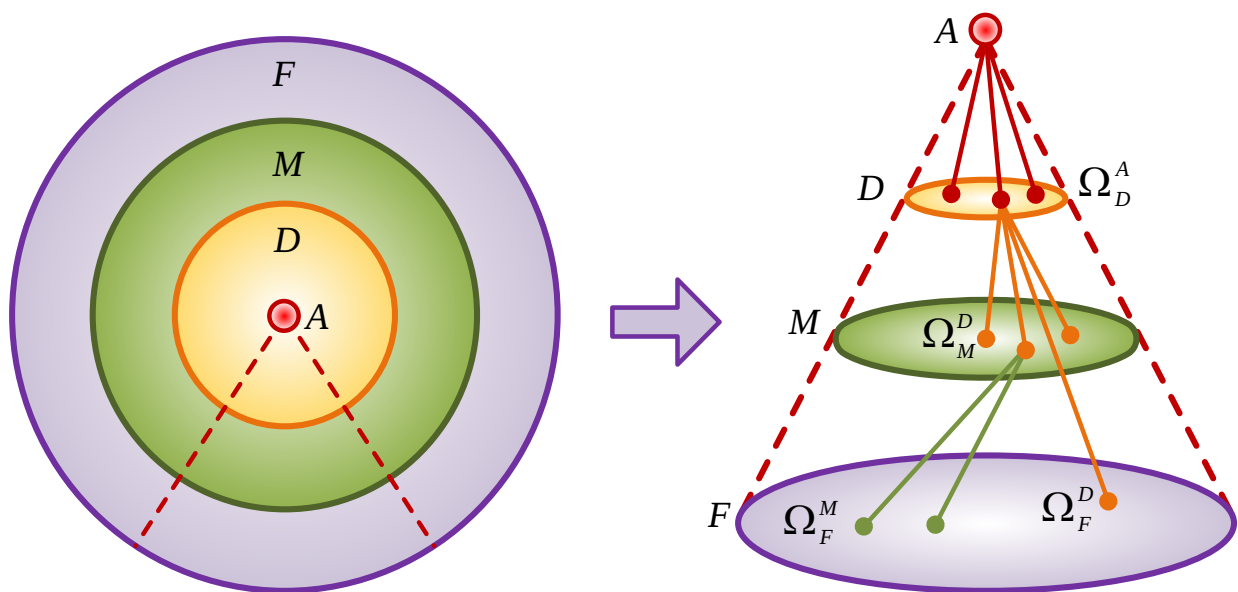


Рисунок 1. Стратифицированная P -модель

Паттерн *единичного наследования* – таксон $C(\tilde{P}^c)$, в котором любой класс непосредственно обобщается только одним другим, имеет иерархическую структуру дерева, корнем которого является базовый класс, $\forall \tilde{P}^c(A), \tilde{P}^c(B), \tilde{P}^c(C) \mid (A * C) \& (B * C) \Rightarrow (A * B) \text{ OR } (B * A)$ и иллюстрирует

интерпретацию обобщения отношением «*kind_of*» («Имеет_разновидности»): $(C * \rangle A), (C * \rangle B) = (C * \rangle A) \Rightarrow \tilde{P}^C(C) \text{ kind_of } \tilde{P}^C(A), \tilde{P}^C(B)$.

На рисунке 2 приведен пример паттерна единичного наследования *Pattern* $\Rightarrow$ *PSingleTaxon* для классов *A* – пешеходный переход, *B* – перекресток улично-дорожной сети, *C* – участок дороги.

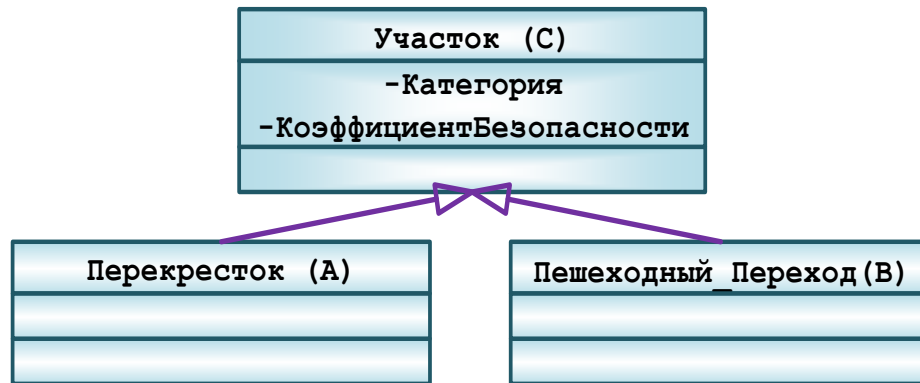


Рисунок 2. Паттерн единичного наследования

Паттерн *множественного наследования* – таксон $C(\tilde{P}^C)$, в котором путем объединения нескольких базовых основ с ограниченным спектром значений конструируется новый класс объектов с более широким спектром, $\tilde{P}^C(A) \text{ in } \tilde{P}^C(C) \text{ in } \tilde{P}^C$; $\tilde{P}^C(B) \text{ in } \tilde{P}^C(C) \text{ in } \tilde{P}^C$; $\tilde{P}^C(A) \cap \tilde{P}^C(B) = \emptyset$, и иллюстрирует интерпретацию наследования частей «*part_of*» («Партномия»).

На рисунке 3 приведен пример паттерна множественного наследования *Pattern* $\Rightarrow$ *PMultipleTaxon* для классов *A* – участок дороги, *B* – светофор, *C* – участок дороги со светофором.

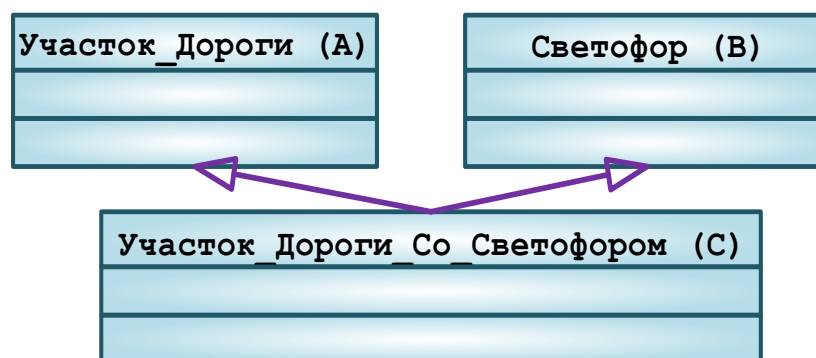


Рисунок 3. Паттерн множественного наследования

Наследуемость паттерном «*is_Instance_Of_a*» вдоль отношения «*is_a*» определяется следующим утверждением: если *A* является подклассом класса *B*, то каждый экземпляр класса *A* также является экземпляром класса *B*.

Экземпляры класса «Светофор» не могут быть помещены в иерархию ниже «Участок_дороги_со_светофором», даже в подкатегорию «Группа_светофорных_объектов», или стать экземпляром этой категории.

В общем случае *P*-модель наследования, основанная на отношении обобщения, определяется иерархической структурой, называемой *таксономическим деревом*. Множество имманентных свойств предметной области определяет базу знаний о предметной области, а таксономическая модель – систематизацию этих знаний на основе построения иерархической структуры понятий – классов предметной области. При этом инструментальные классы, размещаемые в корнях таксономического дерева и содержащие общие свойства класса, могут быть разработаны автономно, как *CASE*–инструменты проектирования объектов и ассоциаций, независимо от их предметной ориентации. В этом случае разработка моделей предметной области базируется на заранее подготовленной и отлаженной основе, что позволяет в процессе проектирования реализовать новые программные «наслоения».

P-модель абстрактной предметной области определяется триадой $\langle \text{Класс}, \text{Свойство}, \text{Отношение} \rangle$: $P(\text{Domain}) = \langle C, \tilde{P}^C, \tilde{R}^C \rangle$,

где C – набор непустых множеств классов объектов предметной области;

$\tilde{P}^C$ – имманентные свойства объектов класса;

$\tilde{R}^C$ – отношения – предикаты на основных множествах, истинность которых свидетельствует о корректности *P*-модели.

Все включаемые в спецификацию классы одной основы имеют уникальные имена:

$$\forall c_i \text{ has_a } (\tilde{P}^C(C_i), \text{name}_i), c_j \text{ has_a } (\tilde{P}^C(C_j), \text{name}_j) \in C : \\ \text{name}_i = \text{name}_j \rightarrow i = j.$$

Декларативные спецификации классов дизъюнктивны:

$$\forall c_i \text{ has_a } (\tilde{P}_d^C(C_i), name_i), c_j \text{ has_a } (\tilde{P}_d^C(C_j), name_j) \in C : \\ \tilde{P}_d^C(C_i) = \tilde{P}_d^C(C_j) \rightarrow i = j.$$

Декларативная составляющая $\tilde{P}_d^C$ концептуальной модели системы управления Три регламентирует структуру информационного объекта (общего шаблона базы данных), каким является объектная модель системы, и всякая поименованная репликация такого паттерна порождает пустую модель (рисунок 4).



Рисунок 4. Паттерн информационного объекта – базы данных

Декларативные Р-модели

Р-модели различаются уровнем абстракции и степенью детализации: архитектурные паттерны, паттерны предметной области, проектирования, аналитические, тестирования и реализации (рисунок 5).

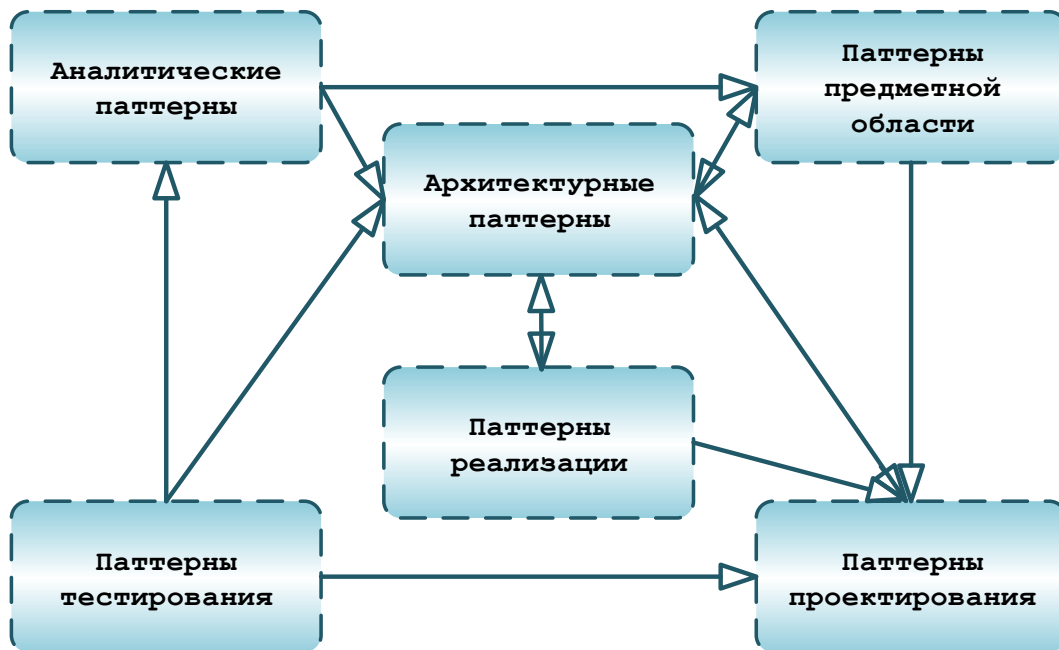


Рисунок 5. Взаимодействие паттернов

Архитектурные паттерны ($Pattern \Rightarrow P_{Architectural}$) – множество предварительно определенных подсистем со спецификацией их ответственности, правил и базовых принципов установления отношений между ними. Спецификация фундаментальной схемы структуризации программной системы является основоположением архитектурных паттернов. Ярким примером архитектурных паттернов являются паттерны распределения ответственности *GRASP* (*General Responsibility Assignment Software Pattern*). Архитектура отражает потребности решаемой задачи, но, в то же время, – является общей, чтобы соответствовать требованиям к системе, которые могут возникнуть в будущем. Эти паттерны относятся к уровню системы и подсистем, но не к уровню классов, формулируются в обобщенной форме, используют обычную терминологию и не зависят от области приложения.

Архитектура системы управления ТрИ имеет иерархическую структуру последовательно вложенных друг в друга составляющих ее элементов (рисунок 6). Классы $C(\tilde{P}^c) = \{c \setminus \tilde{P}^c\}$ инкапсулируют поля $\tilde{P}_d^c$ и методы $\tilde{P}_f^c$ согласно принципам объектно-ориентированной парадигмы программирования.

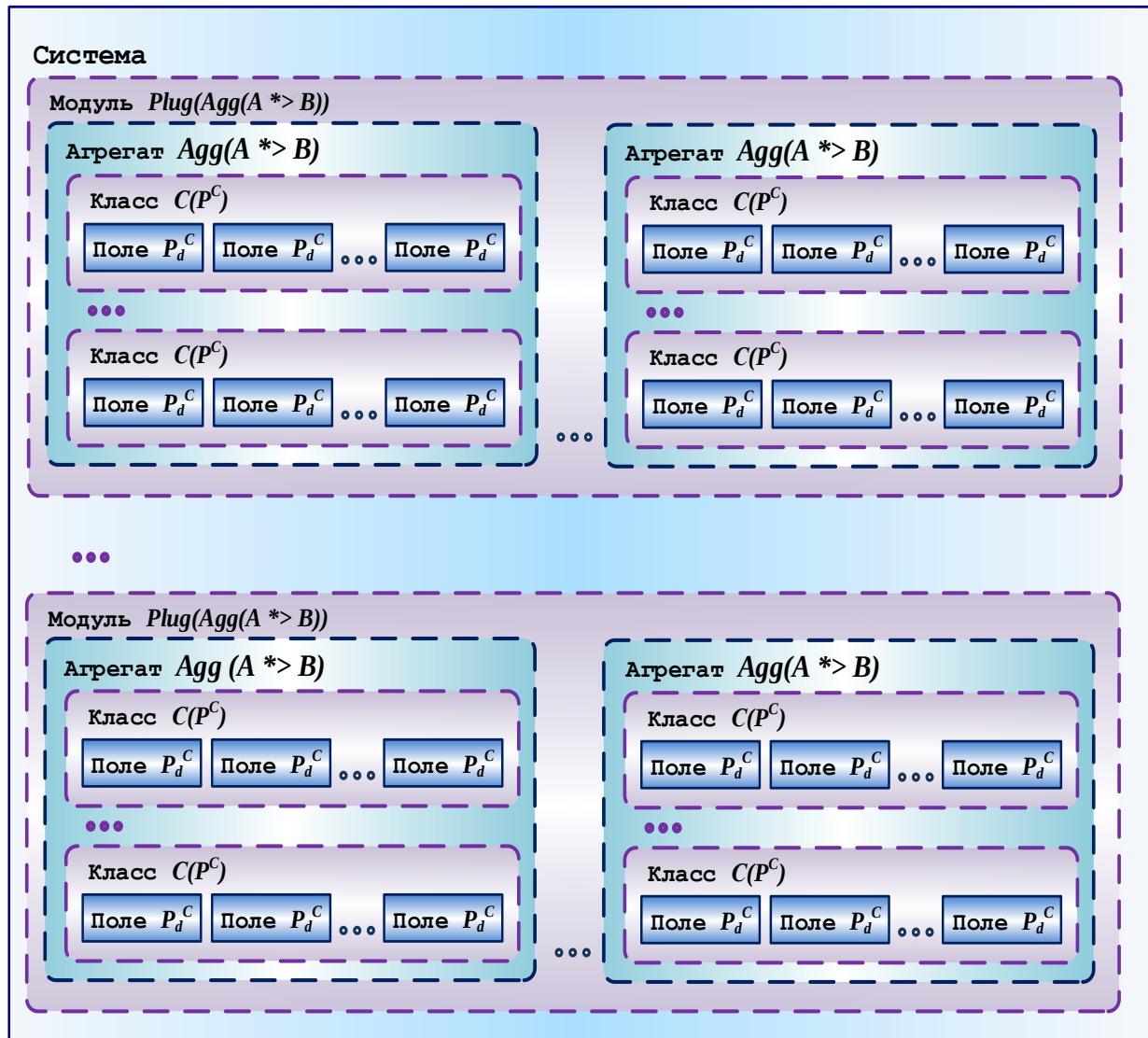


Рисунок 6. Архитектурный паттерн элементов системы

Агрегация классов объектов $Agg(A *> B)$, обеспечивающая обобщение декларативных спецификаций $\tilde{P}_d^C$ классов, декларирует интерпретацию наследования частей «part_of». Каждый модуль $Plug(Agg(A *> B))$ содержит множество агрегатов – совокупностей взаимосвязанных классов, которые воспринимаются как единое целое. Модули $Plug$ соответствуют смысловым частям модели, выражая суть и структуру модели в крупном масштабе, находятся на верхнем уровне иерархии инструментария системы управления ТРИ. Все элементы архитектуры взаимодействуют между собой как в пределах содержащих их контейнеров, так и между контейнерами. При синтезе системы для оптимального распределения элементов архитектуры между контейнера-

ми необходимо определить скрытые фундаментальные особенности взаимодействия объектов системы и свойств $\tilde{P}^C$ их формирующих с учетом атрибутов.

Паттерн формирования атрибутов класса, согласно декларативной спецификации класса, имеет структуру: $\langle Quantifier_visibility \rangle \langle Name_Attribute \rangle [Limit]: \langle Type_Attribute \rangle = \langle Initial_Value \rangle \{Immanent\ Property\}$, в которой *Quantifier_visibility* – определяет уровень видимости и доступности атрибута класса; *Name_Attribute*, *Type_Attribute* – имя атрибута и его тип, соответственно; *Limit* содержит замкнутый интервал целых чисел, для обозначения количества атрибутов данного типа; *Initial_Value* – предназначен для задания начального значения атрибута во время создания объекта класса; *Immanent Property* – имманентные свойства, определяющие фиксированное, неизменяемое значение атрибута для класса в целом.

Паттерны геопространственного зонирования $Pattern \Rightarrow PGeo_Zona$.

Паттерны геопространственного зонирования обеспечивают инкапсуляции пространственных функций дислокации и пространственного анализа с учетом семантики объектов транспортной инфраструктуры. Геопространственная однородность позволяет построить кластер объектов, характеризующийся атрибутом дислокации объекта на тематическом слое электронной карты геоинформационной системы.

Декомпозиция геопространственного зонирования представляется тремя классами эквивалентности зон управления: точечная геозона управления $\tilde{A}^D$; линейная геозона управления $\tilde{A}^S$; полигональная геозона управления $\tilde{A}^P$.

Геопространственная зона управления транспортной инфраструктурой $\tilde{A} = \{\tilde{a}_i^x\}$, $(\tilde{A} \neq \emptyset)$ содержит подмножества:

$\tilde{A}^D \subset \tilde{A}$ – множество точечных геопространственных зон управления;

$\tilde{A}^S \subset \tilde{A}$ – множество линейных геопространственных зон управления;

$\tilde{A}^P \subset \tilde{A}$ – множество полигональных геопространственных зон управле-

ния.

Точечная геозона управления $\tilde{A}^D = \{\tilde{a}_i^D\}$, $i = 1, 2, \dots, n$ – зона, содержащая точечные объекты пространственной модели транспортной инфраструктуры, дислоцированных на электронной карте ГИС. Множество точечных геозон управления отвечает условиям: $\tilde{A}^D \neq \emptyset$; $\bigcap_{i=1}^n \tilde{a}_i^D = \emptyset$. В качестве точечных объектов выступают объекты классов «Дорожный_знак», «Светофор», «Транспортное_средство», «Световая_опора».

Линейная геозона управления $\tilde{A}^S = \{\tilde{a}_j^S\}$, $j = 1, 2, \dots, m$ – зона, содержащая линейные объекты пространственной модели транспортной инфраструктуры. Множество линейных геозон управления отвечает условиям: $\tilde{A}^S \neq \emptyset$; $\bigcap_{j=1}^m \tilde{a}_j^S = \emptyset$. В качестве линейных объектов выступают объекты классов «Дорожное_ограждение», «Линия_дорожной_разметки», «Осевая_линия_проезжей_части».

Полигональная геозона управления $\tilde{A}^P = \{\tilde{a}_k^P\}$, $k = 1, 2, \dots, l$ – зона, содержащая полигональные объекты пространственной модели транспортной инфраструктуры. Множество полигональных геозон управления отвечает условиям: $\tilde{A}^P \neq \emptyset$; $\bigcap_{k=1}^l \tilde{a}_k^P = \emptyset$. В качестве полигональных объектов выступают объекты классов «Перекресток», «Перегон», «Административный_район».

Пространственная зона управления $\tilde{A}$ не может быть одновременно определена как точечная зона $\tilde{A}^D$, линейная $\tilde{A}^S$ или полигональная $\tilde{A}^P$.

$$\tilde{A} = \tilde{A}^D \cup \tilde{A}^S \cup \tilde{A}^P. \quad \tilde{A}^D \cap \tilde{A}^S \cap \tilde{A}^P = \emptyset$$

Проектирование модели управления транспортными процессами использует геопространственную привязку технических средств управления, синтез топологических геозон управления. Пример онтологии зон и видов управления транспортной инфраструктурой с учетом атомарных паттернов таксономии представлен на рисунке 7.

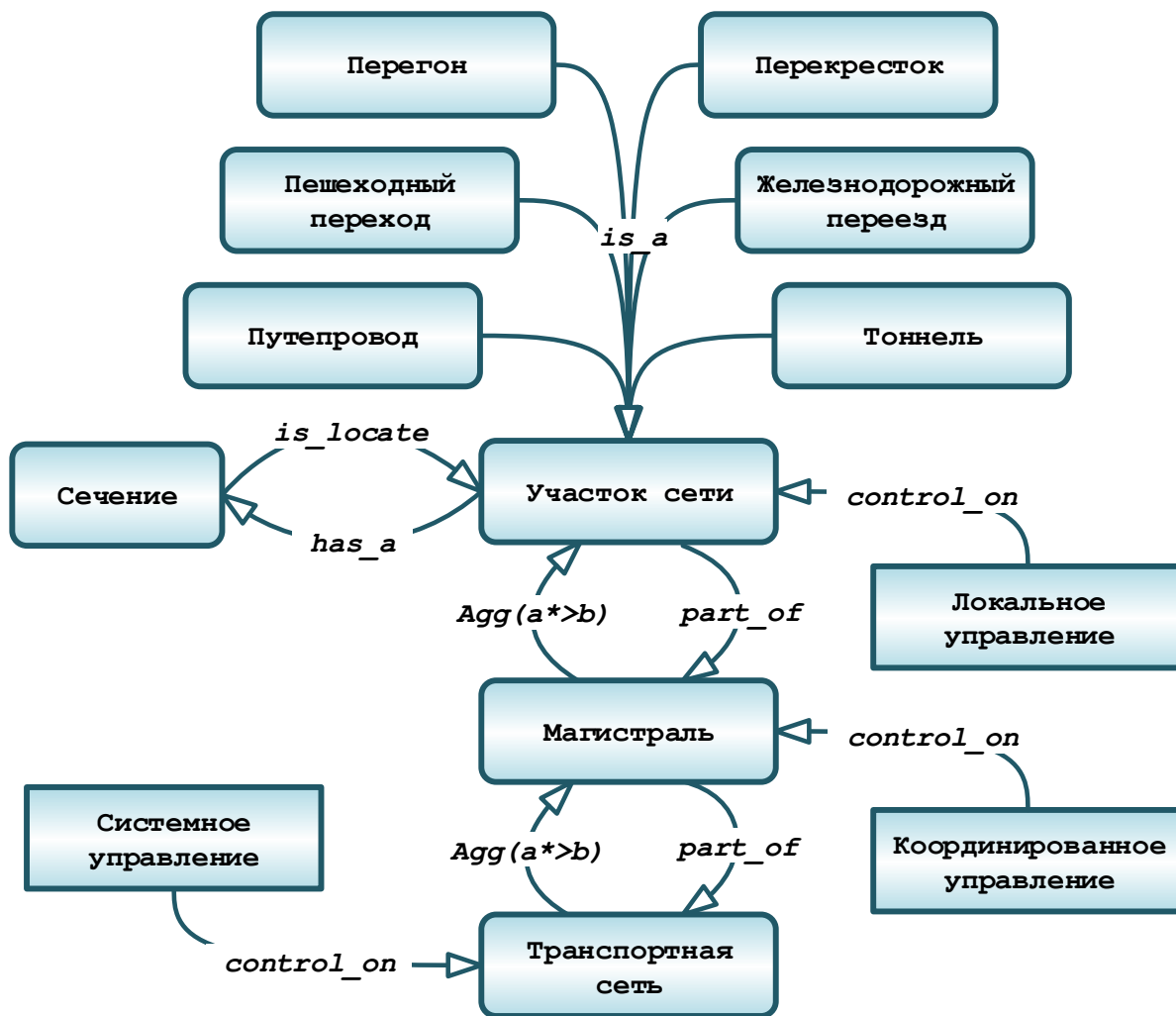


Рисунок 7. Онтология зон и видов управления

В задачах моделирования управления транспортной инфраструктурой, интерактивными транспортными процессами, резонно использовать шаблонный метод обработки событий – паттерн *Pattern* $\Rightarrow$ *PEvent_Control*, отображающий транспортную сеть визуализации текущих событий. Паттерн *PEvent_Control* представляет собой концепцию обработки интерактивных ситуаций (штатных или нештатных) (рисунок 8).

Информация о возникшей ситуации инкапсулируется в событие *Event*, идентифицирующее ситуацию. Событие генерируется в источнике (производитель события), в качестве которого выступают информационные средства. Событие доставляется потребителю события, в качестве которого выступает система управления или организация, ответственная за принятие мер, например, по ликвидации нештатной ситуации.

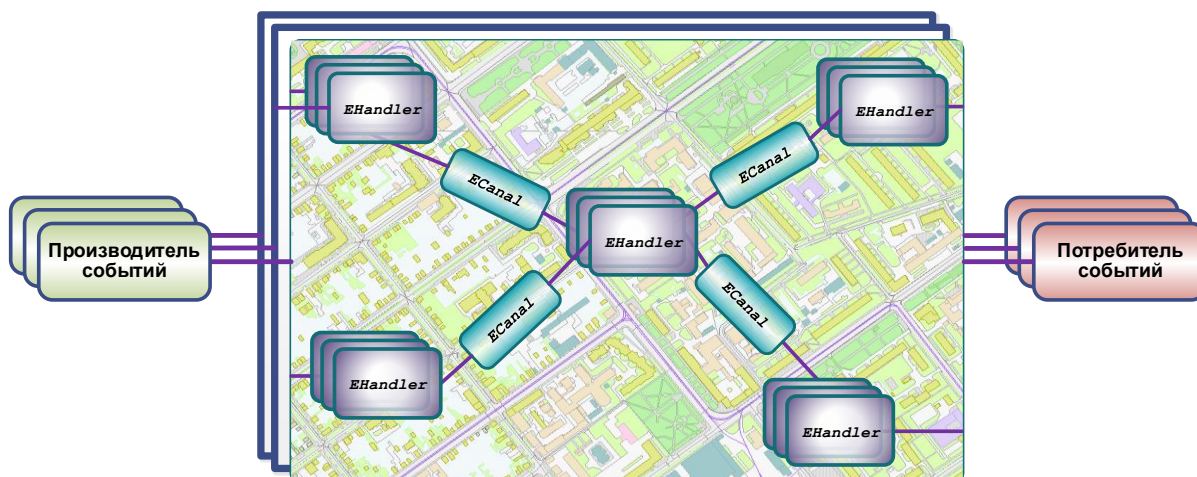


Рисунок 8. Паттерн «Обработка событий»

Наряду с синтезом методических *P*-моделей определены паттерны формализации и систематизации релевантных информационно-логических и функциональных аспектов транспортной инфраструктуры, разработаны паттерны формального преобразования исходной информации в паттерновую спецификацию, реализующую все ключевые формы семантического абстрагирования, моделирования, зонального управления и анализа транспортной инфраструктуры на основе методов искусственного интеллекта.

Имплементативные *P*-модели содержат результативности паттерного проектирования системы поддержки принятия решения при управлении транспортной инфраструктурой процессов моделирования, управления, экспериментальных исследований. Этот дает качественное объяснение основному феномену фазы концептуализации в задачах построения сложноорганизованных систем – возможности формирования и сосуществования различных типов моделей в рамках одной предметной области.

Список использованных источников

1. Михеев, С.В. Паттерны проектирования сложноорганизованных систем / Т.И. Михеева, С.В. Михеев, О.К. Головин, О.Н. Сапрыкин. – Самара : Интелтранс, 2015. – 216 с.
2. Щербаков, М.В. Проблемы проектирования систем прогнозирования эксплуатационного состояния автомобильных дорог на основе нечётких нейронных сетей / М.В. Щербаков, Д.А. Скоробогатченко, А.А. Авдеев, М.А. Аль-Гунаид // Известия Волгоградского государ-

- ственного технического университета : межвуз. сб. научн. ст. № 3(76) / ВолГТУ. Волгоград : ИУНЛ ВолГТУ, 2011. – С. 82 – 87.
3. Михеев, С.В. Использование паттерна проектирования «Decorator» в задачах построения маршрута перевозки опасного груза / С.В. Михеев, Т.И. Михеева, И.А. Рудаков, И.А. Чугунов // Математическое моделирование информационных процессов и систем в науке, технике и образовании: Межвузовский сборник научных трудов. Самарск. гос. арх.-строит. ун-т. – Самара, 2010. – С. 85–88.
 4. Интеллектуальная транспортная геоинформационная система ITSGIS. Ядро / Т.И. Михеева, С.В. Михеев, О.К. Головнин, А.В. Сидоров, Е.А. Савинов. – Самара : Интелтранс, 2016. – Т.1. – 171 с. – ISBN 978-5-9906857-4-1.
 5. Токарчук, А.М. Паттерн для отражения бизнес-логики / А.М. Токарчук // Мир транспорта. – 2010. – Т. 30, № 2. – С. 114 – 118.
 6. Mikheeva, T. Recognition of Urban Transport Infrastructure Objects Via Hyperspectral Images / O. Saprykin, A. Fedoseev, T. Mikheeva // In Proceedings of the 2nd International Conference on Vehicle Technology and Intelligent Transport Systems (VENITS-2016). Rome, Italy: SCITEPRESS, 2016. – P. 203-208.
 7. Сапрыкина, О.В. Построение архитектуры аналитического инструментария интеллектуальной транспортной системы на основе паттернов проектирования [Текст] / О. В. Сапрыкина, Т. И. Михеева, О.Н. Сапрыкин // Вестник Самарского гос. техн. ун-та. Серия «Технические науки». – 2010. – №4 (27). – С. 27 – 35.
 8. Михеев, С.В. Диагностика состояния транспортной инфраструктуры с использованием нейронных сетей / С.В. Михеев, А.В. Сидоров, А.А. Осьмушин // Современные проблемы науки и образования. – 2013. – №6. – С. 215-216.
 9. Норенков, И.П. Поддержка принятия решений на основе паттернов проектирования / И.П. Норенков, М.Ю. Уваров // Наука и образование. – 2011. – № 9. – Режим доступа : <http://technomag.edu.ru/doc/228646.html>.
 10. Михеева, Т.И. Интеллектуальная транспортная геоинформационная система ITSGIS / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Современные проблемы безопасности жизнедеятельности: интеллектуальные транспортные системы : материалы IV Международной научно-практической конференции (Казань, 25–26 февраля 2016 г.). – Казань : ГБУ «Научный центр безопасности жизнедеятельности», 2016. – С. 362–368. – ISBN 978-5-85247-837-5.

11. Курилкин, В.В. Перспективная САПР с использованием паттернов / В.В. Курилкин // Известия высших учебных заведений. Серия : Машиностроение. – 2012. – № 13. – С. 26 – 29.
12. Михеева, Т.И. Метод синтеза системы зонального сетцентрического управления транспортными процессами / Т.И. Михеева, С.В. Михеев, О.К. Головнин // Известия Самарского научного центра Российской академии наук. – 2016. – Т. 18, № 4 (4). – С. 799–807.
13. Кузьмич, Р.И. Модификация целевой функции при построении паттернов для увеличения различности правил в модели классификации / Р.И. Кузьмич, И.С. Масич // Системы управления и информационные технологии. – 2014. – № 2 (56). – С. 14 – 17.

Mikheev S.V.

DECISION SUPPORT SYSTEMS METHODOLOGICAL P-MODELS BASED ON PATTERNS

*Samara, Samara University named after academician S.P. Korolev
IntelTrans*

The original techniques of object-oriented design based on pattern theory were used in the development. The construction and reconstruction of such patterns requires tools that provide these processes with universal means of creation and dynamic modification of objects. The basic definitions of the pattern design of P-models are introduced, the types and structures of patterns are specified, and the organization of the construction of objects based on P-models is determined. In general, the axiomatics and systematization introduced in the chapter are based on the use of original definitions and constitute the concept of stratified pattern construction of objects and associations.

Keywords: object detection, pattern, logical data model, orthoimage, modular architecture, geoinformation system, design, address plan.

СПИСОК АВТОРОВ СБОРНИКА

Михеева Татьяна Ивановна, доктор технических наук, профессор кафедры информационных систем и технологий Самарского университета (национального исследовательского университета), генеральный директор группы компаний «ИнтелТранС» (ITSGIS).

Михеев Сергей Владиславович, кандидат технических наук, доцент кафедры организации и управления перевозками на транспорте Самарского университета (национального исследовательского университета).

Смолев Александр Михайлович, диссертация Самарского университета им. академика С.П. Королева (национального исследовательского университета), программист отдела программирования научно-производственного центра «Интеллектуальные транспортные системы».

Клепиков Никита Михайлович, диссертация Самарского университета им. академика С.П. Королева (национального исследовательского университета), кафедра технической кибернетики института информатики, математики и электроники.

СОДЕРЖАНИЕ

<i>Михеева Т.И., Смолев А.М. ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ СИСТЕМЫ «ITSGIS» В ЗАДАЧАХ ОПТИМИЗАЦИИ ИЗМЕРЕНИЯ ИНТЕНСИВНОСТИ.....</i>	<i>4</i>
<i>Смолев А.М., Михеева Т.И. ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АДРЕСНОГО ПЛАНА ГЕОИНФОРМАЦИОННОЙ СИСТЕМЫ НА ПАТТЕРНОВОЙ ОСНОВЕ.....</i>	<i>18</i>
<i>Смолев А.М., Михеева Т.И. РЕАЛИЗАЦИЯ ПРОГРАММНОГО ИНТЕРФЕЙСА «ITSGIS» ДЛЯ ПРИМЕНЕНИЯ НА ТРАНСПОРТНОЙ ИНФРАСТРУКТУРЕ.....</i>	<i>30</i>
<i>Михеева Т.И., Клепиков Н.М. УЛУЧШЕННОЕ ОБНАРУЖЕНИЕ ОБЪЕКТОВ С ИСПОЛЬЗОВАНИЕМ ДРОНОВ И ИСКУССТВЕННОГО ИНТЕЛЛЕКТА.....</i>	<i>42</i>
<i>Михеев С.В. СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ МЕТОДИЧЕСКИЕ Р-МОДЕЛИ НА ОСНОВЕ ПАТТЕРНОВ.....</i>	<i>50</i>

Сетевое издание

IT & TRANSPORT / ИТ & ТРАНСПОРТ**Редакционная коллегия****Главный редактор:**

Михеева Татьяна Ивановна – доктор технических наук, профессор, заслуженный деятель науки и образования РФ», (Самара)

Заместители главного редактора:

Прохоров Сергей Антонович – д.т.н., заместитель генерального директора ООО «НПЦ «ИТС», заслуженный работник высшей школы РФ, академик Академии навигации и управления движением, доктор технических наук, профессор, (Самара),

Филиппова Анна Сергеевна – д.т.н., «Башкирский государственный педагогический университет им. М. Акмуллы», (Уфа)

Секретарь редакционной коллегии:

Чекина Елена Владимировна – ООО «НПЦ «ИТС», (Самара)

Редакционный совет:

Бурдин Антон Владимирович – д.т.н., «Санкт-Петербургский Государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича», (Санкт-Петербург),

Гераськин Михаил Иванович – д.э.н., Самарский университет, (Самара),

Жанказиев Султан Владимирович – д.т.н., интеллектуальные транспортные системы МАДИ, (Москва),

Заболотнов Юрий Михайлович – д.т.н., Самарский университет, (Самара),

Зырянов Владимир Васильевич – д.т.н., РГСУ, (Ростов-на-Дону),

Иващенко Антон Владимирович – д.т.н., Передовая медицинская инженерная школа Самарского государственного медицинского университета Министерства здравоохранения Российской Федерации, (Самара),

Минниханов Рифкат Нургалиевич – д.т.н., ГБУ «Безопасность дорожного движения», (Казань),

Хайтбаев Валерий Абдурахманович – д.э.н., Самарский университет, (Самара), Самарский университет, (Самара),

Зеленко Лариса Сергеевна – к.т.н., Самарский университет, (Самара),

Золотовицкий Аркадий Владимирович – к.т.н., Самарский университет, (Самара),

Михеев Сергей Владиславович – к.т.н., ООО «Научно-производственный центр «Интеллектуальные транспортные системы», (Самара),

Сапрыкин Олег Николаевич – к.т.н., Самарский университет, (Самара),

Сапрыкина Ольга Валерьевна – к.т.н., Самарский университет, (Самара),

Федосеев Александр Андреевич – к.т.н., ОАО «РКУ «Прогресс», (Самара),

Шопин Андрей Геннадьевич – к.т.н., ООО «СМС-информационные технологии», (Самара).

Издательство ООО «НПЦ «ИТС»
443099, Самарская обл., г.о. Самара, г. Самара,
вн. р-н Самарский, ул. Куйбышева, д. 71, оф. 2
Подписано к выпуску 17.06.2024.
Распространяется бесплатно

Учредитель
ООО «НПЦ «ИТС»
Адрес редакции журнала «ИТ & Транспорт»
443125, г. Самара, пр. Кирова, 328, 67