

Министерство образования и науки РФ

Федеральное государственное автономное образовательное учреждение
высшего образования
«Севастопольский государственный университет»



Институт «Информационные технологии и
управление в технических системах»

Кафедра «Информационные технологии и компьютерные системы»

СБОРНИК СТАТЕЙ
всероссийской студенческой
научно-технической конференции

«В МИРЕ КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ»



г. Севастополь
2 – 6 апреля 2018 г.

УДК 004

Научный редактор:

канд. техн. наук, доцент кафедры информационных технологий и компьютерных систем (СевГУ) **Е.Н. Мащенко**

Редакционная коллегия:

профессор (СевГУ), д-р техн. наук **А.В. Скатков**,
профессор (СевГУ), д-р техн. наук **Ю.К. Апраксин**,
канд. техн. наук, доцент (СевГУ) **А.А. Брюховецкий**,
канд. техн. наук, доцент (СевГУ) **В.И. Шевченко**,
канд. техн. наук, доцент (СевГУ) **Е.Н. Мащенко**,
канд. техн. наук, доцент (СевГУ) **О.В. Ченгарь**

Ответственный за выпуск:

канд. техн. наук, доцент кафедры информационных технологий и компьютерных систем (СевГУ) **О.В. Ченгарь**

Мир компьютерных технологий: Сборник статей всероссийской студенческой научно-технической конференции, г. Севастополь, 2 – 6 апреля 2018 г/ М-во образования и науки РФ, Севастопольский государственный университет; науч. ред. Е.Н. Мащенко – г. Севастополь: СевГУ, 2018. – 181 с.
Режим доступа: <http://lib.sevsu.ru:8080/xmlui/handle/123456789/8365> — Загл. с титул. экрана.

В сборник входят статьи по материалам докладов студентов и магистрантов–участников всероссийской студенческой научно-технической конференции (2-6 апреля 2018г.), по направлениям: «Современные направления моделирования в компьютерной инженерии, образовании и науке», «Робототехника. Технологии искусственного интеллекта», «Интеллектуальный анализ данных. Технологии Big Data. Системы поддержки принятия решений», «Облачные и Грид-технологии», «Проектирование аппаратных средств вычислительной техники», «Компьютерные сети», «Web – разработка», «Информационная безопасность», «Системное и прикладное программное обеспечение».

Издание рассчитано на ученых, аспирантов и студентов.

***Сборник подготовлен к изданию
кафедрой «Информационные технологии и компьютерные системы»
ФГАОУ ВО «Севастопольский государственный университет».***

Научное электронное издание

© Коллектив авторов, 2018
© СевГУ, 2018

ОГЛАВЛЕНИЕ

СЕКЦИЯ 1. СОВРЕМЕННЫЕ НАПРАВЛЕНИЯ МОДЕЛИРОВАНИЯ В КОМПЬЮТЕРНОЙ ИНЖЕНЕРИИ, ОБРАЗОВАНИИ И НАУКЕ	6
Андриевская А.В., Светличная В.А., Андриевская Н.К. ИСПОЛЬЗОВАНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В КОМПЬЮТЕРИЗИРОВАННОЙ СИСТЕМЕ УПРАВЛЕНИЯ ТРАНСПОРТНЫМИ ПОТОКАМИ В УСЛОВИЯХ БОЛЬШОГО ГОРОДА	7
Бараш В.В., Шевченко В.И. ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ДОРОЖНЫХ КОММУНИКАЦИЙ В ГОРОДЕ СЕВАСТОПОЛЬ НА ПРИМЕРЕ ИМИТАЦИОННОЙ МОДЕЛИ ДОРОЖНОЙ РАЗВЯЗКИ.....	13
Гримута А.В., Шевченко В.И. ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ СИСТЕМЫ ИССЛЕДОВАНИЯ УСПЕВАЕМОСТИ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА UML.....	18
Коваленко И.Ю., Заикина Е.Н. АДАПТИВНЫЕ МЕТОДЫ В PIR-ДАТЧИКАХ ДВИЖЕНИЯ	23
Пашкова Ю.И., Ченгарь О.В. МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ ШКОЛЬНЫХ АВТОБУСОВ ПО ГОРОДСКИМ МАРШРУТАМ В СРЕДЕ ANYLOGIC С ПОМОЩЬЮ ГЕОГРАФИЧЕСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ.....	26
Первушина Е.Ю., Шевченко В.И. ФУНКЦИОНАЛЬНО-СТРУКТУРНАЯ МОДЕЛЬ КОММУНИКАЦИОННЫХ ВЗАИМОДЕЙСТВИЙ В ГБУ РК «МНОГОФУНКЦИОНАЛЬНЫЙ ЦЕНТР ПРЕДОСТАВЛЕНИЯ ГОСУДАРСТВЕННЫХ И МУНИЦИПАЛЬНЫХ УСЛУГ»	34
Рыдзывыло К.Г., Шевченко В.И. КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ СИСТЕМЫ РАЗМЕЩЕНИЯ ФАЙЛОВ В РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СРЕДАХ.....	39
 СЕКЦИЯ 2. РОБОТОТЕХНИКА. ТЕХНОЛОГИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА	 44
Захарченко И.А., Хлебникова В.В. ВОЗМОЖНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА	45
Пахольчук И.С., Букач Б.А. ЗАМЕНИТ ЛИ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ ТОП-МЕНЕДЖЕРОВ	49
 СЕКЦИЯ 3. ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ. ТЕХНОЛОГИИ BIG DATA. СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ	 52
Безпалый И.А., Машенко Е.Н. КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ОБРАБОТКИ ТЕКСТОВЫХ СООБЩЕНИЙ	53
Какаев А.В., Шевченко В.И. ИССЛЕДОВАНИЕ АЛГОРИТМОВ ЭФФЕКТИВНОЙ БАЛАНСИРОВКИ НАГРУЗКИ В РАСПРЕДЕЛЕННОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ.....	58

Ковальчук Н.А., Ченгарь О.В. ПОДХОД К ПРОЕКТИРОВАНИЮ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В ЗАДАЧАХ СОЦИАЛЬНОЙ СЛУЖБЫ	64
Макаров И.В., Ченгарь О.В. ПРИМЕНЕНИЕ ФУНКЦИИ ПРИНАДЛЕЖНОСТИ ДЛЯ СТРУКТУРНОГО РАСПОЗНАВАНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ	68
Панасюк Н.А., Мащенко Е.Н. ИССЛЕДОВАНИЕ ПРОЦЕССОВ АНАЛИЗА ТЕКСТОВЫХ ДАННЫХ	72
Поволоцкий С.С., Заморёнов М.В. СИНЕРГИЯ BIG DATA И BLOCKCHAIN	77
Цокало Ю.А., Лелеков С.Г. ПРОГРАММНАЯ СИСТЕМА АВТОМАТИЧЕСКОГО ВОССТАНОВЛЕНИЯ ГРАММАТИКИ ПО ЦЕПНОМУ КОДУ ОБЪЕКТОВ В СИСТЕМАХ РАСПОЗНАВАНИЯ ОБРАЗОВ	81
 СЕКЦИЯ 4. ОБЛАЧНЫЕ И ГРИД-ТЕХНОЛОГИИ	 88
Белоусова А.Э., Инюшин В.И., Потанина М.В. ПЕРСПЕКТИВЫ ИСПОЛЬЗОВАНИЯ SEASAAS-ТЕХНОЛОГИЙ В РОССИИ.....	89
Капустина Д.Д., Потанина М.В. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ОБЛАЧНЫХ ТЕХНОЛОГИЙ В ОБРАЗОВАНИИ.....	93
Лисовский М.С., Воронин Д.Ю. ОРГАНИЗАЦИЯ УПРАВЛЕНИЯ АКТОРНЫМ РАВНОВЕСИЕМ В ОБЛАЧНЫХ СЕРВИСАХ КРИТИЧЕСКОГО ПРИМЕНЕНИЯ	96
 СЕКЦИЯ 5. ПРОЕКТИРОВАНИЕ АППАРАТНЫХ СРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ	 100
Галасюк А.А., Явкун Ю.Л. ПРОГРАММНОЕ И АППАРАТНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ АВТОМАТИЗАЦИИ РАБОТЫ ИНФУЗИОННОЙ СИСТЕМЫ ПОДАЧИ ПРЕПАРАТА (КАПЕЛЬНИЦЫ)	101
Шамков А.С., Чернега В.С. ОСОБЕННОСТИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ СИСТЕМЫ КОНТРОЛЯ МЕДИЦИНСКИХ КАПЕЛЬНИЦ	105
 СЕКЦИЯ 6. КОМПЬЮТЕРНЫЕ СЕТИ	 108
Евсеева В.В., Мащенко Е.Н. ИССЛЕДОВАНИЕ КРИТЕРИЕВ ЭФФЕКТИВНОСТИ ФУНКЦИОНИРОВАНИЯ БЕСПРОВОДНЫХ СЕНСОРНЫХ СЕТЕЙ.....	109
Соколова Е.Ю., Потанина М.В. РОЛЬ КОМПЬЮТЕРНЫХ СЕТЕЙ И ИНТЕРНЕТА В РАЗВИТИИ БИЗНЕСА	114
 СЕКЦИЯ 7. WEB – РАЗРАБОТКА	 118
Дереглазов Н.А., Заморёнов М.В. ПРОГРАММНАЯ СИСТЕМА, РЕАЛИЗОВАННАЯ В ВИДЕ ВЕБ-ПРИЛОЖЕНИЯ НА МИКРОСЕРВИСНОЙ АРХИТЕКТУРЕ	119

Захаров Д.А., Заикина Е.Н. РЕАЛИЗАЦИЯ ПЛАТФОРМЫ ДЛЯ ПРОВЕДЕНИЯ ВЫБОРОВ ПРИ ИСПОЛЬЗОВАНИИ ТЕХНОЛОГИИ БЛОКЧЕЙН	124
Мачавариани Д.А., Строганов В.А. WEB-СИСТЕМА “ПАНЕЛЬ УПРАВЛЕНИЯ QIWI КОШЕЛЬКАМИ”	127
 СЕКЦИЯ 8. ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ	 129
Акимов А.А., Белоног А.А., Лебедева М.А. ВИРТУАЛЬНАЯ ЧАСТНАЯ СЕТЬ НА БАЗЕ БРАНДМАУЭРА	130
Киричук П.А., Корепанова Н.Л. ПРАВИЛА ПАРОЛЬНОЙ ЗАЩИТЫ В СОВРЕМЕННЫХ СЕТЕВЫХ СТРУКТУРАХ	134
Петров С.В., Пичугова Т.Б., Загоренов М.В. ПРОГРАММНЫЙ КОМПЛЕКС ПРЕДОСТАВЛЕНИЯ УСЛУГ АВТОРИЗАЦИИ И АУТЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ	137
Фатеева Н.А., Потанина М.В. ОСНОВНЫЕ АСПЕКТЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В МЕДИЦИНСКИХ УЧРЕЖДЕНИЯХ РОССИИ	140
Хасанов Ш.А., Агишев Т.Х. БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ СИСТЕМ ПЕРСОНАЛЬНЫХ ДАННЫХ.....	144
 СЕКЦИЯ 9. СИСТЕМНОЕ И ПРИКЛАДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	 148
Дзюба С.Н., Фисун С.Н. ИССЛЕДОВАНИЕ МЕТОДОВ АВТОМАТИЗАЦИИ ПОСТРОЕНИЯ НИСХОДЯЩИХ СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ	149
Жадан Г.Б., Левченко Г.А., Супрунов В.С., Дымченко И.В., Сырых О.А., Кузнецов С.А. МОБИЛЬНОЕ ПРИЛОЖЕНИЕ «МОБИЛЬНЫЙ СЕВГУ»	153
Омельченко Д.Ю., Фисун С.Н. ПРОГРАММНАЯ СИСТЕМА ДЛЯ ПРЕОБРАЗОВАНИЯ ГРАММАТИК В НИСХОДЯЩИХ СИНТАКСИЧЕСКИХ АНАЛИЗАТОРАХ	159
Петраков В.А., Строганов В.А. ИСПОЛЬЗОВАНИЕ МЕТОДА ДОПОЛНЕННОЙ РЕАЛЬНОСТИ В КОНТЕКСТЕ ПОВЫШЕНИЯ ПРИВЛЕКАТЕЛЬНОСТИ ГОРОДА БАЛАКЛАВА	163
Тукманов Г.А. СЕМАНТИЧЕСКАЯ МОДЕЛЬ ГЕНЕРАЦИИ SQL-ЗАПРОСОВ.....	166
Шевченко Д.Д., Дубенецкий В.А. ПРОГРАММНАЯ СИСТЕМА ДЛЯ ПОИСКА НЕЧЕТКИХ ДУБЛИКАТОВ В ТЕКСТОВЫХ ДОКУМЕНТАХ	170
Шлемко А.Г., Фисун С.Н. ИССЛЕДОВАНИЕ МЕТОДОВ АВТОМАТИЗАЦИИ ПОСТРОЕНИЯ ЛЕКСИЧЕСКИХ И СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ	177

СЕКЦИЯ 1

**Современные направления
моделирования в компьютерной
инженерии, образовании и науке**



УДК 681.3

А.В. Андриевская, студентка 2-го курса

В.А. Светличная, к.т.н., доцент

Н.К. Андриевская, старший преподаватель

Донецкий национальный технический университет

ИСПОЛЬЗОВАНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В КОМПЬЮТЕРИЗИРОВАННОЙ СИСТЕМЕ УПРАВЛЕНИЯ ТРАНСПОРТНЫМИ ПОТОКАМИ В УСЛОВИЯХ БОЛЬШОГО ГОРОДА

Аннотация: В статье рассмотрены вопросы использования имитационного моделирования при регулировании транспортных потоков в крупных городах и динамически изменяющихся условиях. Приведены способы определения рациональных параметров работы светофора и регулирования скорости движения.

Ключевые слова: транспортный поток, интенсивность, скорость, светофор, зеленая волна, имитационное моделирование.

Annotation: The article discusses the use of simulation modeling in regulating transport flows in large cities and dynamically changing conditions. The ways of determining the correct parameters of traffic lights and speed control.

Key words: Traffic flow, intensity, speed, traffic light, green wave, simulation of simulation.

Общая постановка проблемы

Современное состояние автомобильных перевозок характеризуется все возрастающими объемами, увеличением скоростей и интенсивности движения. Связанный с этим рост числа дорожно-транспортных происшествий предъявляет новые, все более высокие требования к техническому совершенству автомобильных дорог, их инженерному оборудованию и состоянию, транспортно-эксплуатационным характеристикам и организации движения в процессе эксплуатации.

Одну из основных ролей в решении этих задач играет создание компьютеризированных систем. Разработка таких систем остается актуальной на протяжении многих последних лет. С одной стороны, это объясняется резким увеличением интенсивности движения автомобилей, а с другой стороны все возрастающими техническими возможностями средств регистрации, слежения и управления движением транспорта и, особенно, возрастающими возможностями компьютерных технологий.

К настоящему времени разработан целый ряд автоматизированных систем мониторинга и управления транспортными потоками, использующих различные математические методы и принципы, и по совокупности решающих те или иные задачи. К ним можно отнести такие системы как ИТС (интеллектуальная транспортная система), АСУДД (автоматизированная система управления дорожным движением), АРДАМ (автоматизированное регулирование движения на автомобильных магистралях) [1]. После проведения анализа возможностей существующих подсистем можно сказать, что они не позволяют в полной мере улучшить динамику движения транспортных потоков.

Проведенный анализ схемы управления дорожным движением позволил определить существующие способы управления трафиком. Анализ же технических средств мониторинга и регулирования транспортными потоками, позволил выделить основные параметры, которые можно получить о движении транспортных средств. Следовательно, во время каких-либо критических ситуациях изменяя эти параметры можно добиться оптимизации дорожного движения на определенном участке дороги.

Основную задачу автоматизированной системы управления транспортными потоками можно сформулировать следующим образом: при изменяющихся характеристиках транспортного потока с учетом аварийных ситуаций, образования пробок, состояния метеорологических условий в каждый момент времени обеспечивать требуемые показатели функционирования дорожного движения, к которым следует отнести скорость, пропускную способность и безопасность движения. Результатами решения задачи должны быть измененные длительности сигналов светофоров (длина цикла, длительность зеленого и красного света) и рекомендуемая скорость движения на заданном участке. [2]

Решение задачи

Для поиска эффективных стратегий управления транспортными потоками в большом городе и организации дорожного движения необходимо учитывать широкий спектр характеристик транспортного потока, закономерности влияния внешних и внутренних факторов на динамические характеристики транспортного потока[3,4].

В крупных городах, страдающих от «пробок», светофорные объекты подключают к единой системе регулирования движения. На определенных участках задается длительность сигналов светофора таким образом, чтобы обеспечить «зеленую волну». «Зеленая волна» - методика обеспечения безостановочного движения транспорта за счёт согласованного включения зелёного света светофора на перекрёстках.

С целью достижения эффективного управления транспортными потоками предлагается создание “зеленых волн” на магистралях, изменяемых во времени, путем имитационного моделирования поведения транспортных потоков, реализация которого требует создания адекватной модели транспортного потока с учетом сложности конфигурации транспортных развязок, внешних воздействий, и случайных факторов, влияющих на всю структуру взаимодействия движущихся потоков.

Поводом для начала работы системы служит выход некоторых параметров транспортных потоков за определенные пределы. Возникает вопрос, какие из параметров, с помощью которых описан перекресток, являются более весомыми и определяют неприятности, связанные с длинной очереди из машин, которая образуется возле светофора. Очевидно, что вторым параметром, от которого зависит длина очереди является интенсивность и он будет определяющим фактором моделирования.

Для разработки модели были проанализированы разные виды перекрестков и выделены их параметры:

- длительность красного сигнала светофора;
- длительность зеленого сигнала светофора;
- интенсивность транспортного потока;
- длина очереди на перекрестке.

В зависимости от интенсивности транспортного потока и длительности красного и зеленого сигналов светофора будет различной длина очереди на перекрестке.

Таким образом, искомую функцию можно выразить следующим образом:

$$K_{\text{длины}} = f(T_c, t_{\text{кр}}, t_{\text{зел}}, \lambda), \quad \text{где}$$

$K_{\text{длины}}$ – длина очереди на перекрестке,

T_c – длина цикла светофора,

$t_{\text{кр}}$ – длительность красного сигнала светофора,

$t_{\text{зел}}$ – длительность зеленого сигнала светофора,

λ – интенсивность транспортного потока.

Алгоритм, моделирующий работу регулируемого перекрестка приведен на рис.1 и функционирует следующим образом. Длина цикла длительности светофора варьируется в пределах 50 – 120 секунд: ($50 \leq T_c \leq 120$). Длительности зеленого и красного сигналов варьируются с шагом в 5 секунд. Дороги могут быть с однополосным движением и многополосным, и общая интенсивность разделится на фактическое количество полос на дороге.

Случайный поток автомобилей на ограниченном промежутке времени был принят стационарным и интенсивность его моделировалась с помощью экспоненциального закона. Далее рассматривались ситуация включенного зеленого сигнала светофора и проверка наличия или отсутствия очереди перед светофором. Если очередь пустая и горит зеленый светофор, то автомобиль проезжает перекресток. Время проезда моделировалось как случайная величина с нормальным законом распределения. Если есть очередь и горит зеленый светофор, то автомобиль становится в очередь. С «головы» очереди автомобили последовательно покидают перекресток и очередь уменьшается. Если горит красный сигнал

светофора, то автомобиль становится в очередь и ожидает начало движения автомобилей после включения зеленого сигнала светофора. Каждый раз производится подсчет автомобилей в очереди, не успевших проехать на зеленый сигнал светофора.

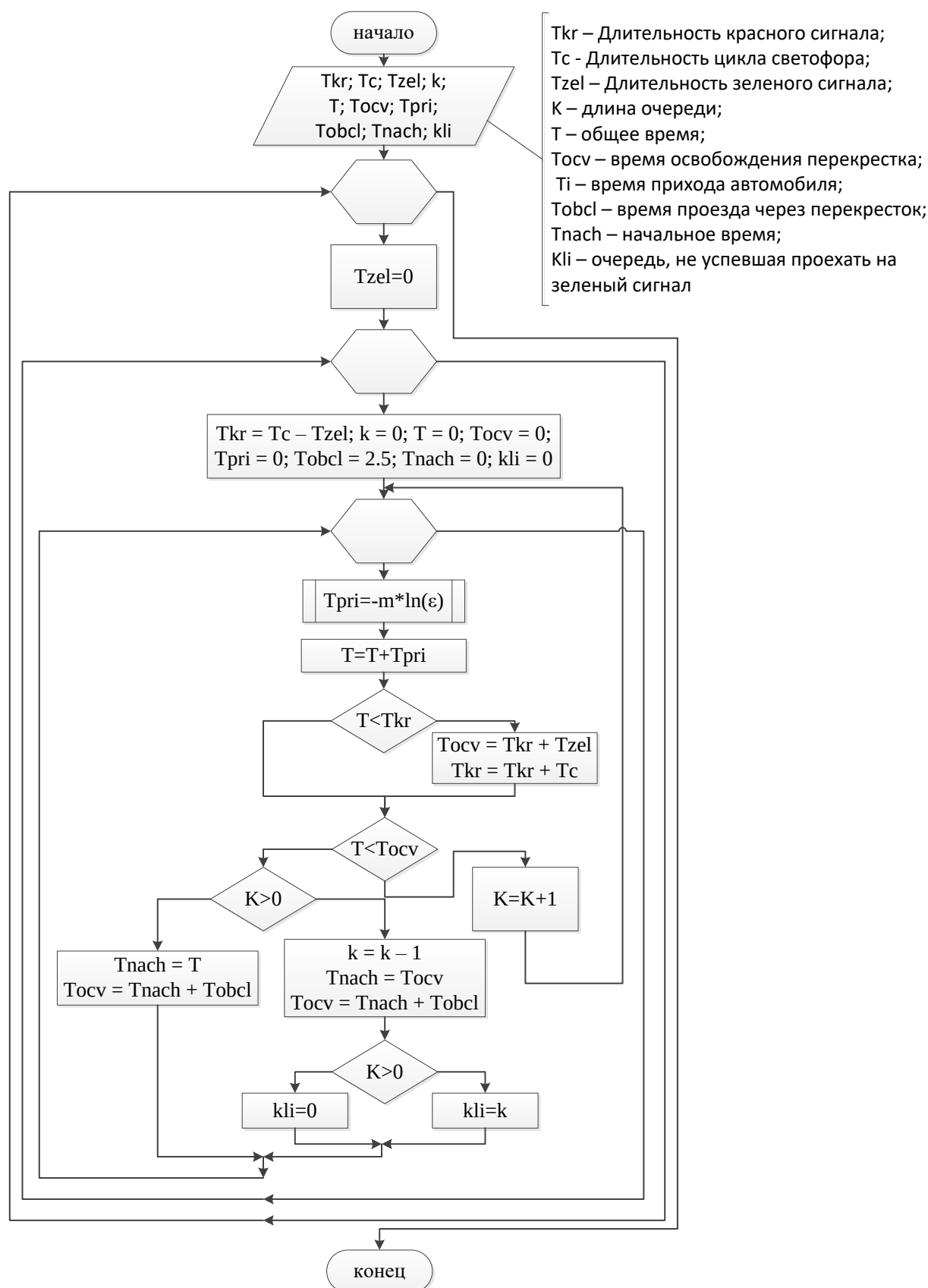


Рис. 1. Блок-схема алгоритма, моделирующего работу перекрестка

Каждый из этих объектов обладает своими специфическими свойствами. Например, объект «Улица» можно охарактеризовать такими свойствами: название улицы, длина улицы, количество перекрестков, расстояние между перекрестками.

Перекресток имеет следующие параметры: количество направлений движения, количество полос движения, количество светофоров, длина очереди.

Параметрами светофора будут: длительность цикла светофора, длительность зеленого сигнала светофора, длительность красного сигнала светофора.

Свойства транспортного потока: интенсивность транспортного потока, скорость движения транспортного потока.

Для взаимосвязи данных объектов предметной области и реализации основных управляющих алгоритмов требуется определить классы-обработчики, как показано в реализации объектных моделей компьютерных систем[5-7].

Главная улица должна обладать свойством «зеленая волна». Благодаря этому, транспортный поток будет с заданной средней скоростью преодолевать перекрестки. Поток будет попадать на зеленый сигнал светофора и не создавать огромную очередь на перекрестке. Возникает необходимость для создания еще одного класса обработчика объектов. Данный обработчик должен содержать алгоритм реализации зеленой волны для главной дороги. Выходными параметрами должны быть длительности сигналов светофоров и средняя скорость движения автомобилей. Обработчик должен использовать свойства объектов «Улица», «Перекресток» и «Транспортный поток».

Реализуя «зеленую волну» на главной дороге, а также имея возможность имитационного моделирования движения транспортных потоков во всех направлениях на перекрестках, требуется решить проблему уменьшения очереди на перекрестке во время появления непредвиденных ситуаций.

Алгоритм решения этой задачи должен включать в себя имитационное моделирование движения транспортных потоков на каждом объекте «Перекресток» таким образом, чтобы, изменяя длительность сигнала светофора на проблемном участке, автоматически производился пересчет «зеленой волны», а также сохранялись приемлемые значения в каждом направлении всех перекрестков. В итоге становится возможным выделить еще один класс обработчик в объектно-ориентированной модели, который будет включать в себя реализацию данного алгоритма.

Выводы и направления дальнейших исследований

В результате проведенных исследований была разработана имитационная модель, формализующая функционирование перекрестков и позволяющая оценивать параметры их работы в различных условиях при изменении внешних факторов. Приведены результаты работы модели в условиях реально существующей улицы с несколькими перекрестками.

Дальнейшие исследования следует вести в направлении использования технологии ООП и в полной мере решения задачи разработки объектно-ориентированную модели компьютеризированной системы управления транспортными потоками в условиях большого города.

Библиографический список

1. Интеллектуальные транспортные системы — проблемы на пути внедрения в России/ Интернет-ресурс. - Режим доступа: <http://habrahabr.ru/post/175497>
2. Барышев В.В., Светличная В.А. Функциональная структура компьютеризированной системы управления транспортными потоками в условиях большого города // Информационные управляющие системы и компьютерный мониторинг. - Донецк: ДонНТУ, 2014. - С. 33-38.

3. Руководство по оценке пропускной способности автомобильных дорог. – М.: Транспорт, 1982 г. – 95с.
4. Левашев А.Г., Михайлов А.Ю., Головных И.М. Проектирование регулируемых пересечений: Учеб. пособие. – Иркутск: Изд-во ИрГТУ, 2007 г. – 208с.
5. Дорош А.И., Светличная В.А., Андриевская Н.К. Функциональная структура подсистемы поддержки принятия решения для задач финансовой реструктуризации предприятия.
Информатика и компьютерные технологии: в 2-х томах / А.И. Дорош, В.А. Светличная, Н.К. Андриевская // Сборник трудов VIII международной научно-технической конференции студентов, аспирантов и молодых ученых -18-19 сентября 2012 г., Донецк, ДонНТУ. -2012.
6. Землянская С.Ю., Сложеницын А.В., Андриевская Н.К. Структура модели для прогнозирования поведения групп лиц. ИНФОРМАТИКА И КИБЕРНЕТИКА. – Д.: ДонНТУ, – 2016. – № 4(6).
7. Светличная В.А., Андриевская Н.К., Чаленко К.Ю. Разработка функциональной структуры логистической системы формирования заказов для интернета-магазина. ИНФОРМАТИКА И КИБЕРНЕТИКА. – Д.: ДонНТУ, – 2017. – № 3(9). –111-118 с.

УДК 004.942

В.В. Бараш, магистрант**В.И. Шевченко, к.т.н, доцент***Федеральное государственное автономное образовательное учреждение высшего образования «Севастопольский государственный университет»***ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ДОРОЖНЫХ КОММУНИКАЦИЙ В ГОРОДЕ СЕВАСТОПОЛЬ НА ПРИМЕРЕ ИМИТАЦИОННОЙ МОДЕЛИ ДОРОЖНОЙ РАЗВЯЗКИ**

Аннотация: в данной статье разработана и представлена имитационная модель дорожной развязки на улице Меньшикова, города Севастополя. Представленная модель используется с целью сбора статистических данных для исследования эффективности эксплуатации дорожных коммуникаций. При разработке имитационной модели использован мультиагентный подход, в модели выделено два класса транспортных агентов: легковые автомобили, общественный транспорт (городские автобусы и микроавтобусы). В модели собирается статистика «времени жизни» каждого из агентов, среднему числу агентов, среднему времени непроизводительного ожидания.

Ключевые слова: имитационная модель, AnyLogic, моделирование, дорожные коммуникации.

Annotation: in this article, an imitation model of the road intersection on Menshikov Street, the city of Sevastopol, was developed and presented. The presented model is used to collect statistical data for the study of the efficiency of the operation of road communications. When developing the imitation model, a multi-agent approach was used, two transport agent classes were identified in the model: cars, public transport (city buses and minibuses). The model collects the "life time" statistics of each agent, the average number of agents, the average time of unproductive waiting.

Key words: simulation model, AnyLogic, modeling, road communications.

В настоящее время в Севастополе, как и во многих других городах, весьма остро стоит проблема транспортных заторов. При этом исследования по регионам показывают, что потенциал существующих улично-дорожных сетей (УДС) используется далеко не полностью. Повышение пропускной способности УДС может быть достигнуто за счет внедрения автоматизированных систем мониторинга и управления дорожным движением [1]. Эффективная работа такой системы невозможна без использования средств моделирования, позволяющих провести прогнозное исследование того или иного участка УДС на предмет загруженности, в зависимости от изменений транспортных потоков или структурных изменений транспортного участка.

Особенности транспортных систем делают невозможным построение адекватной аналитической модели, позволяющей исследовать варианты управления в этой системе и ее характеристики в различных условиях. В то же время имитационная модель как метод исследования подобных объектов представляется обещающим подходом к решению этой проблемы: оно позволяет быстро и с хорошей точностью прогнозировать характеристики сложных систем подобной природы и оптимизировать существенные параметры, выбирая соответствующие параметры оптимизации.

Главной целью данного исследования является создание максимально реалистичной модели транспортной сети, что позволит решать широкий спектр задач по оптимизации дорожного движения, возможности расширения дорожной сети и поиску её проблемных участков. В статье приведен фрагмент такой модели УДС - модель дорожной развязки на улице Меньшикова.

Для моделирования транспортных систем можно выдвинуть ряд требований, учитывающих специфику УДС и транспортных потоков [2]:

- Возможность загрузки карты с изображением дорог и перекрестков через файл, подготовленный в графическом редакторе.
- Задание входного потока автомобилей в любом месте карты (имитация мест рождения транспортного потока: магазины, места работы и учебы и т.д.).
- Моделирование проезда автомобилей через регулируемый перекресток. На входе задаются: интенсивность движения, границы красного и зеленого сигналов светофора. В модели рассматриваются все варианты сигналов светофора и для каждого варианта выполняет прогон автомобилей.

- Получение объединенных результатов для моделируемого участка УДС в целом. Если задать четыре интенсивности (как на перекрестке), то программа вычислит объединенные результаты (средневзвешенные значения параметров: пропускная способность, средняя длина очереди, среднее время задержки).
- Ускорение вычислений (виртуализация времени). Имитационная программная модель позволяет смоделировать десятичасовой прогон автомобилей за 30-60 секунд реального времени.
- Нахождение оптимальных режимов светофорного регулирования для одного перекрестка. Можно указать ограничения на ряд выходных параметров, модель отобразит только те результаты, которые удовлетворяют заданным условиям.
- Сохранение базы знаний результатов модельных экспериментов для последующего анализа, например, в среде MS Excel [3].

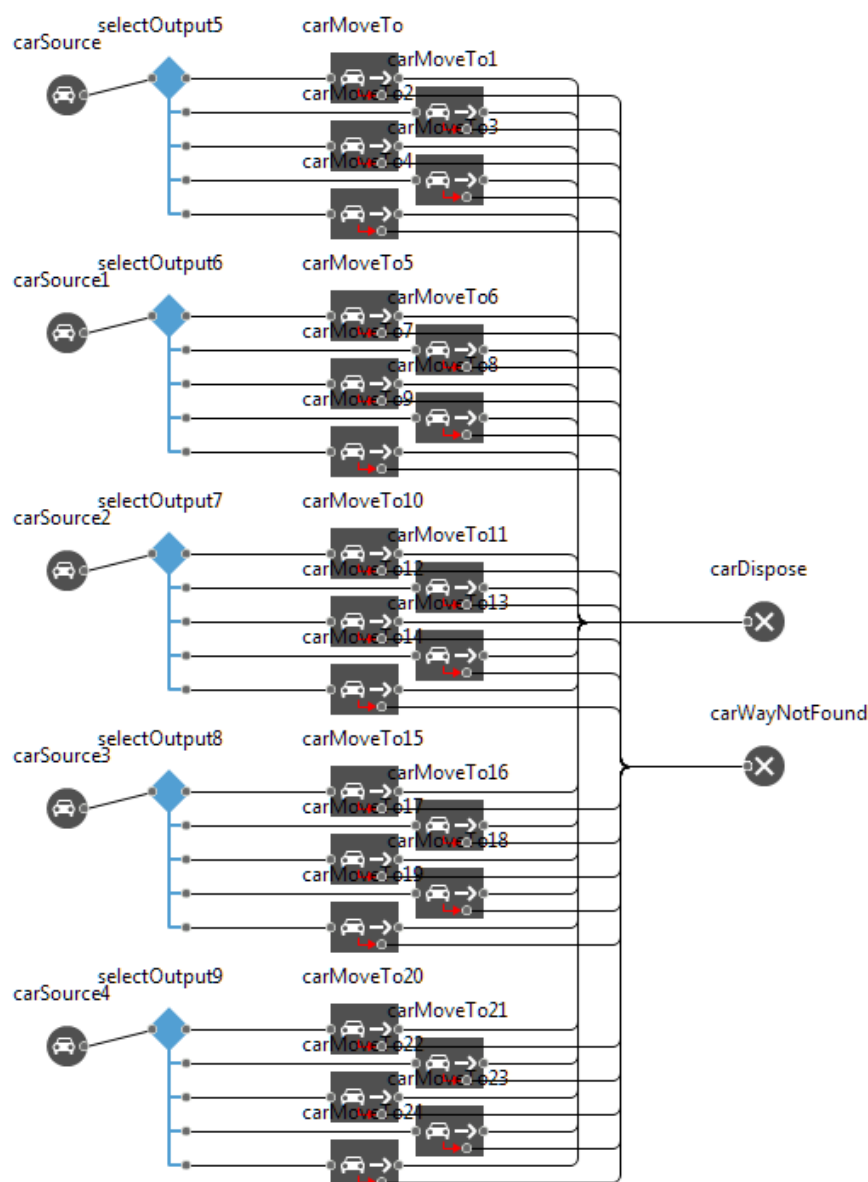


Рис. 1. Алгоритм движения легкового транспорта

Предлагаемая имитационная модель транспортного узла условно разделена на следующие модуля:

- модуль управления агентом;
- модуль управления светофорами;
- модуль сбора статистических данных.

Далее последовательно рассмотрим каждый из модулей. Модуль управления агентом включает в себя алгоритм, которому будет следовать агент (транспорт). Для каждого агента задается свой собственный алгоритм действий (рисунок 1), на текущем этапе модель отрабатывает обработку данных от трех видов транспортных агентов: легковые автомобили, городские автобусы среднего класса и микроавтобусы. Так же в системе использован отдельный вид агентов, отвечающих за светофорное регулирование на перекрестках – светофорный объект.

Обобщенно, алгоритм движения легкового транспорта можно разбить на ряд следующих базовых этапов:

1. В заданной точке дорожного полотна создается агент – транспортное средство.
2. С установленной целью, принимающим решение (ЛПР) вероятностью, (в иллюстрациях на примере 0.2) выбирается конечная точка маршрута агента.
3. Агент движется до ближайшего перекрестка;
4. Если перекресток:
 - а) не регулируемый – агент пересекает перекресток и движется к следующему перекрестку;
 - б) регулируемый – агент продолжает движение и пересекает перекресток только если на светофоре горит зеленый свет;
5. По достижению конечной точки маршрута агент уничтожается.

Также в случае если агент не может найти путь к конечной точке он удаляется из модели во избежание возникновения ошибок.

В данной модели реализованы также агенты имитирующие маршрутный транспорт (городские автобусы среднего класса и микроавтобусы), рассмотрим их работу далее. Схема алгоритма обслуживания транспортных агентов приведена на рисунке 2.

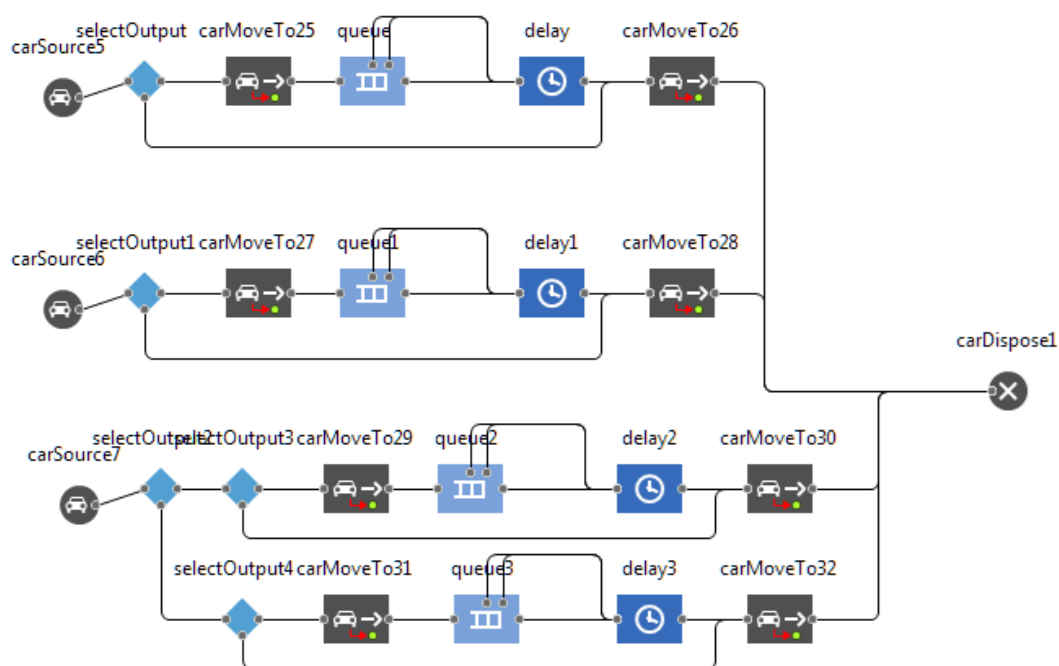


Рис. 2. Алгоритм движения маршрутного транспорта

Обобщенно, алгоритм движения маршрутного транспорта можно разбить на ряд следующих базовых этапов:

1. В заданной точке дорожного полотна создается агент, имитирующий работу одной единицы маршрутного транспорта.
2. С определенной вероятностью агент движется к элементу «автостанция» (переход к п.3), либо в конечную точку маршрута, тем самым имитируя переполненный транспорт (переход к п.6).

3. При достижении элемента «автобусная остановка» агент попадает в очередь, значение которой зависит от вместимости конкретной «автобусной остановки». Вместимость автобусной остановки задаётся ЛПР перед началом эксперимента.

4. Агент останавливается на «автобусной остановке» на время, заданное блоком «delay», имитирующим процесс ожидания входа/выхода пассажиров на остановке.

5. По истечению времени задержки агент продолжает движение до конечной точки маршрута.

6. По достижению конечной точки маршрута агент уничтожается.

Блок «delay» (пример блока можно видеть на рисунке 2) позволяет управлять временем задержки агента в модели, что позволяет имитировать реальное время задержки маршрутного транспорта, затрачиваемое на погрузку и выгрузку пассажиров.

Модуль управления светофорами несет в себе информацию обо всех регулируемых перекрестках, регулируемых пешеходных переходах, и управления ими. Для каждого отдельного регулируемого перекрестка и пешеходного перехода существует свой элемент типа «trafficLight» (рисунок 3).

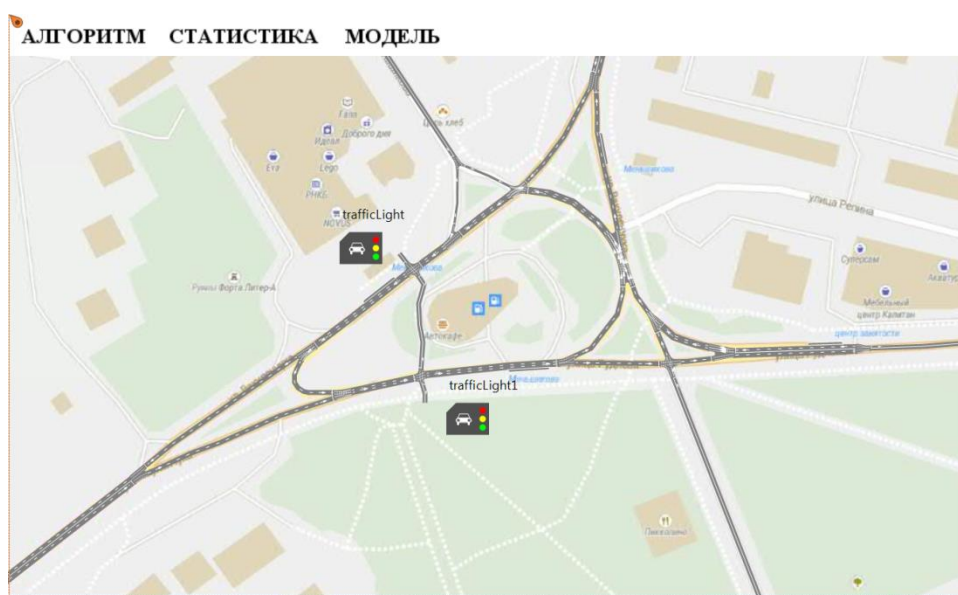


Рис. 3 Дорожная развязка на улице Меншикова

Каждому регулируемому перекрестку и пешеходному переходу назначен отдельный управляющий элемент «trafficLight», который управляет временем задержки на каждом отдельном перекрестке.

Можно выделить основные входные и выходные параметры для модели. Обработка информации в модели представляет собой предсказание параметров транспортных потоков, на основе следующих входных данных: текущих интенсивностей транспортных потоков; скоростей движения транспортных средств (по типам); расстояний между смежными управляемыми перекрестками в системе; предсказания маршрутов движения транспортных средств на основе статистики для текущего дня недели и времени суток; текущих длин фаз светофорных объектов на перекрестках УДС.

При моделировании движения автомобилей на участке УДС могут быть зафиксированы следующие выходные параметры:

- общее количество обслуженных автомобилей. Учитываются автомобили, которые пересекли участок УДС (перекресток) за все время проведения вычислительного эксперимента;

- общее количество циклов светофорного регулирования. Количество циклов светофорного регулирования за все время проведения вычислительного эксперимента;

- пропускная способность участка УДС (перекрестка), авто/мин. Среднее количество автомашин, проезжающих через перекресток за одну минуту;
- среднее время жизни автомобиля, чч:мм:сс. Среднее время полного прохождения автомобилем дорожного участка с момента генерации и до пересечения перекрестка;
- среднее время задержки автомобиля, с. Сколько времени в среднем автомобиле не двигаются (скорость равна нулю) на данном дорожном участке;
- общее количество автомобилей в системе, авто/мин. Среднее количество автомобилей в системе на момент включения зеленого сигнала, с любой скоростью;
- длина очереди, авто/мин. Среднее количество автомобилей в системе на момент включения зеленого сигнала, у которых скорость равна нулю.

Модуль сбора статистических данных соответственно сохраняет результаты моделирования и осуществляет их вывод в виде графиков. На рисунке 4 приведен фрагмент выходных данных, получаемых при моделировании транспортного узла. Первый график отображает количество агентов в модели в каждый момент времени, второй – среднее время остановок агентов за все время их существования в модели, третий – среднюю скорость движения агентов за все время их существования в модели, четвертый – среднее время пребывания агентов в модели.

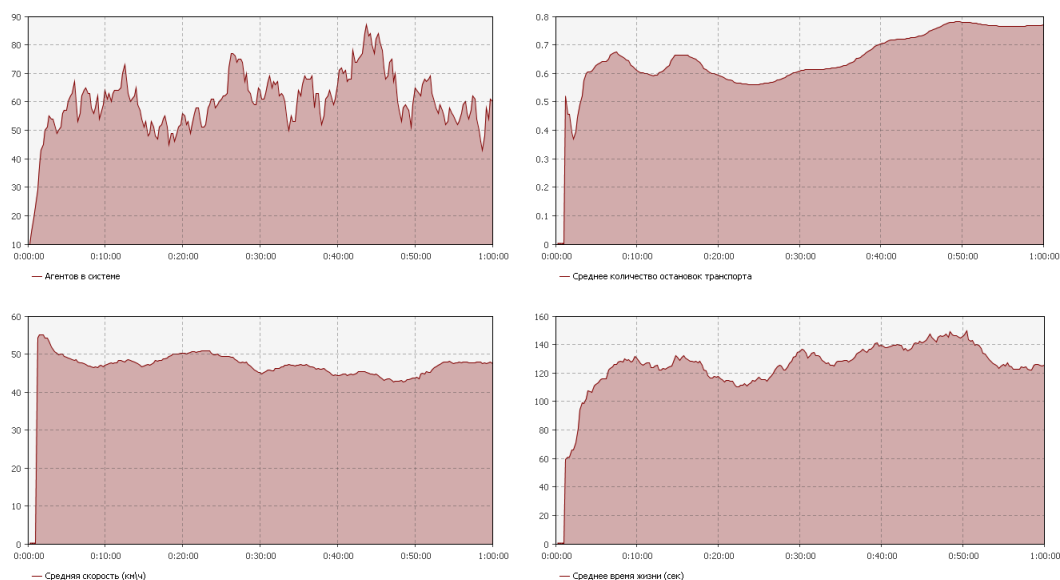


Рис. 4. Графики отображения статистических данных

Представленная имитационная модель транспортного узла может являться базой для разработки комплекса имитационных моделей в среде многоподходового имитационного моделирования AnyLogic.

Библиографический список:

1. Бродский Г. С., Айвазов А. Р. Автоматизированное управление дорожным движением в городской среде // Мир дорог. 2007. № 26. С. 2–3.
2. Ахмадинуров М. М., Тимофеева Г. А. Модели массового обслуживания в задаче оптимизации работы светофора // Вестник СГТУ. 2011. №1 (57). URL: <https://cyberleninka.ru/article/n/modeli-massovogo-obsluzhivaniya-v-zadache-optimizatsii-raboty-svetofora> (дата обращения: 21.03.2018).
3. Захаров Ю.И., Карнаух Е.С. Основные современные инструменты имитационного моделирования транспортных потоков // Вісник Придніпровської державної академії будівництва та архітектури, № 1 (190), 2014, с. 46-51.

УДК 681.3.053

А.В. Гримута, студент 1-го курса

В.И. Шевченко, канд. техн. наук, доцент

Севастопольский государственный университет

ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ СИСТЕМЫ ИССЛЕДОВАНИЯ УСПЕВАЕМОСТИ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА UML

Аннотация: В статье представлена модель системы исследования успеваемости студентов, реализуемой с целью оптимизации метрик качества учебного процесса. В модели рассматриваются альтернативные образовательные траектории, включающие контактную аудиторную работу, работу в среде дистанционного обучения и работу с применением смешанных технологий (blended learning). Разрабатываемая модель позволит выявить эффективность использования дистанционного обучения в высших учебных заведениях. После построения концептуальной модели проектируемой системы было сформулировано описание предметной области, а также ее диаграммы классов, взаимодействия, состояний и видов деятельности в виде UML-диаграмм. Результаты формализации модели с использованием языка UML представлены в данной работе и необходимы для документирования проектируемой системы.

Ключевые слова: Визуальное моделирование, концептуальная модель системы, UML-диаграммы.

Annotation: The article is about the model of the system for studying the progress of students, implemented to optimize the quality metrics of the learning process. The model considers alternative educational trajectories, including contact auditor work, work in the environment of distance learning and work with the use of mixed technologies (mixed training). The developed model allows to reveal the effectiveness of using distance learning in higher education institutions. After constructing the conceptual model of the projected system, a description of the domain was formulated, as well as its diagrams of classes, interactions, states and activities in the form of UML diagrams. The results of formalizing the model using the UML language are presented in this paper and are necessary for documenting the projected system.

Key words: Visual modeling, the conceptual model of the system, UML diagrams».

Введение

Внедрение информационных технологий в различных сферах общества позволяет решить проблемы связанные с быстро возрастающими интеграционными процессами, проникающими во все сферы человеческой деятельности: науку, культуру, образование, производство, управление и т. д.. При использовании информационных технологий в образовательных процессах высших учебных заведений можно обеспечить академическую, виртуальную мобильность студентов, позволять вырабатывать персональные образовательные траектории развития, тем самым повышая эффективность обучения студентов.

Технологии дистанционного образования активно исследуются на уровне моделей и классификаторов моделей, этим вопросом посвящен ряд исследований [1-6]. Согласно классификации R.S.J. Tuninga и I.B.J.Seinen [5] можно выделить три вида моделей: 1) консультационная модель, в основе которой лежит посещение обучающими консультационных центров; 2) модель корреспонденции, отличием которой является отсутствие личного взаимодействия с преподавателями; 3) модель регулируемого самообучения, которая подразумевает полностью самостоятельное обучение.

В ходе данного исследования планируется выполнить визуальное моделирование консультационной модели [2], которая объединяет традиционную очную форму образования с дистанционной. На текущем этапе, это идеализированная модель процесса, которая по мере адаптации к реальным бизнес-процессам ВУЗа будет модифицироваться.

Визуализация модели

Применение языка UML при выполнении визуального моделирования можно охарактеризовать как разработку от абстрактного описания к конкретной реализации для соответствующей программной системы. На первом этапе визуализации необходимо построить диаграмму вариантов использования (Use case).

В работе рассматривается циклическая модель, согласно которой студент очной формы обучения автоматически переходит на очно-дистанционную форму в случае, если показывает недостаточный уровень компетенций по результатам промежуточного

оценивания и возвращается на очную форму по улучшению соответствующих показателей. Планируемый процесс обучения регламентируется длительностью обучения и количеством занятий. По результатам обучения студенты получают диплом (сертификат) об успешном прохождении курса очной или дистанционной формы обучения. На основании описания этой модели обучения на естественном языке была построена диаграмма вариантов использования (Рисунок 1).

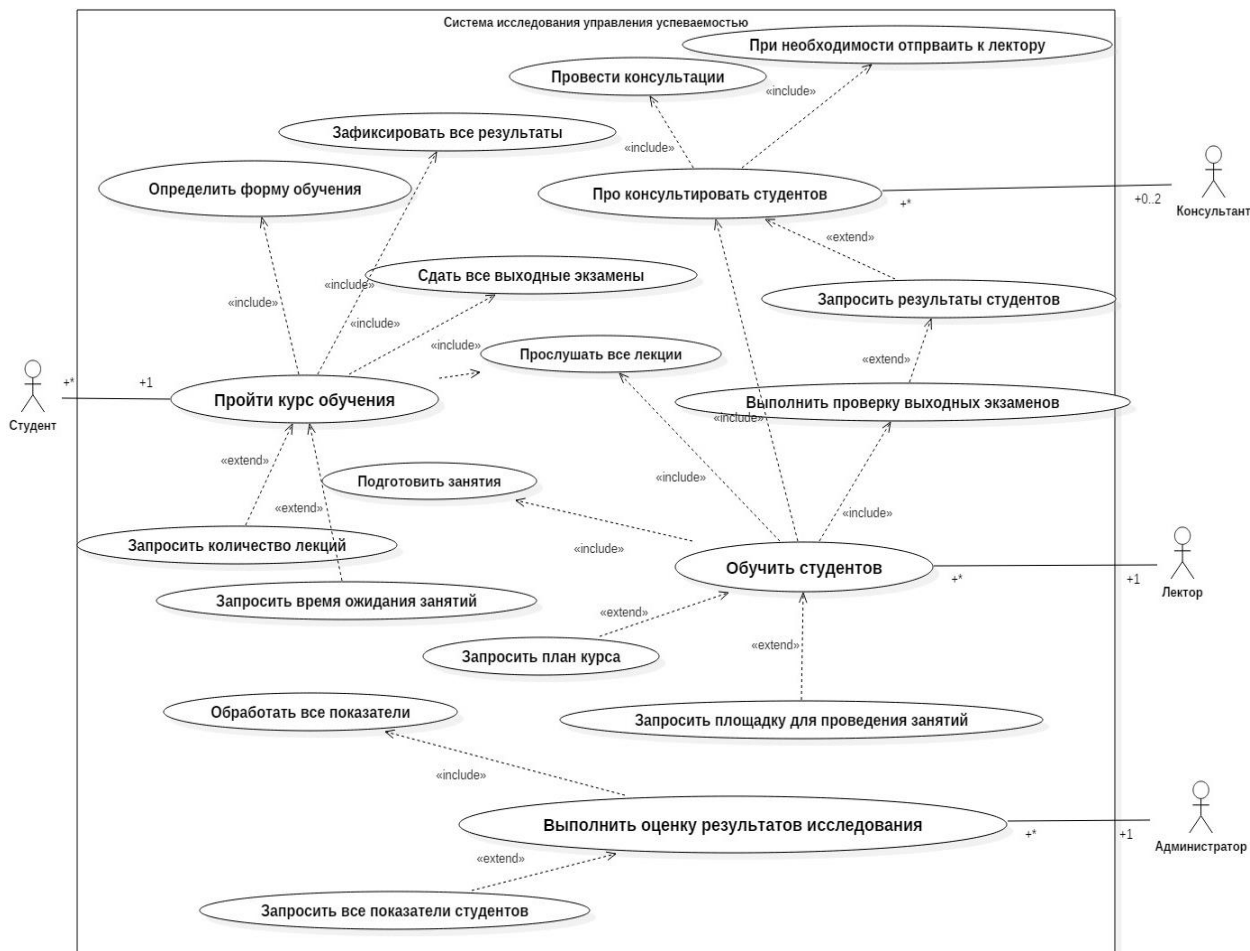


Рис. 1. Диаграмма вариантов использования системы исследования управления успеваемостью

На данной диаграмме можно видеть таких акторов как: студент, консультант, лектор (преподаватель), администратор.

Основной задачей студента является прохождение курса обучения, для этого он уточняет количество лекций по курсу, а также время ожидания занятий. При этом прохождение курса включает в себя определение формы обучения, прослушивание всех лекций, успешную сдачу выходного контроля.

Задача консультанта коррелируется с задачей преподавателя, к ней относится проведение консультаций. Для проведения консультации необходимо выполнить запрос результатов по выходному экзамену.

Спектр задач преподавателя является более широким, но при этом основной задачей преподавателя считается обучение студентов. Для этой задачи преподавателю необходимо получить план курса и запросить площадку для проведения занятий. Блок обучения состоит из таких элементов, как: подготовка занятия, проведение занятия, проведение выходного контроля, а также проведение консультации со студентами, не освоившими материал лекции.

Важным актором системы является администратор системы, задачей которого является выполнение оценки результатов исследования, для этого он должен выполнить запрос по показателям результатов обучения студентов.

Следующим шагом визуализации является построение модели концептуальных классов (Class diagram). Согласно описанию модели обучения на естественном языке, можно определить такие классы (и их атрибуты) как (Рисунок 2):

- Студенты (Студент, Количество прослушанных лекций, Форма обучения, Время начала обучения, Время начала прослушивания лекции);
- Занятия (Занятие, Продолжительность, Вместимость);
- Выходные контроли (Выходной контроль, Вероятность сдачи, Критерий оценивания);
- Преподаватели (Максимальная загрузка);
- Консультанты (Максимальная загрузка);
- Консультации (Выходной контроль, Признак повторности).

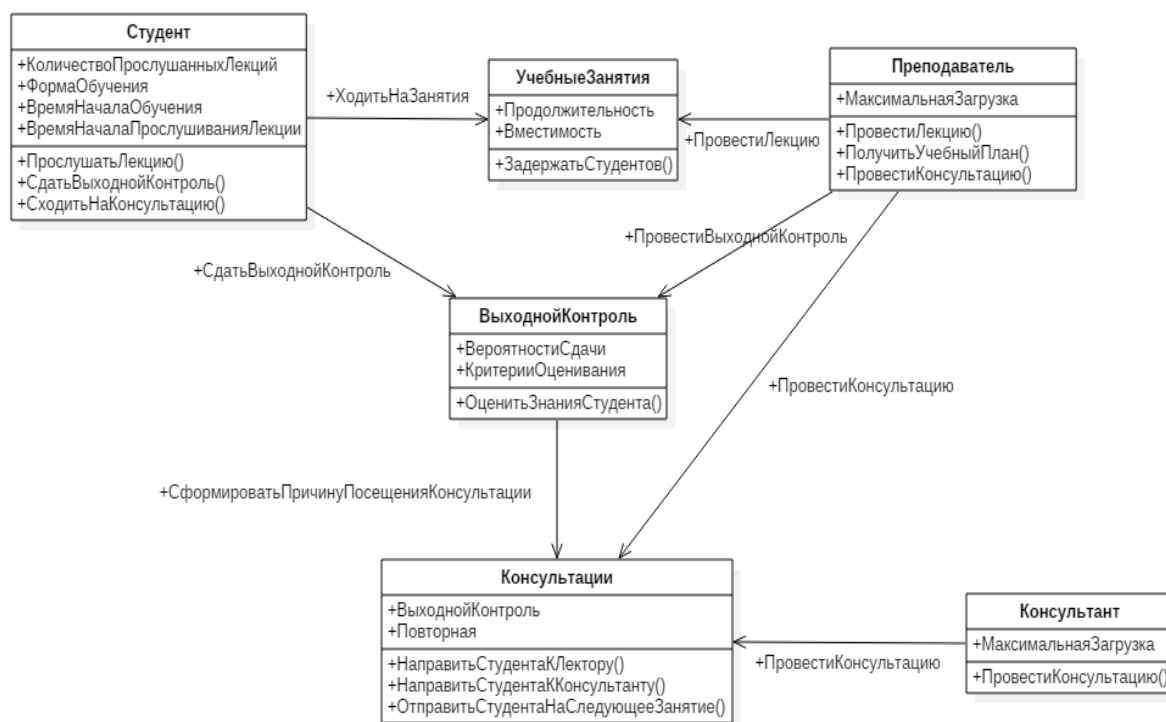


Рис. 2. Диаграмма классов системы исследования управления успеваемостью

На основании формализации классов можно перейти к этапу построения диаграммы последовательности (sequence diagram), которая приведена на рисунке 3. Первым шагом данной последовательности является проведение лекции, на которую студенты приходят в роли слушателей, а преподаватель – лектора. После прослушивания лекции, студенты должны разобраться в материале и приступить к написанию выходного контроля, который проводит преподаватель. По результатам выходного контроля студенты отправляются на следующую лекцию или на консультации к лектору или консультанту. В случае если у студента остаются вопросы после консультации с консультантом, тогда студенты отправляются к лектору. Если студент попадает к лектору, тогда считается, что он не освоил материал и ему требуется повторить лекцию дистанционно до тех пор, пока он не освоит материал и не сдаст успешно экзамен.

Данная модель является циклической, согласно которой после попадания студента на курс обучения происходит выбор его формы обучения: дистанционной или очной. Далее осуществляется проверка на необходимость в дальнейшем обучении. В случае если студент

успешно освоил материал всего курса и закончил все занятия, тогда он выходит из системы полностью, в обратном же случае он отправляется на прослушивание необходимой ему лекции с последующим выполнением выходного контроля.

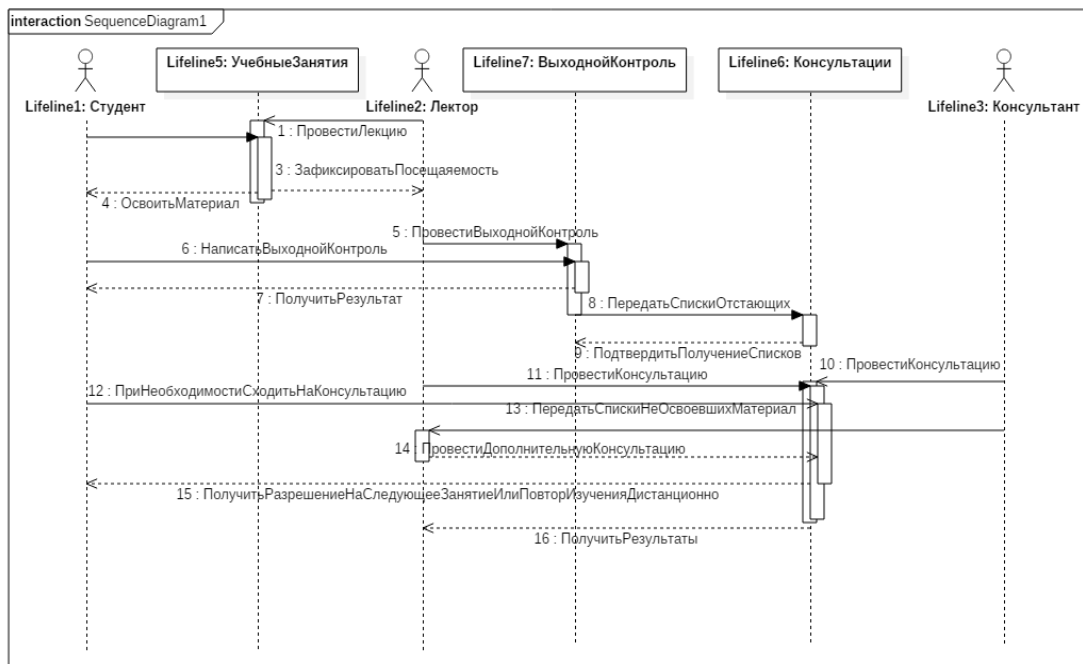


Рис. 3. Диаграмма последовательностей системы

По успешному результату сдачи выходного контроля, обучающиеся переходят в состояние ожидание следующего занятия. Если же студент справился с оцениванием, но у него появился вопрос, тогда он обращается за помощью к консультанту (ассистенту курса), который помогает в разборе несложных вопросов. В случае если студент понял вопрос, считается, что оценивание он прошел успешно и может двигаться по курсу дальше, если же консультация не помогла, тогда студент отправляется за помощью к лектору.

На рисунках 4 и 5 представлены диаграмма видов деятельности (activity diagram) и диаграмма состояний (state machine diagram) соответственно, которые основываются на представленном ранее алгоритме работы системы.

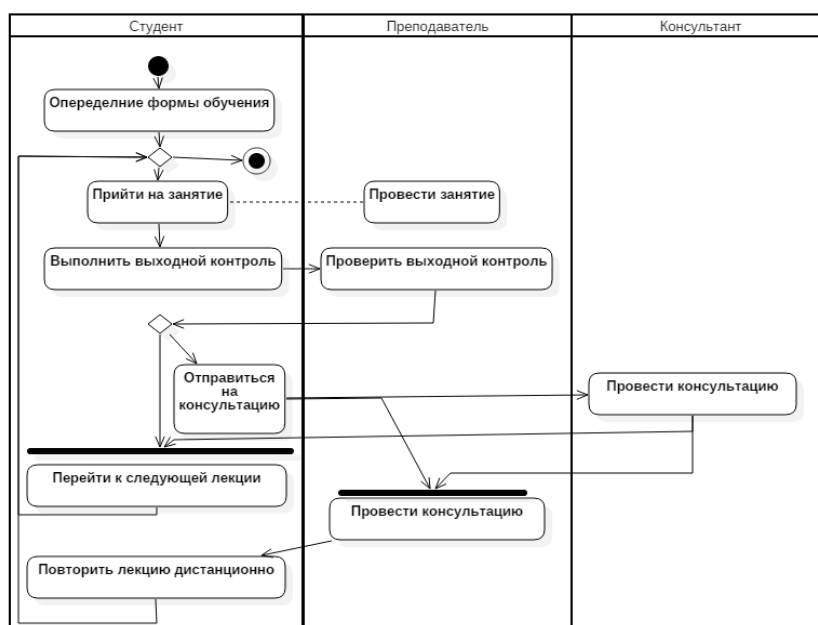


Рис. 4. Диаграмма видов деятельности акторов системы исследования управления успеваемостью



Рис. 5. Диаграмма состояний системы исследования управления успеваемостью

Выводы

В данной статье была рассмотрена модель системы «Исследование успеваемости студентов», которая описывает процесс обучения на очной форме с использованием дистанционного обучения, а также ее визуальное представление, описанное с использованием UML диаграмм. В качестве используемой модели дистанционного обучения была выбрана модель смешанной формы обучения, которую можно успешно применять в рамках высшего учебного заведения. Проиллюстрированный в данной статье фрагмент является лишь небольшим этапом исследования использования дистанционной формы обучения в университетах.

Перспективой дальнейших исследований является проведение ряда параметрических экспериментов с имитационной моделью, на основе чего определение эффективных стратегий проведения смешанного обучения.

Библиографический список

1. Воронина Т.П., Кашицин В.П., Молчанова О.П. Образование в эпоху новых информационных технологий. М.: АМО, 1995. – С. 220.
2. Андреев А.А., Солдаткин В.И. Дистанционное обучение: сущность, технология, организация. М.: Издательство МЭСИ, 1999. –С. 196.
3. Теория и практика дистанционного обучения: Учеб. пособие для студ. высш. пед. учеб. заведений / Под ред. Е.С. Полат. М.: Академия, 2004. – С. 416.
4. Tuninga R.S.J, Seinen I.B.J. The Supply and Demand of Distance Education in Russia / The World Bank, Bureau Cross, 1995. – P. 110.
5. Дистанционные образовательные технологии: проектирование и реализация учебных курсов / Под общ. ред. М.Б. Лебедевой. СПб.: БХВ-Петербург, 2010. – 336 с.
6. Галяев В.С., Гасанова З. А. О классификации моделей дистанционного обучения // Высшее образование в России. 2012. №4. –С. 103-108.

УДК 004.09

И. Ю. Коваленко, магистрант

Е.Н. Заикина, старший преподаватель

Севастопольский государственный университет

АДАПТИВНЫЕ МЕТОДЫ В PIR-ДАТЧИКАХ ДВИЖЕНИЯ

Аннотация: в статье предлагается адаптивный подход к выделению полезного сигнала в датчиках движения, позволяющий не только повысить качество фильтрации, но и применять их для охраны подвижных объектов. Проведена оценка вычислительной сложности различных методов расчёта коэффициентов передаточной функции адаптивного цифрового фильтра. В системе MATLAB реализована модель адаптивного фильтра и проведены исследования влияния помех различной природы на параметры фильтра.

Ключевые слова: PIR-датчик движения, адаптивная фильтрация, вычислительная сложность, нестационарные параметры, цифровая фильтрация

Annotation: adaptive approach to detect wanted signal in motion sensors is considered. It improves the accuracy of filtering and is able to use PIR-sensors on moving objects guard. The estimation of computational complexity to different approaches to transfer function evaluation for discrete filtering is done. The model is built in MATLAB and noise influence on filter is estimated.

Key words: PIR-sensor of motion, adaptive filtering, computational complexity, non-stationary parameters, digital filtering

Существующие современные датчики движения имеют сходные принципы работы и элементную базу, различие, в основном, состоит в конструктивных особенностях, обеспечивающих максимальную надёжность срабатывания датчика. Все они характеризуются большим количеством ложных срабатываний при максимальной чувствительности и пропуском реальных сигналов тревоги в случае низкой чувствительности. Совершенно неприемлемая ситуация возникает при охране движущихся объектов. Условия охраны, в этом случае, не позволяют использовать традиционные датчики движения, предназначенные для охраны неподвижных объектов.

Решить задачу охраны подвижных объектов и уменьшить количество ложных срабатываний возможно, если использовать в датчике движения принцип адаптивной фильтрации сигнала тревоги.

Адаптивные фильтры целесообразно применять тогда, когда полоса частот шума неизвестна или полоса частот сигнала изменяется со временем, при перекрытии спектров шума и полезного сигнала, когда параметры полезного сигнала и шума являются нестационарными.

Аналоговые фильтры уступают цифровым фильтрам (ЦФ) как по точности, так и по стабильности. Точность цифровых фильтров ограничена лишь длиной слова. Единственное, в чем ЦФ уступают аналоговым фильтрам – это быстродействие. Преимуществом ЦФ является также простота их реконфигурации. К недостаткам ЦФ можно отнести шум квантования сигналов, а также погрешности округления вычислений [1-3].

Функциональная схема предлагаемого устройства представлена на рисунке 1.

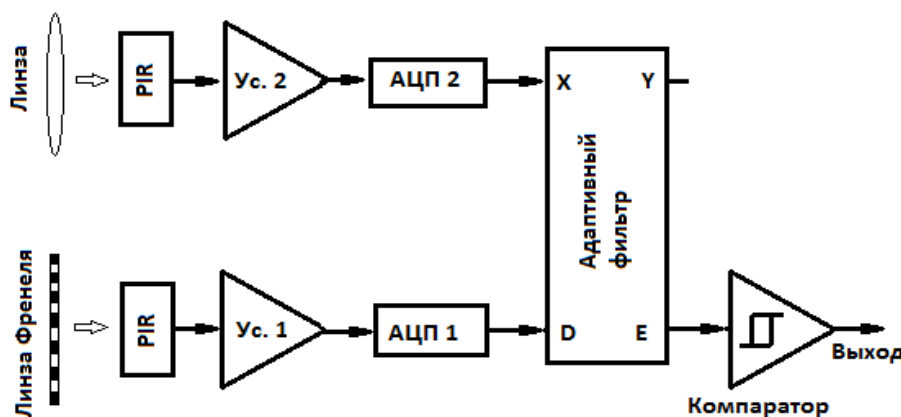


Рис. 1. Функциональная схема датчика

Из рисунка видно, что устройство состоит из двух практически идентичных каналов с той разницей, что в сигнальном канале используется линза Френеля, а в опорном канале обычная линза. Наличие опорного канала позволит компенсировать случайные флуктуации температуры и эффекты, возникающие при движении датчика.

Адаптивный фильтр (АФ) включён по схеме подавления помех. Здесь D – вход сигнала тревоги, X – вход сигнала опорного канала, E – сигнал ошибки. Структурная схема АФ представлена на рисунке 2.

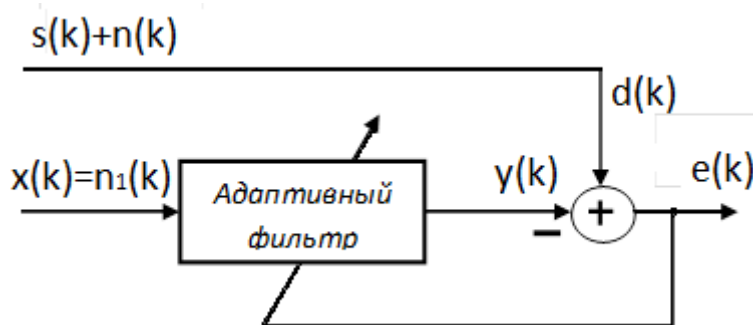


Рис. 2. Структурная схема АФ

Здесь $x(k)$ – опорный вход фильтра, куда подается сигнал помехи $x[k] = n1[k]$. На основной вход $d[k]$, подаётся результат суммы $d[k] = s[k] + n[k]$ полезного сигнала и помехи $n[k]$, коррелированной с помехой $n1[k]$. Сигнал помехи $x[k]$ и сигнал $s[k]$ с выхода датчика движения, должны быть не коррелированы. В результате работы адаптивный алгоритм настраивает коэффициенты фильтра так, чтобы минимизировать среднеквадратичную ошибку фильтра $e^2(k)$. После завершения настройки $y(k) = n(k)$, а сигнал ошибки $e(k) \approx s(k)$.

В результате исследований был определён наиболее оптимальный метод расчёта коэффициентов цифрового фильтра [1]. Было показано, например, что вычислительная сложность фильтра Винера равна $O(N^3)$ операций сложения-умножения, метода наискорейшего спуска $O(N^2)$, а вычислительная сложность алгоритма МНК составит $O(N)$ операций на итерацию. Это обстоятельство позволило отдать предпочтение методу МНК и его модификациям, для расчёта коэффициентов передаточной функции адаптивного цифрового фильтра.

Требования к самому вычислителю (микроконтроллеру) коэффициентов становятся вполне удовлетворительными по критериям быстродействия и соотношения цена-качество.

Моделирование работы фильтра проводилось в системе MATLAB [4]. В качестве входных данных для фильтра использовался выходной сигнал усилителя реального датчика с наложенными на него различными видами помех, как со второго канала детектора движения, так и встроенных в пакет MATLAB. На рисунке 3 представлены результаты работы программной модели адаптивного фильтра. В этом эксперименте входной сигнал фильтра представляет собой сумму сигнала с датчика движения и случайной последовательности распределённой по равномерному закону. При этом, амплитуда случайного сигнала сравнима с амплитудой полезного сигнала. Как видно из рисунка 3, результаты фильтрации вполне приемлемы для выделения импульсов датчика на уровне 0.2.

В заключение следует отметить, что проведённые исследования и макетирование самого датчика, показали состоятельность применения адаптивной фильтрации в PIR-датчиках для охраны как подвижных, так и неподвижных объектов. Расчёт коэффициентов передаточной функции по методу МНК позволит применять недорогие маломощные микроконтроллеры, что уменьшит стоимость изделия в целом.

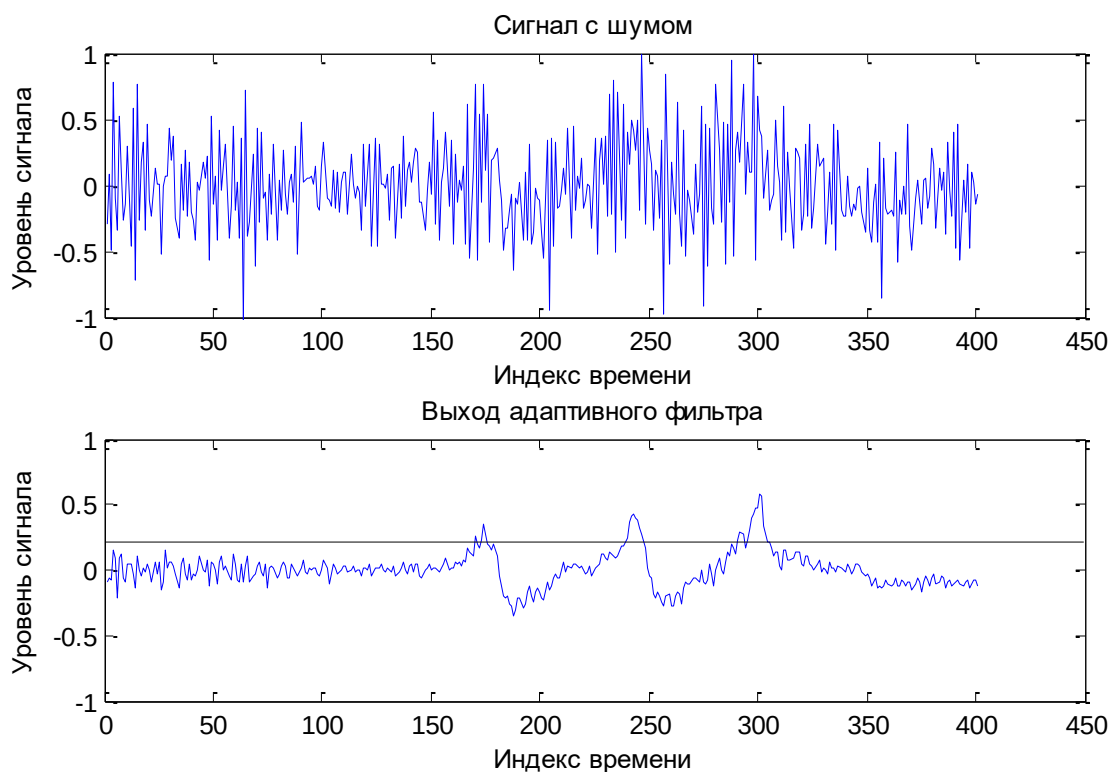


Рис. 3. Сигналы на входе и выходе адаптивного фильтра, размер шага сходимости $\mu_i=0.01$, число коэффициентов адаптивного фильтра $N=35$.

Библиографический список

1. Коваленко И. Ю. Исследование методов расчёта коэффициентов цифрового фильтра: Севастополь:, 13-ая Международная молодёжная научно-техническая конференция, «Современные проблемы радиоэлектроники и телекоммуникаций, РТ-2017», 20 — 24 ноября 2017 г.
2. Коуэн К.Ф.Н. Адаптивные фильтры: Пер. с англ./ Под ред. К.Ф.Н. Коуэна и П.М. Гранта. М.: Мир, 1988. — 392 с.
3. Джиган В.И. Адаптивная фильтрация сигналов: теория и алгоритмы. М.: Техносфера, 2013.— 528 с.
4. Сергиенко А.Б. Алгоритмы адаптивной фильтрации: особенности реализации в MATLAB.— <http://images.nature.web.ru/nature/2003/04/11/0001193683/06.pdf>

УДК 004.94

Ю.И. Пашкова, магистрант 1-го курса**О.В. Ченгарь, канд. техн. наук, доцент***Севастопольский государственный университет***МОДЕЛИРОВАНИЕ ДВИЖЕНИЯ ШКОЛЬНЫХ АВТОБУСОВ ПО ГОРОДСКИМ МАРШРУТАМ В СРЕДЕ ANYLOGIC С ПОМОЩЬЮ ГЕОГРАФИЧЕСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ**

Аннотация: В современном мире остро стоит проблема управления транспортными потоками. Одним из аспектов обеспечения безопасности дорожного движения является вопрос безопасной транспортировки школьников из учебного заведения. Статья посвящена моделированию движения школьных автобусов по городским маршрутам с помощью среды имитационного моделирования AnyLogic.

Ключевые слова: имитационное моделирование, транспортная логистика, оптимизация, AnyLogic.

Annotation: In today's world, the problem of managing traffic flows is acute. One of the components of ensuring road safety is the issue of providing safe home-to-school-to-home transport for pupils. The article concerns the modeling of the movement of school buses on urban routes with the help of the simulation environment AnyLogic.

Key words: simulation modeling, transport logistics, optimization, AnyLogic.

Жизнь современного общества невозможна без транспорта, современные транспортные средства обеспечивают скорость, комфортабельность, безопасность передвижения. Вместе с тем они являются причиной возникновения чрезвычайных ситуаций, в результате которых травмируются и гибнут люди.

Источником повышенной опасности на дорогах является совокупность факторов, среди которых на одном из первых мест стоят автомобили. Это связано с огромным их количеством, большой скоростью движения и массой, ограниченной маневренностью, наличием «человеческого фактора», невозможностью мгновенной остановки в случае возникновения опасной ситуации. Транспортные средства представляют угрозу не только для взрослых, но, особенно, и для детей, так как у них неразвито чувство опасности перед движущимся транспортом.

Для обеспечения безопасности школьников в [1] было предложено внедрить программу «Школьный автобус» в городскую транспортную систему для государственных образовательных учреждений. Так же в [1] было проведено исследование различных маршрутов перевозки детей в школы. Исследование заключалось в решении задачи, которая сводится к отысканию самого выгодного маршрута, проходящего через остановки по одному разу с последующим возвратом автобуса в школу.

В последние годы активно применяется имитационное моделирование для усовершенствования транспортных систем. Имитационная модель – это компьютерная программа, которая описывает структуру и воспроизводит поведение реальной системы во времени. Имитационная модель позволяет получить подробную статистику о различных аспектах функционирования системы в зависимости от входных данных [2].

В данной статье рассматривается процесс построения имитационной модели движения школьных автобусов по городским маршрутам в среде AnyLogic. Целью данного имитационного моделирования процесса транспортировки школьников является предотвращение нерационального использования школьных автобусов. Оптимизировать движение школьных автобусов можно благодаря имитационному моделированию.

Моделирование процесса движения автобусов будет осуществляться с помощью географической информационной системы «Карта» (ГИС Карта), которая имеется в среде имитационного моделирования AnyLogic. Карты ГИС часто используются в агентных моделях для задания ГИС пространства, в котором обитают агенты. ГИС пространство позволяет размещать агентов в геопространственной среде, заданной с помощью карты ГИС [3].

Формализуем постановку задачи для дальнейшего моделирования.

Имеется 11 остановок на следующих улицах: Пр. Генерала Острякова, ул. Маринеско, Малахов Курган, ул. Меньшикова, Пл. Нахимова, Казачья Бухта, ул. Юмашева, Пр. Победы, ул. Пожарова, ул. Жидилова, ул. Новикова. На данные остановки один раз в определенный

промежуток времени приезжает автобус, целью которого является транспортировка школьников от школы до дома (остановок) после занятий. Определённый промежуток времени - время, в которое школьники возвращаются домой, а именно это время с 13.00 до 17.00 (4 часа).

Пусть школа, из которой будут приезжать дети, находится на улице Хрусталева. На посадку детей в автобусы в школах, а также на высадку детей на остановках требуется время. Пусть, посадка детей занимает от 15 до 30 минут, а высадка - от 5 до 10 минут.

Требуется промоделировать процесс транспортировки школьников для оценки оптимального количества автобусов, учитывая, что школа имеет всего 5 автобусов.

Перед построением имитационной модели введем понятие «вызов». Под вызовом будем понимать время, в которое должен начаться процесс посадки детей в автобус в школе. Каждая остановка имеет свой «вызов». То есть, если автобус должен приехать на остановку в $t_{\text{прибытия}}$, то вызов для этой остановки вычисляется как разность между $t_{\text{прибытия}}$ и суммой, слагаемыми которой являются $t_{\text{посадки}}$ и $t_{\text{транспортировки}}$, требуемые на посадку и транспортировку детей соответственно.

$$t_{\text{вызова}} = t_{\text{прибытия}} - (t_{\text{посадки}} + t_{\text{транспортировки}}) \quad (1)$$

Построение имитационной модели можно свести к следующим действиям:

- 1) Задать местоположение и маршруты до всех пунктов;
- 2) Описать логику обработки вызова школой, где необходимо учесть время на посадку школьников, транспортировку до остановки, высадку школьников, возврат автобуса в школу.

Исходными данными являются координаты остановок, координаты школы. Найдем на карте необходимые остановки и преобразуем их в ГИС точки, после чего на карте появится метка выбранного объекта (см. рисунок 1).

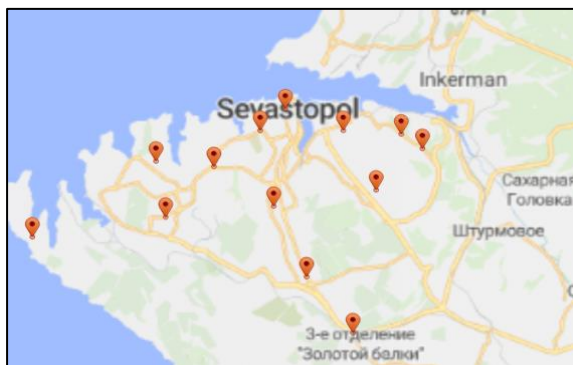


Рис. 1. Карта с метками всех объектов

Для удобства размещения агентов в отмеченных точках объединим их в коллекцию *расположениеОстановок*. Далее, создадим популяцию агентов *Остановки* для ранее найденных точек и выберем для нее анимацию. Для того, чтобы задать начальное количество агентов, запишем следующее выражение: *расположениеОстановок.size()*. Эта функция возвращает количество элементов коллекции *расположениеОстановок* (11 штук). Свяжем остановки с их координатами (см. рисунок 2).

Локальная переменная «index» возвращает номер популяции. По этому номеру осуществляется обращение к элементу из коллекции, которому будет соответствовать одна из точек-остановок на карте.

Также создадим тип агента с именем *Школа* для размещения школы. Далее, через его свойства выберем соответствующую точку на карте, где будет располагаться данная школа.

Создадим агент *Вызов*, у которого будет параметр *местоОстановки* типа *Остановка*.

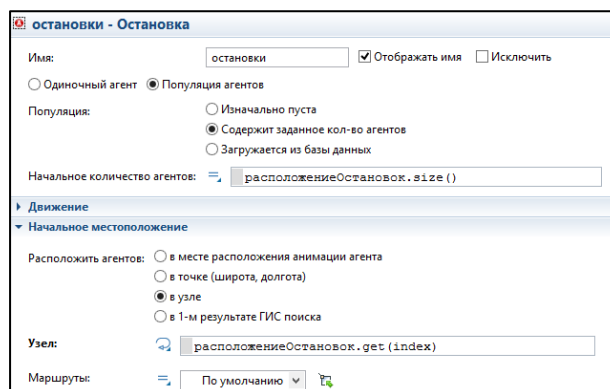


Рис. 2. Свойства популяции агента Остановка

Так как транспортировка школьников осуществляется автобусами, то для того, чтобы их добавить в модель необходимо создать новую популяцию агентов. При создании популяции также создадим отдельный параметр *количествоАвтобусов* типа *int* с начальным значением, равным 5. Так как все автобусы принадлежат школе, что необходимо в свойствах популяции агента *Автобусы* указать начальное местоположение.

Теперь нужно указать, что автобусы являются ресурсом школы. Для этого воспользуемся блоком «Resource Pool», который задает набор доступных ресурсов – автобусов. Свойства блока Resource Pool показаны на рисунке 3.

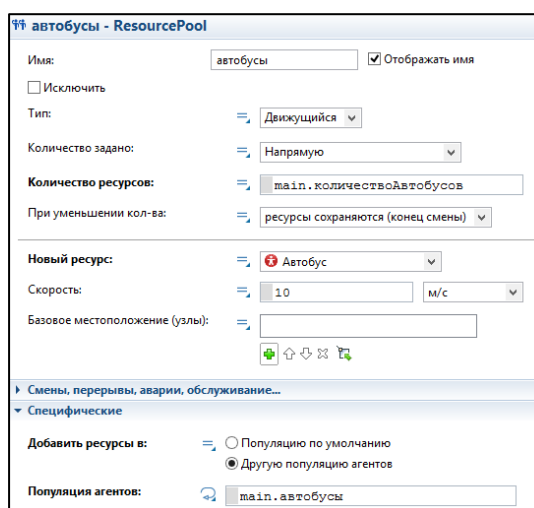


Рис. 3. Свойства блока Resource Pool

Логику работы остановок можно рассматривать как последовательные переходы из состояния «отсутствие ожидания детей» в состояние «ожидание детей» и обратно. Чтобы смоделировать данный процесс воспользуемся диаграммой состояний (см. рисунок 4).

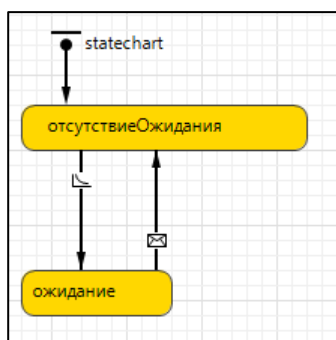


Рис. 4. Диаграмма состояний

Переход из состояния «отсутствие ожидания» в состояние «ожидание» (см. рисунок 5) происходит с заданной интенсивностью (один раз в промежуток времени с 13 00 до 17 00). Как только остановка переходит в состояние ожидания, должен быть сформирован «вызов».

Рис. 5. Свойства состояния «ожидание»

В свойстве «Действие при входе» состояния «ожидание» в первой строчке формируется новый вызов с параметром *this*, который указывает на остановку, с которого вызов «поступает». Вторая строчка кода нужна для отправки – функция *send*, которая имеет два аргумента: что отправить (созданный выше вызов) и кому отправить (в школу – агенту *школа*).

На диаграмме также присутствует и обратный переход (из состояния «ожидание» в состояние «отсутствие ожидания»). Свойства перехода отражены на рисунке 6.

Рис. 6 – Свойства перехода

Так как в школе необходимо выделять автобус для транспортировки детей на остановку, рассмотрим модель обработки вызова агентом *Школа*, представленную на рисунке 7.

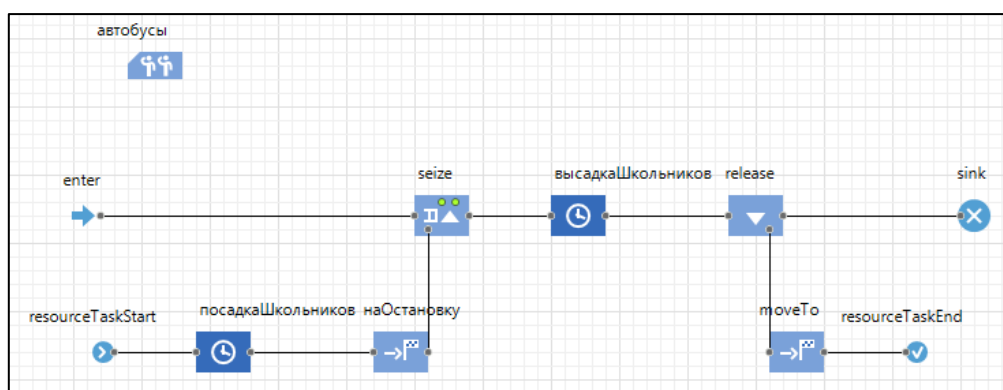


Рис. 7. Модель обработки вызова

Вход в процесс осуществляется через блок *Enter*, в которые поступают вызовы. Далее полученный вызов поступает в очередь на ожидание ресурсов. Перед тем, как ресурс будет захвачен, он должен быть подготовлен (посадка школьников). Подготовка ресурсов и их отправка происходит в специальном подпроцессе для ресурсов, который начинается с блока

ResourceTaskStart. После того, как ресурс подготовлен, его отправляют на остановку. Как только автобус прибыл на остановку, необходимо осуществить высадку детей. После выполнения вызова ресурс становится свободным, а агент Вызов удаляется. Так как ресурс освобожден, он должен быть отправлен в школу. Подпроцесс для ресурса должен быть закончен блоком *ResourceTaskEnd*, для того, чтобы ресурс вернулся в общий пул с ресурсами и был доступен для нового захвата. На рисунках 8, 9, 10, 11 отображены свойства блоков модели.

а) *высадкаШкольников - Delay*

б) *посадкаШкольников - Delay*

Рис. 8. Свойства блоков (а – блок «Высадка школьников»; б – блок «Посадка школьников»)

а) *наОстановку - MoveTo*

б) *moveTo - MoveTo*

Рис. 9. Свойства блоков (а – блок «на остановку»; б – блок «move to»)

seize - Seize

Рис. 10. Свойства блока Seize

moveTo - MoveTo

Рис. 11. Свойства блока moveTo

Все входящие сообщения обрабатываются в стандартном блоке *connections* (см. рисунок 12), который по умолчанию существует внутри каждого агента.

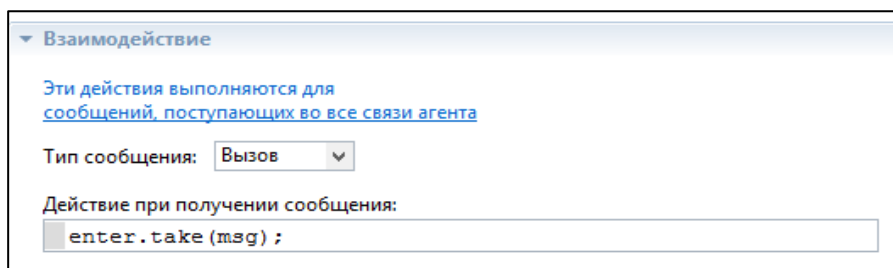


Рис. 12 – Вкладка «Взаимодействие» блока connections

В поле *Действие при получении сообщения* введена команда, которая указывает, что блок *Enter* должен взять на обработку (функцией *take*) пришедшее сообщение (используя локальную переменную *msg*). Тип сообщения указывается согласно созданному ранее в агенте *Школа*.

После запуска модели можно увидеть, как автобусы осуществляют передвижение по городским маршрутам (см. рис.13).

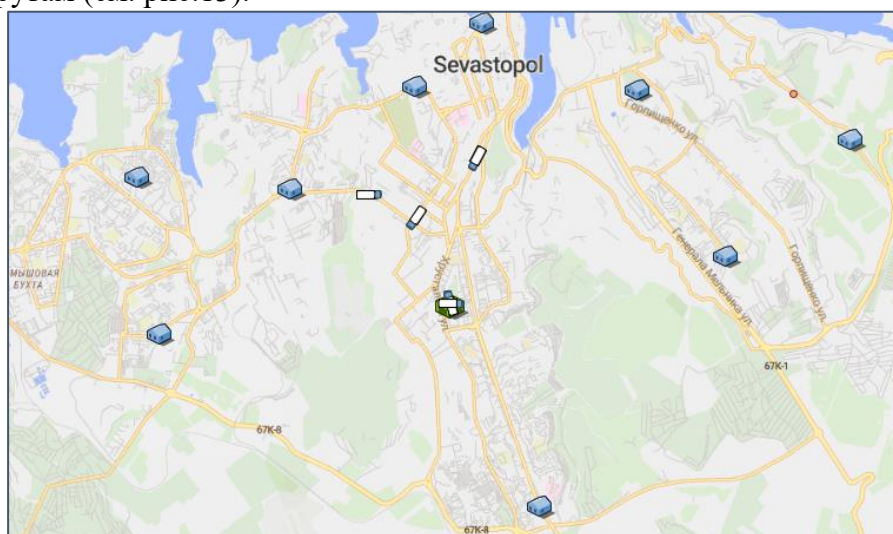


Рис. 13. Демонстрация модели

Можно увидеть, как автобусы выезжают со школы, доезжают до остановок и возвращаются обратно. Автобусы движутся по реальным маршрутам, которые, как и сама карта, подгружаются из сети во время исполнения модели.

По ходу обработки вызова фиксируется количество захваченных ресурсов, по которому можно оценить рациональность использования ресурсов. При использовании пяти автобусов, средняя загруженность ресурсов составляет примерно 40 процентов (см. рис. 14). То есть, часть автобусов простаивает и приносит убытки учебному заведению, следовательно, нужно провести оптимизацию ресурсов.

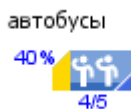


Рис. 14. – Загруженность ресурсов

Сформулируем задачу оптимизации движения автобусов. Необходимо задействовать такое количество автобусов для перевозки обучающихся, при котором загрузка автобусов будет максимальной, но составлять не более 85%. При этом, количество автобусов должно быть минимальным, но не более 5.

Математическая модель оптимизационной задачи:

Целевая функция №1: загрузка автобусов $\rightarrow \max$ (2)

Целевая функция №2: количество автобусов $\rightarrow \min$ (3)
при ограничениях:

$$\begin{cases} \text{загрузка автобусов} \leq 85\% \\ \text{число автобусов} \leq 5 \\ \text{число автобусов} \geq 1 \end{cases} \quad (4)$$

Преобразуем частные критерии (загрузка автобусов, количество автобусов) в один комплексный критерий. Так как оценки по частным критериям имеют разные шкалы, то необходимо провести нормировку, то есть сделать их безразмерными и сделать так, чтобы они принимали значения от 0 до 1 [4].

Пусть лучшая оценка критерия К составляет $f_{\max}$, а худшая - $f_{\min}$.

Формула нормировки при максимизации критерия:

$$f = \frac{f_{\text{измер}} - f_{\min}}{f_{\max} - f_{\min}} \quad (5)$$

Формула нормировки при минимизации критерия:

$$f = \frac{f_{\max} - f_{\text{измер}}}{f_{\max} - f_{\min}} \quad (6)$$

Для критерия «количество автобусов»:

$$\begin{aligned} f_{\min} &= 5 \\ f_{\max} &= 1 \\ f &= \frac{f_{\text{измер}} - 5}{1 - 5} = \frac{f_{\text{измер}} - 5}{4} \end{aligned}$$

Для критерия «загрузка автобусов»:

$$\begin{aligned} f_{\min} &= 0 \\ f_{\max} &= 0,85 \\ f &= \frac{f_{\text{измер}} - 0}{0,85 - 0} = \frac{f_{\text{измер}}}{0,85} \end{aligned}$$

Согласно формулам (5) и (6), формула свертки критериев в среде AnyLogic будет иметь вид:

$$(\text{root.школа.автобусы.utilization()}/0,85) - (\text{root.количествоАвтобусов} - 5)/4.$$

Для осуществления оптимизации создадим оптимизационный эксперимент в среде AnyLogic (рисунок 15).

В качестве изменяемого параметра выбран параметр *количество Автобусов*, которое меняется дискретно от 1 до 5. Строчка кода *root.школа.автобусы.utilization()* позволяет обратиться к параметру средней загрузки *utilization()* ресурса *автобусы* агента *Школа*, находящегося на корневом агенте *root* (в данном случае *main*). На вкладке «Требования» также указано, что средняя загрузка автобусов должна быть не более 0,85.

При равнозначном вкладе критериев в комплексный критерий, получаем, что 3 автобуса являются оптимальным количеством для школы (см. рисунок 16).

Теперь установим значение параметра «количество автобусов» равное 3. После запуска имитационной модели видим, что среднее значение загруженности ресурсов учебного заведения составляет 70%. Если же количество автобусов принять равным 2, то загруженность ресурсов будет составлять 90%, что не удовлетворяет требованиям задачи.

Optimization - Оптимизационный эксперимент

Имя: ☐ Исключить

Агент верхнего уровня:

Целевая функция: ☐ минимизировать ☒ максимизировать

$(\text{root.школа.автобусы.utilization()}/0.85) - (\text{root.количествоАвтобусов}-5)/4$

☒ Количество итераций:

☐ Автоматическая остановка

Максимальный размер памяти: M6

Параметры

Параметры:

Параметр	Тип	Значение			
		Мин.	Макс.	Шаг	Начальное
количе...обусов	дискретный	1	5	1	

Модельное время

Ограничения

Требования

Требования (проверяются после "прогона" для определения того, допустимо ли найденное решение):

Вкл.	Выражение	Тип	Гран...
☒	root.школа.автобусы.utilization()	<=	0.85

Рис. 15. – Свойства оптимизационного эксперимента

	Текущее	Лучшее
Итерация:	5	1
Функционал: ↑	0.349	0.443

Параметры

количествоАвтобусов	4	3
---------------------	---	---

Копировать лучшее решение в буфер

Рис. 16. – Результат оптимизационного эксперимента

Библиографический список

1. Ю.И. Пашкова, О.В. Ченгарь. Методы комбинаторной оптимизации для исследования оптимальных городских маршрутов движения школьного автобуса // Оптимизация и моделирование в автоматизированных системах: материалы Всерос. молодежной научной школы. Воронеж: ФГБОУ ВО «Воронежский государственный технический университет», 2017. Ч.2. 182 с
2. Куприяшкин, А.Г. Основы моделирования систем [Текст]: учеб. пособие / А.Г. Куприяшкин; Норильский индустр. ин-т. – Норильск: НИИ, 2015. – 135 с.
3. Справка AnyLogic. ГИС Карта. [Электронный ресурс]. Режим доступа - <https://help.anylogic.ru/index.jsp?topic=%2Fcom.anylogic.help%2Fhtml%2Fgis%2FGIS+Map+Shape.html>
4. Многокритериальные задачи принятия решений. Часть 2. Свёртка критериев. [Электронный ресурс] / авт. Исканцев Н.В. – Электрон. дан. – Россия, 2012 – . – Режим доступа: <http://crypto.hut2.ru/svertka.html>

УДК 004.942

Е.Ю. Первушина, магистрант**В.И. Шевченко, к.т.н, доцент***Севастопольский государственный университет***ФУНКЦИОНАЛЬНО-СТРУКТУРНАЯ МОДЕЛЬ КОММУНИКАЦИОННЫХ ВЗАИМОДЕЙСТВИЙ
В ГБУ РК «МНОГОФУНКЦИОНАЛЬНЫЙ ЦЕНТР ПРЕДОСТАВЛЕНИЯ ГОСУДАРСТВЕННЫХ
И МУНИЦИПАЛЬНЫХ УСЛУГ»**

Аннотация: в данной статье разработана и представлена функционально-структурная модель ГБУ РК «Многофункциональный центр предоставления государственных и муниципальных услуг» на основе методологии IDEF0. В результате многоуровневой декомпозиции сформированы группы функций, которые отображаются в модели. Представленная модель используется в качестве инструмента исследования процессов протекающих в ГБУ РК «Многофункциональный центр предоставления государственных и муниципальных услуг» с целью повышения эффективности и оптимизации работы.

Ключевые слова: функционально-структурная модель, IDEF0, моделирование, диаграмма декомпозиции

Annotation: in this article, a functional-structural model of the State budgetary institution of the Republic of Crimea "Multifunctional Center for Provision of State and Municipal Services" was developed and presented on the basis of the IDEF0 methodology. As a result of the multi-level decomposition, groups of functions are created that are displayed in the model. The presented model is used as a tool for studying the processes of the State budgetary institution of the Republic of Crimea "Multifunctional Center for Provision of State and Municipal Services" with a view to increasing efficiency and optimizing work.

Key words: functional-structural model, IDEF0, modeling, decomposition diagram

Главными направлениями административной реформы осуществляемой в России последнее десятилетие, являются повышение качества жизни граждан и эффективности деятельности государственных органов власти. Одной из важнейших целей реформы было повышение доступности и качества государственных услуг.

Основной идеей создания системы многофункциональных центров предоставления государственных и муниципальных услуг является реализация принципа «одного окна», когда гражданин освобождается от необходимости получать справки в других государственных учреждениях или ходить по инстанциям.

На основании распоряжения Совета министров Республики Крым от 01.12.2014 №1287-р «О создании Государственного бюджетного учреждения Республики Крым «Многофункциональный центр предоставления государственных и муниципальных услуг» с изменениями от 23.01.2015 №32 создано Государственное бюджетное учреждение Республики Крым «Многофункциональный центр предоставления государственных и муниципальных услуг» [1, 2].

С помощью моделирования различных областей деятельности можно эффективно проанализировать и оптимизировать исследуемую систему. Для решения задач моделирования сложных систем существуют проверенные методологии и стандарты, к которым относится и методологии семейства IDEF (Integrated DEFinition) [3, 4].

В данной статье предлагается разработка функционально-структурной модели средствами IDEF0, как концептуальной модели проекта автоматизации бизнес-процессов государственного учреждения.

Функционально-структурной модель процесса исследования коммуникационных взаимодействий, разработанная с помощью методологии IDEF для ГБУ РК «Многофункциональный центр предоставления государственных и муниципальных услуг», представлена на рисунке 1, где изображена контекстная диаграмма A0, представляющая базовые подпроцессы с их взаимосвязями.

Основным методологическим приемом IDEF0 является декомпозиция – каждый функциональный блок может быть представлен своим более подробным описанием в виде дочерней диаграммы, на которой могут присутствовать несколько соединенных между собой функциональных блоков.

Поскольку конечной целью данного проекта является оптимизация работы многофункционального центра (МФЦ), то на этапе предварительного анализа и построения

концептуальной модели необходимо как можно более полно учесть все входные, выходные параметры модели, а так же управляющие воздействия и исполнительные механизмы. Эти элементы, согласно методологии IDEF0 представлены соответствующими входами (выхода) по сторонам прямоугольника контекстной диаграммы A0 [5].

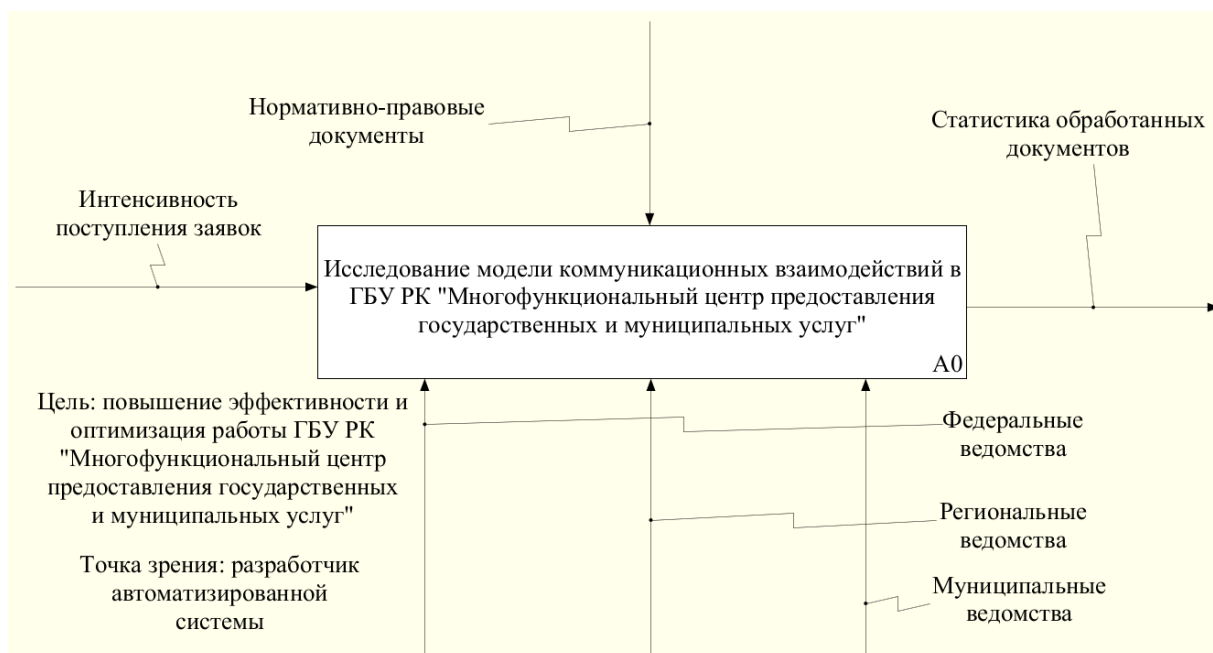


Рис. 1. Функционально-структурная модель процесса исследования коммуникационных взаимодействий

В результате декомпозиции контекстной диаграммы A0, получаем диаграмму следующего уровня (рисунок 2) состоящую из 6 блоков:

1. Получение талона.
2. Приём документов оператором.
3. Формирование документов для отправки в ведомство.
4. Отправка документов в ведомство.
5. Получение документов из ведомства.
6. Выдача готового результата.

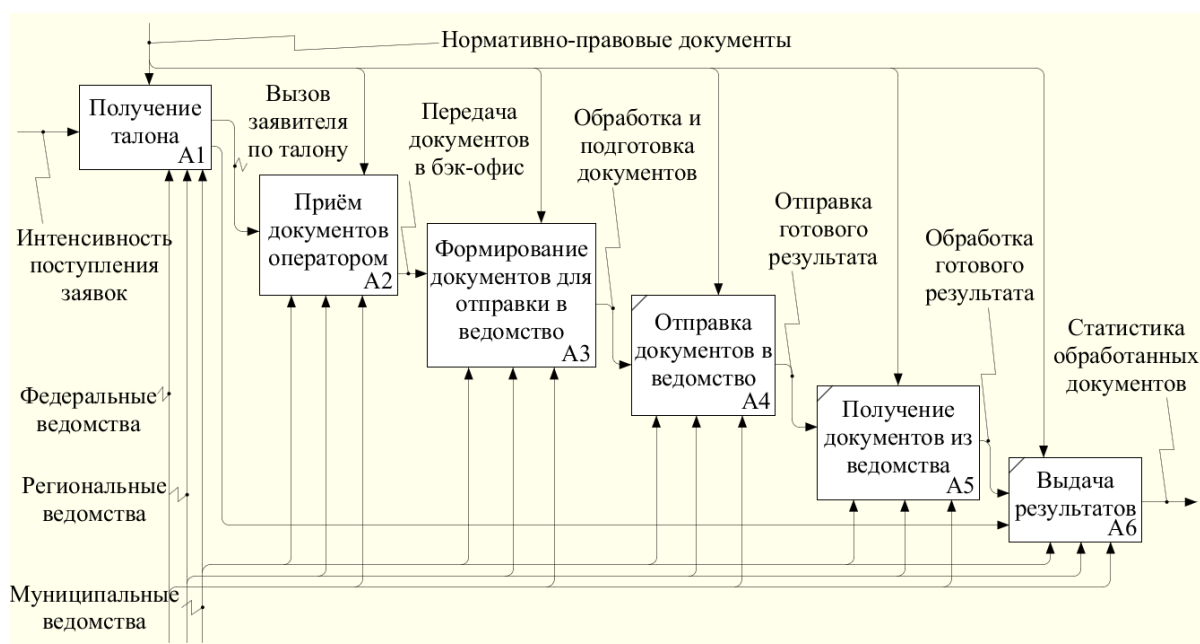


Рис. 2. Диаграмма декомпозиции блока A0

Исполнительными механизмами на данном уровне декомпозиции являются автоматизированные информационные системы и операторы связи на уровнях: федеральных, региональных и муниципальных ведомств.

Блок «Получение талона» A1 (рисунок 2) – в результате декомпозиции состоит из трёх блоков:

1. Регистрация на портале «Госуслуги».
2. Взятие талона на выбранную услугу и дату.
3. Взятие талона в терминале.

В данной диаграмме декомпозиции предполагается взятие талона на получение государственных и муниципальных услуг двумя способами:

– через портал «Госуслуги», чтобы воспользоваться предварительной записью, необходимо авторизоваться в личном кабинете при помощи учетной записи на Едином портале государственных и муниципальных услуг (ЕПГУ), и взять талон на любую свободную дату и время. Если учетная запись отсутствует, нужно пройти процедуру регистрации на ЕПГУ;

– через терминал, талон выдается на любое свободное время в день обращения.

Этот алгоритм представлен на рисунке 3.

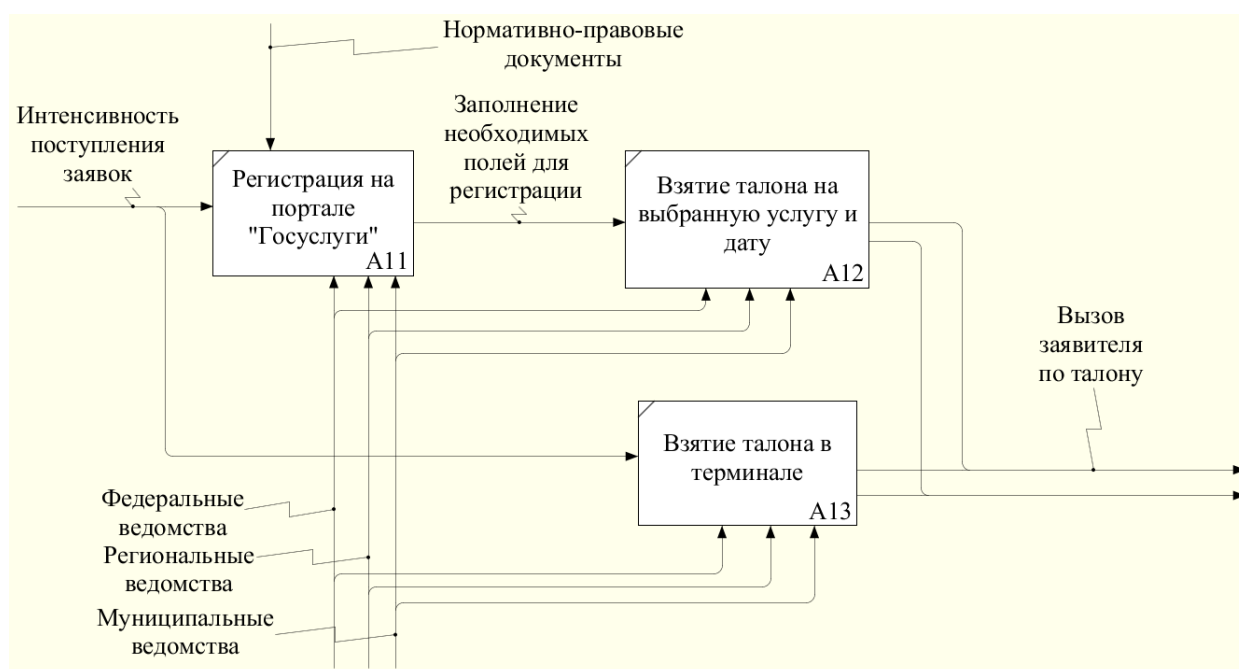


Рис. 3 Диаграмма декомпозиции блока A1

После получения талона заявитель приходит в центр «Мои документы» в выбранную дату и время для получения государственной услуги.

Блок «Приём документов оператором» A2 – в результате декомпозиции состоит из трёх блоков (рисунок 4):

1. Вызов заявителя по талону.
2. Приём документов оператором.
3. Формирование пакета документов.

Вызов заявителя осуществляется автоматически и сопровождается отображением номера талона и номера окна, к которому приглашается заявитель, на информационных табло в залах МФЦ.

Основанием для начала предоставления государственной (муниципальной) услуги является личное обращение заявителя (его представителя) с комплектом документов, необходимых для получения государственной (муниципальной) услуги.

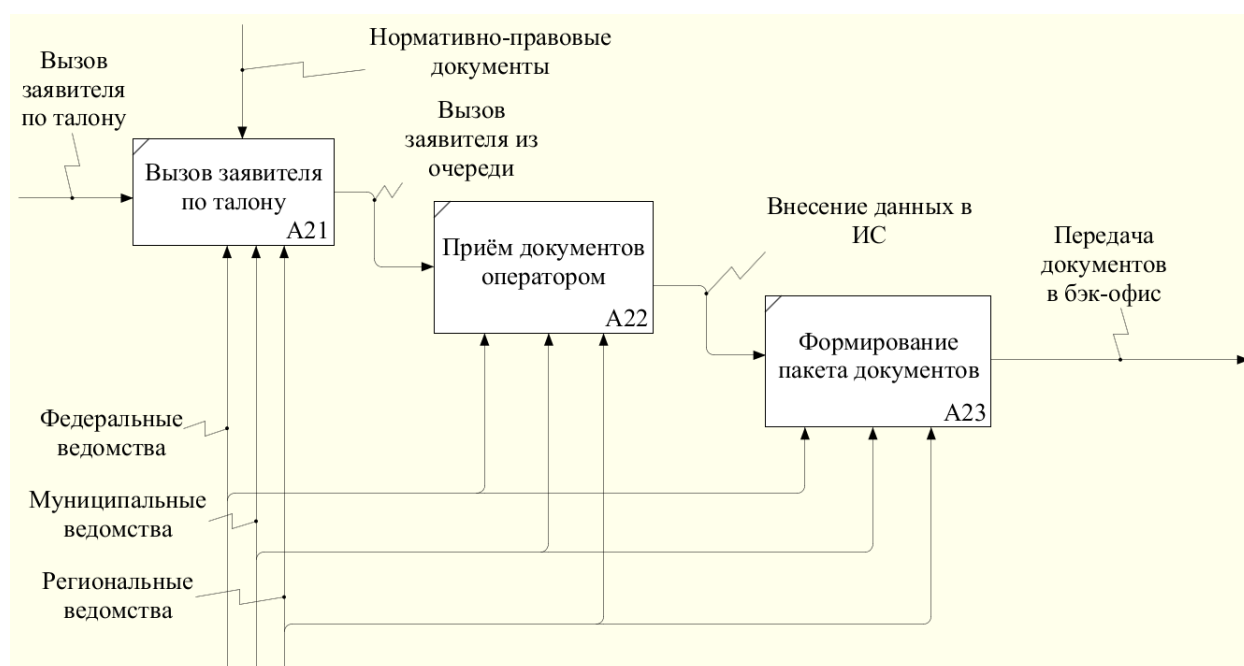


Рис. 4. Диаграмма декомпозиции блока A2

Специалист МФЦ, ответственный за обработку документации принимает комплект документов от специалиста МФЦ ответственного за прием документов у заявителей.

Блок следующего уровня «Приём документов оператором» A22 – в результате декомпозиции состоит из трёх блоков (рисунок 5):

1. Проверка правильности заполнения документов.
2. Формирование реестров для передачи в ведомство.
3. Передача документов курьеру.

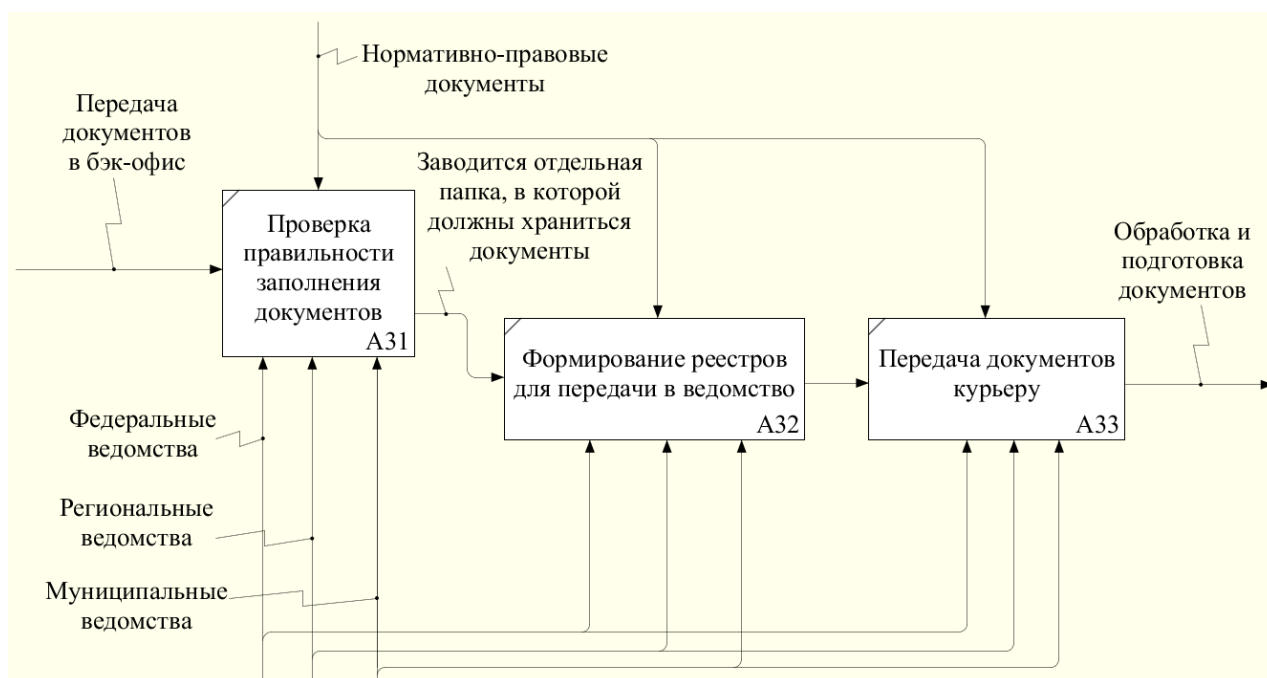


Рис. 5. Диаграмма декомпозиции блока A3

После получения готового результата из ведомства, документы поступают специалисту ответственному за обработку документации, он формирует окончательный

пакет выдаваемой заявителю документации (результат предоставления государственной (муниципальной) услуги) и передает результат предоставления государственной (муниципальной) услуги специалисту МФЦ, ответственному за выдачу документов. Для получения готового результата заявитель должен взять талон на получение документов и по талону обратиться к специалисту ответственному за выдачу документов.

Таким образом базовые бизнес-процессы автоматизации государственного учреждения по обслуживанию заявок пользователей могут быть декомпозированы на три уровня иерархии, включая от двух до четырех блоков на каждом уровне, что соответствует методологии IDEF0.

Представленная в работе функционально-структурная модель коммуникационных взаимодействий ГБУ РК «Многофункциональный центр предоставления государственных и муниципальных услуг» является базой для разработки комплекса имитационных моделей в среде AnyLogic, что позволит определить параметры входных потоков (интенсивность поступления заявок в систему, закон распределения случайных интервалов времени между поступлениями заявок, трудоемкость обработки заявок и т.п.), характеристики обрабатывающих устройств модели и провести ряд экспериментов с целью повышения эффективности обработки заявок пользователей за счет сокращения времени обработки на узлах информационных систем МФЦ.

Библиографический список

1. Федеральный закон от 27.07.2010 г. № 210-ФЗ "Об организации предоставления государственных и муниципальных услуг". Российская газета - Федеральный выпуск №5247 (168) от 30 июля 2010 г.
2. Постановление Правительства Российской Федерации от 22.12.2012 г. № 1376 г. Москва "Об утверждении Правил организации деятельности многофункциональных центров предоставления государственных и муниципальных услуг". Российская газета - Федеральный выпуск №5976 (303) от 31 декабря 2012 г.
3. Методология функционального моделирования IDEF0: Руководящий документ. М.: ИПК Издательство стандартов, 2000 - 75 с.
4. Методические указания для выполнения лабораторных работ на тему «Структурный подход к моделированию критических систем» по дисциплине «Моделирование критических процессов» для магистров дневной и заочной форм обучения по направлению 09.04.01 «Информатика и вычислительная техника»/ Сост.: В.И. Шевченко – Севастополь: 2016. – 40 с.
5. Белая К.В., Шевченко В.И. Диверсификация моделей поведения сотрудников удаленного офиса // Сборник статей студенческой научно-технической конференции «Мир компьютерных технологий» . Научный редактор Е.Н. Машенко. – Севастополь: Изд-во СевГУ, 2017.— С. 7-16.

УДК 004.942

К.Г. Рызывыло, магистрант**В.И. Шевченко, канд. тех. наук, доцент***Севастопольский государственный университет***КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ СИСТЕМЫ РАЗМЕЩЕНИЯ ФАЙЛОВ В РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СРЕДАХ**

Аннотация: В статье представлена концептуальная и имитационная модели системы размещения файлов в распределенной вычислительной среде. Модель реализуется с целью оптимизации времени доступа к файлам и объемов занимаемой памяти. Концептуальная модель реализована с применением методологии структурно-функционального анализа IDEF0. Имитационная модель реализована в среде многоподходового моделирования AnyLogic с использованием компонент визуализации процесса размещения файлов.

Ключевые слова: распределенная система, многоагентный подход, имитационное моделирование, преимущества имитационного моделирования, нотация IDEF0, Anylogic.

Annotation: The article presents a conceptual and simulation model of the file allocation system in a distributed computing environment. The model is implemented to optimize the access time to files and the memory used. The conceptual model is implemented using the methodology of structural and functional analysis of IDEF0. The simulation model is implemented in the environment of the multi-path modeling AnyLogic using the visualization component of the file allocation process.

Keywords: distributed system, multi-agent approach, simulation modeling, advantages of simulation modeling, IDEF0 notation, Anylogic.

Введение

Каждый день человек использует гигабайты информации. Для получения доступа к информации требуется организовать быстрый и надежный способ обращения к ней. Данной проблемой занимаются многие известные компании: Google, Amazon, Microsoft и др. Каждая из компаний имеет множество решений данной проблемы. В данной статье будет рассмотрен метод хранения информации без дублирования данных.

1. Анализ методов исследования

С появлением первых компьютерных систем решались задачи объединения компьютеров для увеличения совокупных вычислительных мощностей. Такие задачи привели к появлению нового вида компьютерных систем — распределенных систем. Распределенная система — это набор независимых компьютеров, представляющий их пользователям единой объединенной системой [1].

Важной характеристикой распределенной файловой системы является то, что от пользователей скрыты различия между компьютерами и способы связи между ними. То же самое относится и к внешней организации распределенных систем. Другой важной характеристикой таких систем является способ, при помощи которого пользователи и приложения единообразно работают в распределенных системах, независимо от того, где и когда происходит их взаимодействие [1]. Сложность, разнородность, неравномерная загрузка узлов в распределенных системах, все эти свойства приводят к тому, что практически невозможно провести адекватное аналитическое моделирование. В связи с этим для анализа распределенных систем был выбран метод имитационного моделирования и среда AnyLogic, использующая многоагентный подход.

Имитационное моделирование — разработка компьютерных моделей и постановка экспериментов на них. Целью моделирования в конечном счете является принятие обоснованных, целесообразных управленческих решений. Компьютерное моделирование становится сегодня обязательным этапом в принятии ответственных решений во всех областях деятельности человека в связи с усложнением систем, в которых человек должен действовать и которыми он должен управлять. Знание принципов и возможностей имитационного моделирования, умение создавать и применять модели являются необходимыми требованиями к инженеру, менеджеру, бизнес-аналитику [2]. Преимущества имитационного моделирования заключается в возможности выделить аналоги устройств и элементов реальной системы, условия их функционирования. Программа имитации воспроизводит состав исследуемой системы, процессы функционирования системы во

времени, при этом имитируются элементарные явления, составляющие процесс с сохранением их логической структуры и последовательности протекания во времени.

Программа, реализующая аналитическую модель, не содержит аналогов реальной системы, так как построение такой программы определяется не исследуемой системой, а формальной аналитической моделью. Если аналитическая модель есть результат упрощений и существенных ограничений, то имитационная модель позволяет достаточно просто учитывать такие факторы как наличие непрерывных и дискретных элементов, нелинейные характеристики, многокомпонентность [3].

2. Описание концептуальной модели

Для концептуального описания распределенной файловой системы в работе используется нотация структурно-функционального моделирования IDEF0 (*Integrated DEFinition*). Нулевой уровень декомпозиции модели показывает входные данные, управляющие методы, исполнительные механизмы и результат работы, и, как правило, содержит только один блок (см. рисунок 1).



Рис. 1. Нулевой уровень концептуального описания модели распределенной файловой системы в нотации IDEF0

Далее, согласно стандарту IDEF0, производится декомпозиция каркасной диаграммы A0, на функциональные модули, как правило это 3-6 модулей. На первом уровне концептуальной модели (рисунок 2) показаны основные функциональные блоки, которые будут использованы при разработке в имитационной модели. На этом этапе происходит инициализация пользовательских файлов и их свойств (объем, время жизни, частота обращения); инициализация распределенных хранилищ данных (объем, скорость обработки запросов на чтение (запись), адрес узла расположения и т.п.); выбор стратегии размещения файлов в распределенной системе (самое быстрое размещение, эффективное с точки зрения времени чтения/записи блоков информации; размещение элементов в памяти по узлам хранения); непосредственно реализация выбранной эффективной стратегии и сбор статистических данных о параметрах распределения файлов.

Блоки уровня A1-A4, в свою очередь, могут быть декомпозированы на отдельные функциональные процедуры: генерация параметров пользовательских файлов; реализация алгоритмов распределения файлов по узлам системы хранения; обработка результатов моделирования и вывод графической информации (Рисунок 2).

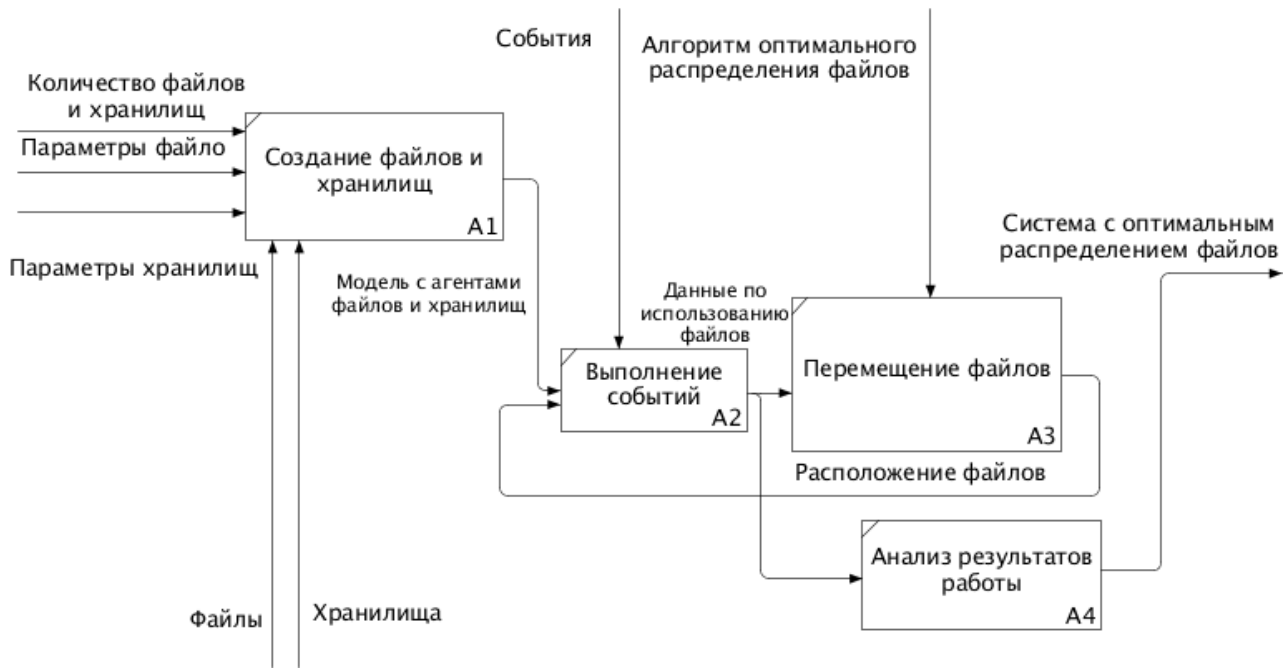


Рис. 2. Первый уровень концептуальной модели распределенной файловой системы в нотации IDEF0

Далее рассмотрим описание имитационной модели распределения файлов.

3. Описание имитационной модели

Основными элементами в моделируемой системе являются файлы, хранилища данных, базы данных и события размещения. В среде имитационного моделирования они описываются как обособленные агенты со своим набором параметров (полей описания). Описание файлов включает следующие поля:

- *fileId* – индекс пользовательского файла в файловой системе;
- *fileSize* – размер пользовательского файла;
- *isReading* – пользовательский файл используется в операции чтения;
- *isCopy* – пользовательский файл используется в операции копирования.

Пользовательские файлы размещаются по узлам в распределенной системе хранения.

Описание хранилища (алгоритм организации хранилища приведен на рисунке 3) включает следующие поля:

- *repositoryId* – идентификатор хранилища в распределенной файловой системе;
- *repositorySize* – размер хранилища (репозитория);
- *storageCost* – стоимость хранения единицы информации в хранилище;
- *cacheSize* – размер буфера временной памяти;
- *averageTimeToRepository* – среднее время обращения к хранилищу;
- *files* – файлы, размещенные в хранилище;
- *events* – список событий для хранилища.
- *cache* – временная память.

Параметры для сбора статистики по использованным файлам (пример диаграммы приведен на рисунке 4):

- *intensity* – интенсивности использования файлов;
- *useFileInRepository* – количество файлов, используемых из хранилища;
- *useFileInCache* – количество файлов, повторно использованных из временной памяти;
- *loadFileInCache* – количество файлов загруженных во временную память.

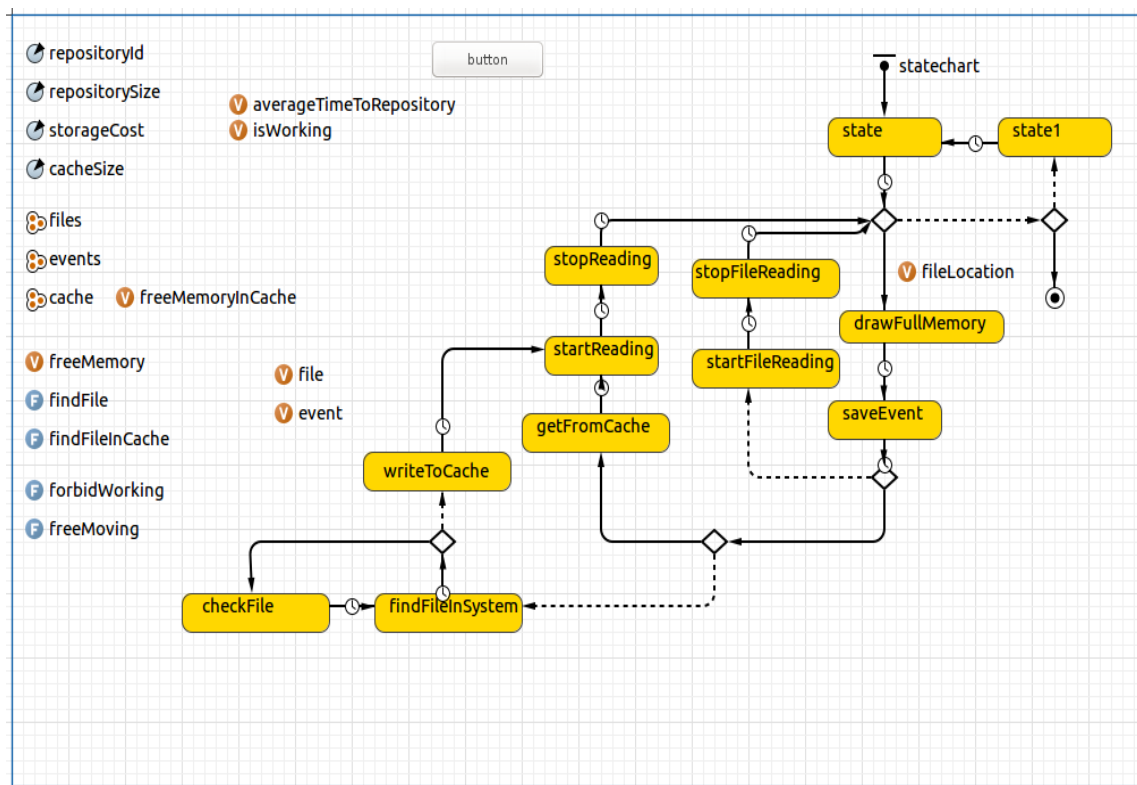
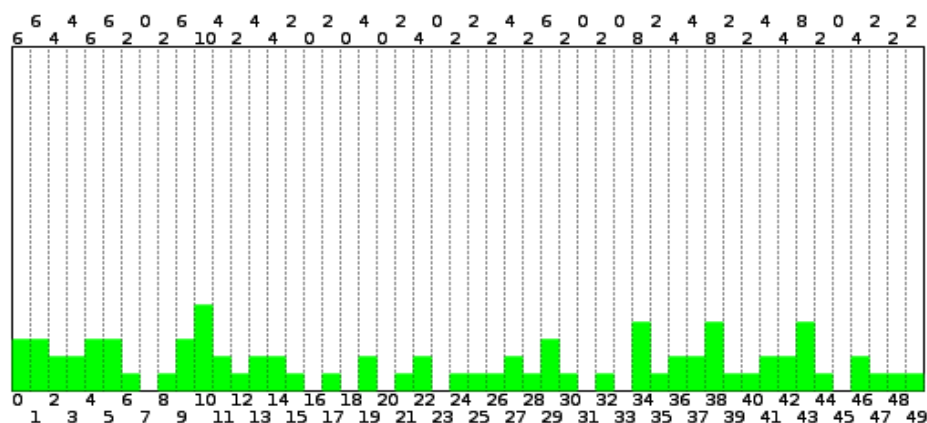


Рис. 3. Алгоритм организации распределенного хранилища

Рис. 4. Диаграмма, иллюстрирующая результат размещения файлов по хранилищам

V intensity
 [50]
 F drawColumn



V useFileInRepository
 54
 V useFileInCache
 3
 V loadFileInCache
 103

Входными данными для модели распределенной файловой системы являются количество и параметры файлов и хранилищ. Число хранилищ в приведенном эксперименте варьируется от 2 до 10 для удобства визуализации, количество файлов находится в диапазоне

от 10 до 1000. Параметры файлов и репозитариев заполняются информацией из встроенной базы данных среды Anylogic. Для генерации параметров файлов и хранилищ использовался нормальный закон распределения.

База данных для файлов состоит из следующих полей (рисунок 5): *file_id* – идентификационный номер файла; *repository_number* – идентификационный номер хранилища, в котором первично располагается файл; *file_size* – размер файла.

	file_id	repository_number	file_size
1	0	1	1
2	1	0	3
3	2	1	8
4	3	4	3
5	4	0	7
6	5	0	10
7	6	1	5
8	7	2	10
9	8	2	10
10	9	3	9

Рис. 5. Формат базы данных файлов

База данных для хранилища состоит из следующих полей (рисунок 6): *repository_id* – идентификационный номер хранилища; *repository_memory* – размер хранилища; *storage_coast* – стоимость хранения одной единицы информации; *average_time* – среднее время обращения к файлу; *cache_size* – размер временной памяти.

События размещения состоят из двух полей: номер хранилища и номер файла, который требуется хранилищу.

	repository_id	repository_memory	storage_coast	avarage_time	cache_size
1	0	60	3	1	20
2	1	60	1	1	20
3	2	80	4	3	30
4	3	100	2	2	30
5	4	35	2	2	20
6	5	100	1	4	30
7	6	200	1	1	35
8	7	300	1	3	100
9	8	125	2	1	50
10	9	250	3	4	40

Рис. 6. Формат базы данных хранилищ

Предлагаемый в работе подход может быть использован для принятия решения по выбору наиболее эффективного по заданным критериям варианта размещения файлов в узлах распределенной системы хранения. Предлагаемая концепция описания и имитационная модель могут применяться при решении задач анализа и оптимизации функционирования реальных распределенных файловых систем хранения. Для использования модели необходимо собрать достаточный объем статистических данных об обрабатываемых файлах и их параметрах.

Библиографический список:

1. Таненбаум Э. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. Ван Стеен. — СПб.: Питер, 2003. — 877 с.
2. Киселева М. В. К44 Имитационное моделирование систем в среде AnyLogic : учебно-методическое пособие / М. В. Киселёва. Екатеринбург : УГТУ - УПИ, 2009. — 88 с.
3. Сравнение аналитической и имитационной модели [Электронный ресурс] — Образовательный блог — всё для учебы - Электрон. данные - URL: <https://all4study.ru/modelirovanie/sravnienie-analiticheskoy-i-imitacionnoj-modeli.html> (дата обращения 27.03.2018).

СЕКЦИЯ 2

Робототехника. Технологии искусственного интеллекта



УДК 004.8

И.А. Захарченко, студент 4-го курса**В.В. Хлебникова, канд. экон. наук, старший преподаватель***Севастопольский государственный университет***ВОЗМОЖНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА**

Аннотация: В статье проведен обзор возможностей применения искусственного интеллекта для упрощения жизнедеятельности человека путем внедрения различных технологий в повседневную жизнь, а также рассмотрен один из видов искусственного интеллекта, такой как квантовый компьютер: его области применения и потенциал развития.

Ключевые слова: искусственный интеллект, квантовый компьютер, мониторинг, технологии.

Annotation: The article reviews the possibilities of using artificial intelligence to simplify human life through the introduction of various technologies into everyday life, and also considered one of the types of artificial intelligence, such as a quantum computer: its scope and development potential.

Key words: artificial intelligence, quantum computer, monitoring, technology.

Актуальность данной статьи заключается в том, что в настоящее время автоматизации подвергаются множество сфер окружающей нас жизни и бизнеса. Робототехника и искусственный интеллект используется практически во всех отраслях экономики и жизни социума. В 21 веке искусственный интеллект неоднократно спасал людям жизни. К примеру, «умная машины Tesla» с водителем, которая отвезла смертельно больного водителя в больницу самостоятельно, основываясь лишь на встроенных датчиках.

Целью данной статьи является обзор возможностей искусственного интеллекта и квантовых компьютеров в различных отраслях хозяйственной деятельности и областях жизнедеятельности человека.

Самыми востребованными областями науки и техники являются робототехника и искусственный интеллект. Попробуем вспомнить, что бы вы сказали, увидев 20 лет назад у человека мобильный телефон? Да, реакция была удивительна, некоторые люди даже не могли осознать зачем необходимо это странное устройство. Примерно 10 лет назад подобная реакция наблюдалась у людей впервые увидевших живого робота на научной выставке, у друга гения в гараже, либо же увидев банальную детскую игрушку, которая чем-то схожа с роботом, за счет того, что она могла ездить по дому, не врезаясь в стены или же проходить различные препятствия без помощи своего хозяина благодаря десяткам различных датчиков. Что же мы наблюдаем сейчас?

В 21 веке никто не удивляется домашнему роботу пылесосу, экзоскелету для военных нужд либо гражданских лиц, который помогает людям утратившим возможность ходить с помощью своих сил; роботы-песики преследуют нас повсюду; различные роботы-испытатели, механизированные заводы с присутствием доли искусственного интеллекта. Не говоря уже о том, что одним из интереснейших занятий в Японии являются бои роботов.

В наше время никого не удивит телефон с поддержкой нейронных сетей. Нейронные сети начали развиваться с бешеной скоростью в последние 2-3 года и дело дошло до того, что топовые флагманы поддерживают функцию самообучения за счет появления нейронных сетей. Наше время стало настолько удивительным, с помощью 3D-принтера люди могут «напечатать» человеческое сердце из клеточной субстанции и затем имплантировать его в человека, чтобы спасти его жизнь. Следуя из этого можно задуматься о том, через сколько лет мы сможем вживую увидеть робота из человеческой плоти и настоящими органами с присутствием настоящей серой субстанции в голове. Фантастика? Может быть. Но не забываем и о том, что уже в наше время существует искусственный интеллект, который может развиваться на фоне человеческого поведения и подстраиваться под его обыденную жизнь, чтобы ее облегчить.

К сожалению, ныне существующий искусственный интеллект способен обучаться только на наших повадках из разряда показа любимых приложений в панели уведомлений, или автонастройки камеры для различных функций, или же самообучения голосового ассистента.

В ближайшее время ожидается большой скачок в развитии искусственного интеллекта благодаря квантовому компьютеру. Искусственный интеллект основывается на возможности обучения в процессе своего жизненного цикла, иначе говоря на основе обратной связи постепенно обзаводиться машинным интеллектом, то есть постепенно и самостоятельно обучается выполнению задач определенного типа.

Искусственный интеллект на базе квантового компьютера непременно станет прорывом 21 века. Многие компании гиганты уже используют квантовые компьютеры Beta версии для различных целей. Например, компания D-Wave использует свой квантовый компьютер для испытаний программного обеспечения на автомобилях в целях автопилота [1,2,6].

Одним из областей использования искусственного интеллекта это помощь при мониторинге за природными ресурсами. С каждым годом все больше истощаются природные ресурсы, поэтому возникает необходимость применения технологии искусственного интеллекта. Существующие дешевые, широко распространенные сенсорные сети, IoT, улучшения в вычислительной мощности, алгоритмы с открытым исходным кодом - все это позволяет нам эффективнее наблюдать за океаном и лесами. Поэтому системы искусственного интеллекта, умеющие чувствовать, думать, обучаться, могут модернизировать попытки по сохранению ресурсов, помочь людям бороться с изменением климата и эффективнее использовать энергию.

Еще одна сфера применения это солнечная энергия. Преимущество солнечных, ветряных и других возобновляемых источников энергии состоит в том, что они являются повсеместными и не содержат углерод. Они могут быть расположены в селах и городах, таким образом, энергия становится ближе к каждому, кто в ней нуждается. Задача заключается в том, чтобы объединить эти несопоставимые источники в единое функциональное целое. Для этого и нужны автоматизированные системы.

«Когда есть сложная система с таким большим количеством возобновляемых источников энергии, нужно научить их взаимодействовать друг с другом, чтобы оптимизировать нагрузку. Это невозможно без искусственного интеллекта. Он позволит объединить эти источники и организовать систему перепродажи и обмена энергией внутри сообщества», – объясняет Гервейер [3].

Благодаря искусственному интеллекту система водоснабжения станет более децентрализованной, ей будут управлять сенсоры и новые технологии типа блокчейна. Смарт-контракты – компьютерные алгоритмы, предназначенные для заключения и поддержания коммерческих контрактов в технологии блокчейн – позволят быстро торговать активами, включая права на водные ресурсы.

Моделирование будущих погодных явлений и климата означает решение таких сложных физических уравнений, как гидродинамика атмосферы и океанов. Климатологи, выполняя вычисления, всегда полагались на суперкомпьютеры. Однако на самом деле во всем мире существует всего лишь несколько настоящих суперкомпьютеров, и доступ к ним очень ограничен.

Искусственный интеллект будет сочетать математические модели атмосферы и вспомогательные данные для составления более точных прогнозов. «Когда-нибудь мы будем выполнять моделирования на домашних компьютерах. Мы сможем прогнозировать ветряные бури, например. Через десять лет значительно улучшится прогнозирование погоды и влияние климата», – говорит Гервейер [3].

Мы постоянно сталкиваемся с проблемами незаконной вырубки лесов и рыбной ловли. Для их решения необходимы лучшие системы отслеживания. Информация, предоставляемая спутниками и беспилотными подводными судами, конечно, помогает определять незаконное обращение с природными ресурсами. Однако искусственный интеллект сможет анализировать эти данные и делать их полезными. Цифровая геопространственная платформа и технологии виртуального моделирования позволят

контролировать и управлять экологическими системами планеты в масштабе и скорости, которые ранее были для нас недоступны. Новые возможности обработки данных позволят властям и даже общественности следить за ресурсами Земли в режиме реального времени.

Стихийные бедствия всегда несут за собой огромные потери и ущерб для тысяч людей. Проблема заключается в том, что на сегодняшний день плохо развита система предупреждения и реагирования на них.

В мире остается огромный нетронутый простор для инноваций. Искусственный интеллект и системная аналитика могут помочь раскрыть биологические и биомиметические возможности. Ученые уже начали работать над природным эквивалентом проекта «Геном человека», чтобы определить последовательности ДНК всех живых существ. Например, Amazon Third Way разрабатывает проект под названием «Банк кодов Земли», преследуя две цели: первая – сделать потенциальные открытия – например, изобрести лекарство от давления на основе яда гадюки. Вторая – записать происхождение объектов интеллектуальной собственности в области биологии, чтобы местные люди смогли получить пользу от последующих открытий.

Дело не только в том, чтобы разгадать генетические коды, но еще и в том, чтобы принимать решения, отталкиваясь от них. Каталогизация этой информации может быть полезной для фармацевтических компаний, но при этом ее использовании она еще и оборачивается выгодой для общества.

Ярким примером искусственного интеллекта является квантовый компьютер. Финансовое моделирование, прогнозирование погоды, сложнейшие математические расчеты – все это возможно благодаря квантовому компьютеру.

Квантовые компьютеры – особые вычислительные устройства, мощность которых возрастает экспоненциальным образом благодаря использованию в их работе законов квантовой механики. Данные устройства состоят из кубитов – ячеек памяти и одновременно примитивных вычислительных модулей, которые могут хранить в себе спектр значений между нулем и единицей.

Можно привести довольно простой пример его работы на стрельбе из лука. Для попадания в цель на расстоянии 10 метров необходимо понять, как высоко необходимо целиться в безветренную погоду. Подобные расчеты можно произвести и на обычном привычном для нас компьютере, но есть небольшая загвоздка. Обычный компьютер будет рассчитывать каждую траекторию поэтапно (1,2,3,4,5 см и т.д.), в свою очередь квантовый компьютер произведет весь расчет одновременно и выдаст нам ответ. Исходя из этого нацеливаясь на будущее, благодаря созданию искусственного интеллекта и квантового компьютера можно будет намного раньше диагностировать рак, узнать существует ли разумная жизнь во вселенной и тому подобные весьма невероятные вещи. Если учесть темпы развития научно-технического прогресса за последний век, появление интернета, компьютеров, электричества или даже электромобиля, то стоит задуматься насколько странным звучит искусственный интеллект в 21 веке [6].

Весь мир занимается созданием квантового компьютера, его появление будет способствовать развитию всего мира в целом. Прорывы в технологиях и новости о квантовых компьютерах появляются постоянно, некоторые компании заверяют нас, что 100 Q-битный квантовый компьютер можно будет назвать полноценным, в данный же момент уже существует 84 Q-битный компьютер, созданный компаниями D-Wave и Google, к сожалению, он находится на самых ранних стадиях развития, однако он будет работать как квантовый, ведь это лишь вопрос времени. Самой актуальной проблемой данного компьютеров являются ошибки, 1 из 100 операций обычно выполняется неверно, кажется не таким уж и плохим результатом, однако не стоит забывать, что на практике уже доказано, квантовый компьютер созданный компанией D-Wave в 100 млн. раз быстрее обычного, соответственно 1/100 ошибок будут являться 1 миллионом ошибок, а это уже довольно плохо.

В настоящее время помимо строительства универсального квантового компьютера IBM задается целью создать сообщество разработчиков и пользователей [4].

Квантовый компьютер уже используют для развлечений и уже написано для него несколько простейших игр. К примеру, уже можно сыграть с искусственным интеллектом в игру камень ножницы бумага на 5 Q-битном компьютере [5].

Однако существуют и недостатки, из-за появления универсального квантового компьютера можно будет легко взломать абсолютно любые ключи шифрования, будь то банковские операции или же переводы на государственном уровне.

В связи с вышеизложенным, можно отметить ожидаемость внедрения искусственного интеллекта, в том числе квантовых компьютеров в различные области деятельности субъектов бизнеса и социума. Подводя итоги, можно сказать, что после создания полноценно функционирующего квантового компьютера в мире будет царить гармония и закончатся войны, ведь мы сможем получить ответы на всевозможные загадки вселенной.

Перспективы дальнейших исследований. Разработка рекомендаций по применению искусственного интеллекта и его разновидностей в различных областях человеческой жизнедеятельности.

Библиографический список

1. Бессмертный, И. А. Системы искусственного интеллекта: учебное пособие для академического бакалавриата / И. А. Бессмертный. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2018. — 130 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-534-02747-1. — Режим доступа: www.biblio-online.ru/book/A1B77687-B5A6-4938-9C0E-F6288FDA143B.
2. Иванов, В. М. Интеллектуальные системы: учебное пособие для вузов / В. М. Иванов; под науч. ред. А. Н. Сесекина. — М.: Издательство Юрайт, 2018. — 91 с. — (Серия: Университеты России). — ISBN 978-5-534-00551-6. — Режим доступа: www.biblio-online.ru/book/39721453-6D87-4D55-8F03-7487C942FF8B.
3. Как искусственный интеллект поможет спасти планету. - Режим доступа: <https://rb.ru/story/ai-save-the-planet/> (дата публикации 16.02.2018)
4. Квантовый шанс: как IBM строит экосистему для компьютера будущего. - Режим доступа: <https://www.rbc.ru/magazine/2018/01/5a3940b09a794764894cb71c> (дата публикации 12.01.2018)
5. Раскрыты секреты работы первого в мире 51-кубитного квантового компьютера. - Режим доступа: <https://mir24.tv/news/16280062/raskryty-sekrety-raboty-pervogo-v-mire-51-kubitnogo-kvantovogo-kompyutera> (дата публикации 30.11.2017)
6. Шесть примеров, когда квантовые компьютеры нам очень помогут. – Режим доступа: <https://hi-news.ru/computers/shest-primerov-kogda-kvantovye-kompyutery-nam-ochen-pomogut.html> (дата публикации 04.07.2017)

УДК 65.011.54

И.С.Пахольчук, студентка 4-го курса

Б.А.Букач, канд. экон.наук, доцент

Севастопольский государственный университет

ЗАМЕНИТ ЛИ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ ТОП-МЕНЕДЖЕРОВ

Аннотация: Искусственный интеллект (ИИ) активно развивается, он уже смог заменить многие профессии. Кассиры, водители, бухгалтеры, сметчики – эти и другие операции уже выполняются машинами. Предполагаемое влияние технологий ИИ на бизнес будет заключаться в увеличении производительности труда на 40% за счет фундаментальных изменений в схеме выполнения той или иной работы и усиления роли людей в создании условий для роста бизнеса. Но смогут ли ИИ заменить руководителей компании? Ведь если к управлению предприятием придет робот, можно предположить, что улучшится вся работа предприятия.

Ключевые слова: менеджмент, искусственный интеллект

Abstract: Artificial intelligence is actively developing, it has already been able to replace many professions. Cashiers, drivers, accountants, estimates - these and other operations are already performed by machines. The expected impact of AI technologies on business will be to increase labor productivity by 40% due to fundamental changes in the scheme of performing a particular job and strengthening the role of people in creating conditions for business growth. But can AI replace the company's leaders? After all, if the robot comes to the management of the enterprise, it can be assumed that the whole work of the enterprise will improve.

Keywords: management, artificial intelligence

В преддверии второго десятилетия текущего столетия тема футурологии актуальна как никогда. Специалисты в области искусственного интеллекта прогнозируют, что роботы заменят человека почти во всех отраслях деятельности, даже в управлении целым предприятием.

Множественные дискуссии о том, какой будет структура экономики, и что будет основным драйвером экономического развития середины XXI века?

Какие профессии исчезнут, а какие наоборот возникнут? Какими знаниями, навыками и умениями нужно обладать, чтобы быть востребованным на рынке труда? А также о том, какие бизнес-функции и технологии будут автоматизированны, или их заменит искусственный интеллект?

По результатам исследования Accenture, искусственный интеллект (ИИ) может удвоить годовые темпы экономического роста к 2035 году, изменив характер выполняемой работы и создав новую модель взаимодействия человека и машины [1].

Предполагаемое влияние технологий ИИ на бизнес будет заключаться в увеличении производительности труда на 40% за счет фундаментальных изменений в схеме выполнения той или иной работы и усиления роли людей в создании условий для роста бизнеса.

Что это означает для топ-менеджеров, и как будет выглядеть лидерство в будущем, принимая во внимание растущую роль ИИ?

Джек Ма, руководитель китайской компании Alibaba, работающей в сфере интернет-коммерции, считает, что появятся робото-руководители и даже робото-СЕО — по его словам, для этого есть все основания, поскольку роботы более объективны и менее чувствительны по сравнению с людьми [2].

Применение искусственного интеллекта привело к постепенному уничтожению целого ряда профессий на рынке труда.

Со временем исчезнут кассиры, водители, бухгалтеры, сметчики – эти и другие операции уже выполняются машинами.

Предполагается, что полностью исчезнет профессия дальнобойщиков благодаря запуску беспилотных грузовиков, не нужна будет вся соответствующая инфраструктура[1].

Это приведет к резкому росту безработицы, а также к повышению требований работодателей.

Но смогут ли ИИ заменить руководителей компании? Ведь если к правлению предприятием придет робот, можно предположить, что улучшится вся работа предприятия:

- все пути и решения будут четко рассчитаны;
- будет абсолютная точность во всем;

- будет налажена финансовая сфера предприятия;
- можно будет быть уверенными в качестве выпускаемой продукции;

Однако, несмотря на то, что машины все больше заменяют человека при выполнении повторяющихся задач, вряд ли можно заменить творческие способности и социальное поведение, которые необходимы менеджеру.

Также всегда найдутся задачи, которые робот выполнить не сможет, для него они могут оказаться слишком сложными, невыполнимыми или вообще «незапрограммированными», например:

- робот никогда не сможет продумать новые пути продвижения компании;
- робот не сможет продумать правильную стратегию предприятия;
- робот не сможет поддерживать хороший психологический климат в коллективе;
- робот – это точная «машина», которая никогда не будет учитывать человеческий фактор;

- если руководителем будет «машина», исчезнет человеческая коммуникация.

Таким образом, можно сделать вывод, что роботы могут стать лишь хорошими помощниками в управлении, но финальное решение должен принимать человек.

Как же в таком случае управленцу использовать искусственный интеллект?!

Внедрение ИИ открывает новые перспективы для усовершенствования деятельности и повышения ее продуктивности.

Машина, действительно, не допускает ошибок, исключается риск сбоев технологических процессов из-за человеческого фактора.

Сложные программы обрабатывают большие объемы информации в доли секунды и учатся запоминать и применять полученные данные.

Применение искусственного интеллекта позволяет сократить трудовые и временные затраты на выполнение простых монотонных действий. Современные машины умеют распознавать препятствия на пути и мгновенно реагировать на внештатные ситуации, принимая решения по алгоритму.

Топ-менеджеры должны приложить усилия, чтобы определить и понять, как будут развиваться технологии в будущем, и какое влияние они окажут на бизнес.

Но недостаточно просто разбираться в технологиях, важно уметь извлекать пользу из ИИ, поскольку, в большинстве случаев, можно получить преимущества не в результате сокращения затрат на оплату труда, а в результате повышения эффективности за счет уменьшения количества ошибок, высокой производительности и улучшения качества[2].

Топ-менеджерам будет необходимо уделять особое внимание персоналу и актуальным организационным вопросам, поскольку ИИ фундаментально изменит бизнес-процессы и, что более важно, культуру организации.

Бизнес-лидеры будут вынуждены «отпускать» то, что противоречит веку организационных изменений.

Умение определить области, наиболее подверженные ИИ с технической точки зрения, позволит руководителям пересмотреть то, каким образом сотрудники вовлекаются в рабочий процесс, и то, как можно улучшить взаимодействие людей, команд и проектов с помощью цифровых профессиональных сетей.

В страховании искусственный интеллект может применяться для управления документооборотом, обработки клиентских данных, селекции рисков, борьбы с мошенничеством, определения персонализированных страховых услуг и распределения страховых выплат [4].

В ритейле основным направлением для внедрения искусственного интеллекта связано должно быть с обслуживанием клиентов, оптимизацией логистики, инвентаризацией складских запасов, снижением затрат и прогнозированием спроса [4].

Финансовая отрасль. Здесь технологии позволят снизить издержки, минимизировать риски, проверять заемщиков, оценивать их платежеспособность, осуществлять прогнозирование и т.д [3].

Маркетинг. Применение искусственного интеллекта позволяет собирать и быстро анализировать информацию о тысячах пользователей для продвижения товаров и услуг [1].

И последнее, но не менее важное — ИИ также может вдохновить топ-менеджеров на определение объема их собственных задач, которые могут быть решены лучше и эффективнее за счет машин, что позволит сосредоточиться на основных функциях, с которыми не справится робот или алгоритм.

Библиографический список

1. Применение искусственного интеллекта в бизнесе// 19.02.2018 [Электронный ресурс] – Режим доступа: <http://delonovosti.ru/business/4176-primenenie-iskusstvennogo-intellekta-v-biznese-.html>
2. Боги из машины// 04.12.2017 [Электронный ресурс] – Режим доступа: <http://new.forbes.ru/karera-i-svoy-biznes/353763-bogi-iz-mashiny-zamenit-li-top-menedzherov-iskusstvennyy-intellekt>
3. Искусственный интеллект VS финансовый менеджмент // Борис Евдокимов, управляющий партнер Financial & Business Intelligence (F&BI)// [Электронный ресурс] – Режим доступа: <http://www.sbr.in.ua/?p=620>
4. Гонка технологий. Как искусственный интеллект помогает бизнесу// 20.12.2017 [Электронный ресурс] – Режим доступа: <http://new.forbes.ru/tehnologii/354727-gonka-tehnologiy-kak-iskusstvennyy-intellekt-pomogaet-biznesu>

СЕКЦИЯ 3
Интеллектуальный анализ данных.
Технологии Big Data. Системы
поддержки принятия решений

УДК 004.822

И.А. Безпалый, студент 4-го курса

Е.Н. Машенко, к.т.н., доцент

Севастопольский государственный университет

КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ОБРАБОТКИ ТЕКСТОВЫХ СООБЩЕНИЙ

Аннотация: статья рассказывает о полезной программе, способной коммуницировать с человеком, при этом упрощая поиск информации.

Ключевые слова: система, обработка данных, имитация, чат-бот, концепт.

Annotation: this article about useful program capable of communicating with a person at the same time making it easier to search for information

Key words: system, data processing, imitation, chat-bot, concept, training

Введение. Одним из самых известных и естественных для человека способов получения и донесения информации является диалог, т.е. совокупность слов и предложений, представляющих собой логически объединенный текст. Такая информация хранится в виде неструктурированных данных, обработка которых статическими методами невозможна без потери понимания семантики текста. Для анализа текстовых сообщений применяются относительно недавно изобретенные методы обработки естественного языка (ОЕЯ).

Рассмотрим предметную область обработки естественного языка, ее задачи и ограничения [1].

Обеспечение взаимодействия с компьютером на естественном языке (ЕЯ) является важнейшей задачей исследований по искусственному интеллекту (ИИ) [3]. Базы данных, пакеты прикладных программ и экспертные системы, основанные на ИИ, требуют оснащения их гибким интерфейсом для многочисленных пользователей, не желающих общаться с компьютером на искусственном языке. В то время как многие фундаментальные проблемы в области обработки ЕЯ (Natural Language Processing, NLP) еще не решены, прикладные системы могут оснащаться интерфейсом, понимающем ЕЯ при определенных ограничениях.

Обработка естественного языка – это формулирование и исследование компьютерно-эффективных механизмов для обеспечения коммуникации с ЭВМ на ЕЯ.

Существуют два вида и, следовательно, две концепции обработки естественного языка:

- для отдельных предложений;
- для ведения интерактивного диалога.

Объектами исследований являются:

- собственно естественные языки;
- использование ЕЯ как в коммуникации между людьми, так и в коммуникации человека с ЭВМ.

Задача исследований - создание компьютерно-эффективных моделей коммуникации на ЕЯ. Именно такая постановка задачи отличает NLP от задач традиционной лингвистики и других дисциплин, изучающих ЕЯ, и позволяет отнести ее к области ИИ. Проблемой NLP занимаются две дисциплины: лингвистика и когнитивная психология.

Основной проблемой NLP является языковая неоднозначность. Существуют разные виды неоднозначности:

- *Синтаксическая (структурная) неоднозначность*: во фразе «Мой плащ» для ЭВМ неясно, идет ли речь об принадлежности плаща кому-то, или о мытье плаща, т.е. является ли слово «мой» глаголом или местоимением.
- *Смысловая неоднозначность*: во фразе «Его кисть была сломана» слово «кисть» может означать как часть тела, так и инструмент для рисования.

- *Падежная неоднозначность*: предлог «в» в предложениях «Он убежал в 4 часа./Он убежал в дом» обозначает либо время, либо место, т.е. представлены совершенно различные отношения. Или, например, слово «кошки», может обозначать как и множественное, так и единичное число (родительный падеж);
- *Референциальная неоднозначность*: для системы, не обладающей знаниями о реальном мире, будет затруднительно определить, с каким словом – «стол» или «торт» - соотносится местоимение его во фразе «Я взял торт со стола и съел его».
- *Литерация (Literalness)*: в диалоге «Не мог бы ты открыть окно? – Мне холодно» и просьба, и ответ выражены нестандартным способом. В других обстоятельствах на вопрос может быть получен прямой ответ «да»/«нет», но в данном случае в вопросе имплицитно выражена просьба открыть окно.

Определение и сфера применения интеллектуальной системы обработки текстовых сообщений.

Интеллектуальная система обработки текстовых сообщений (ИСОТС) – это разновидность интеллектуальной системы, представляющая собой комплекс программных и лингвистических средств для реализации следующей задачи: осуществление поддержки деятельности человека в режиме диалога на естественном языке.

За последнее время спектр применения подобных систем невероятно расширился. Примерный список сфер применения:

- в системе образования [2]:
 - Облегчение изучения языков;
 - Административная поддержка учителей;
 - Вовлечение студентов в работу;
 - Развитие критического мышления;
 - Проверка и закрепление знаний в любой области;
 - Универсальный учитель.
- в бизнес структурах:
 - Рекомендательная система;
 - Система поддержки пользователей;
 - Поиск вакансий и людей на вакансии;
 - Слежение за изменениями цен на рынке;
- в медицинских структурах:
 - Первичное диагностирование состояния здоровья и рекомендации по улучшению здоровья;
- в информационных структурах:
 - Отслеживание изменений в информационном пространстве.

Концептуальная модель ИСОТС.

В данной статье ставится задача разработки концептуальной модели ИСОТС со следующими функциональными возможностями:

Система должна уметь:

1. Выявлять ключевые слова (КС). Особенностью является то, что КС могут быть записаны в сообщении с ошибками, задача системы суметь выявить такие слова.
2. Обрабатывать голосовые сообщения (записывать с микрофона и преобразовывать в текст).
3. Поддерживать текстовый диалог, максимально похожий на общение с настоящим человеком:
 - а. Диалоговая система должна «обладать» примитивной индивидуальностью и эмоциями;

- б. В диалоге система должна иметь собственную цель (к примеру в начале диалога это может быть знакомство с пользователем, выяснять его потребности, интересы, возраст).
- 4. Система должна иметь способность самообучаться.
- 5. Система может выступать в роли виртуального помощника:
 - а. Выполнять поиск информации в интернете и интерпретировать полученную информацию в приемлемый для пользователя вид.
 - б. Система поиска информации должна быть способна адаптироваться под интересы пользователя, для ускорения получения интересующей пользователя информации.
 - с. Напоминать о предстоящих мероприятиях, которые указывает пользователь.

Схема процесса обработки текстового сообщения (ТС) в ИСОТС приведена на рисунке 1, структурная схема ИСОТС приведена на рисунке 2.

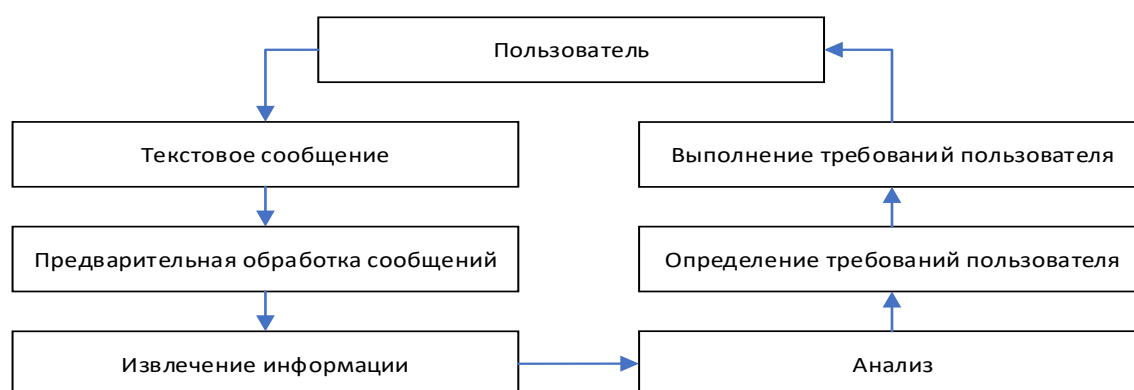


Рис.1 – Схема процесса обработки ТС.

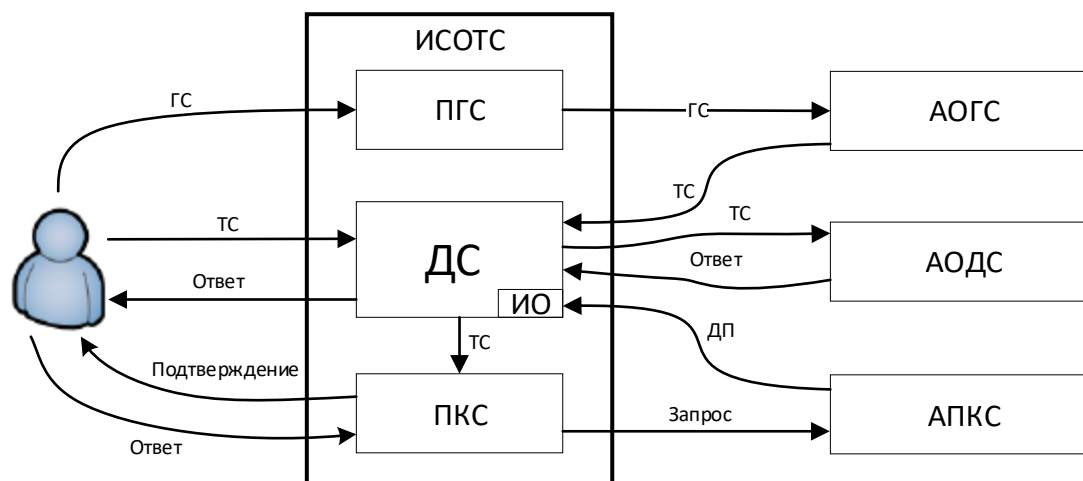


Рис.2 – Структурная схема ИСОТС (ГС – голосовое сообщение, ТС – текстовое сообщение, ПГС – проверка голосового сообщения, АОГС – Агент обработки голосового сообщения, ДС – диалоговая система, АОДС – Агент обработки диалоговых сообщений, ПКС – проверка ключевых слов, АПКС – Агент поиска по ключевым словам, ИО – информация о ключевых словах, ДП – данные поиска);

В структурной схеме приведены три внешних агента:

1. Агент обработки голосового сообщения.
2. Агент обработки диалоговых сообщений.
3. Агент поиска по ключевым словам.

Реализация ИСОТС.

Реализация ИСОТС происходит в три этапа, которые можно описать следующим образом [3]:

1. Создание системы обработки сообщений и извлечение информации. Цель этапа – формирование новых понятий и отношений между понятиями с участиями человека и без его участия.

2. Создание системы поиска. Цель этапа – помочь человеку в поиске информации, в которой могут содержаться необходимые для него данные.

3. Интерпретация новых знаний. Цель этапа – формирование новых или корректировка старых правил поиска и интерпретации информации для пользователя.

На первом этапе создается система обработки ТС для извлечения нужной информации и формирования запроса, а также формирования шаблонов для поиска новой информации.

Процесс проектирования системы извлечения знаний из ТС и формирование шаблонов для поиска состоит из следующих этапов [4]:

- Создание автоматизированной функции фильтрации, агрегации данных (обобщение данных должно выполняться с участием специалиста, при автоматическом выполнении лишь только отдельных операций обобщения);
- Автоматизация процесса обобщения данных и формирование новых запросов и отношений между запросами. Процесс выполняется автоматически. Специалист либо контролирует результаты анализа, либо проводит анализ и синтез новых запросов и отношений;
- Автоматизация всех операций анализа и синтеза запросов. Результаты могут контролироваться специалистом, но могут и выполняться автоматически.

На первом этапе проводятся следующие виды автоматизированных и автоматических работ:

- записывание новых материалов;
- формирование запросов и шаблонов;
- анализ и обобщение новых материалов.

На втором этапе создается автоматизированная система поиска текстов по шаблонам. Шаблоны могут храниться в базе или создаваться оперативно в процессе обработки ТС. На этом этапе осуществляется:

- подбор необходимой информации;
- формирование и подбор материала для пользователя.

На третьем этапе создается система интерпретации результатов анализа и синтеза новых знаний. С помощью данной системы возможно проектировать и корректировать алгоритмы, формировать новые и корректировать старые запросы, интерпретировать новые знания и предоставлять их пользователю.

Благодаря развитию информационных технологий в настоящее время создано множество сервисов и библиотек, предоставляющих инструменты, функции и модули, упрощающий разработку и проектирование ИСОТС для специалиста и расширяющие возможности ИСОТС для нужд пользователя [5].

Планируется использование следующих сервисов, подключаемых к ИСОТС:

Для обработки голосового сообщения:

Yandex SpeechKit Cloud – технология распознавания речи и синтеза речи от российской компании Яндекс. Понимает запросы на русском и турецком языках. Поддерживаются две темы запросов: «общая» (обычные запросы) и «гео» (адреса и названия организаций), причём, по словам разработчиков, точность распознавания составляет 84 % и 94 % соответственно. Скорость распознавания — 1,1 секунды.

Для обработки диалоговых сообщений:

DialogFlow(панель Speaktoit) – это комплексный пакет разработки, поддерживаемый компанией Google, используется для создания диалоговых интерфейсов для веб-сайтов, мобильных приложений, платформ обмена сообщениями и устройств IoT. Его можно использовать для создания интерфейсов, которые способны к естественным и богатым взаимодействиям между пользователями и бизнесом.

Для поиска по ключевым словам: применяется поисковая система (к примеру Google, Яндекс и т.п.), после чего результаты поиска фильтруются и выдается пользователю.

Заключение

В настоящее время повсеместно можно наблюдать, как в нашу жизнь активно входят различные системы информационных услуг. Ранее популярность систем обработки текстовых сообщений относилась в большинстве своем к ботам, используемым в рутинных задачах — сбор данных, реклама, развлекательная сфера, за счет этого наблюдался прогресс в разработках. В данный момент сфера применения ИСОТС значительно расширилась. Появилось огромное количество новых инструментов и сервисов для усовершенствования работы систем обработки текстовых сообщений. В работе предложена концептуальная модель ИСОТС, приведены стратегии и этапы разработки ИСОТС, используемые инструменты разработки.

Библиографический список

1. Обработка естественного языка [Электронный ресурс] Режим доступа <http://chernykh.net/content/view/1105/1189/> Дата доступа: 21.03.2018
2. Чат-боты в образовании: 7 идей, как их использовать [Электронный ресурс] Режим доступа <http://2017.edcrunch.ru/news/bots-in-education> Дата доступа: 21.03.2018
3. Ю.Е. Котельникова. Обработка текстовых документов и эволюция автоматизированных систем проектирования // Известия вузов. Приборостроение. Т. 53. №6. С. 21-25. [Электронный ресурс] Режим доступа <https://cyberleninka.ru/article/v/obrabotka-tekstovyh-dokumentov-i-evolyutsiya-avtomatizirovannyh-sistem-proektirovaniya> Дата доступа: 21.03.2018
4. Суркова А.С. Концептуальный анализ, принципы моделирования и оптимизация алгоритмов синтеза текстовых структур. Диссертация на соискание ученой степени доктора технических наук. Нижний Новгород, 2016. 342 с. [Электронный ресурс] Режим доступа http://www.ntnu.ru/sites/default/files/file/dissertacii/2017/surkova_a_s.pdf Дата доступа: 21.03.2018
5. Yandex.SpeechKit [Электронный ресурс] Режим доступа: <https://ru.wikipedia.org/wiki/Yandex.SpeechKit> Дата доступа: 21.03.2018

УДК 004.05

А.В. Какаев, магистрант 2 курса

В.И. Шевченко, канд. техн. наук, доцент

Севастопольский государственный университет»

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ЭФФЕКТИВНОЙ БАЛАНСИРОВКИ НАГРУЗКИ В РАСПРЕДЕЛЕННОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ

Аннотация: В работе рассматривается задача распределения запросов по узлам распределенной вычислительной системы. Рассмотрены следующие алгоритмы распределения запросов: Round Robin, Weighted Round Robin, Least Connections, Weighted Least Connections, Source IP Hash, Weighted Response Time. Проведен анализ преимуществ алгоритмов методом анализа иерархий. Даны рекомендации по применению алгоритмов.

Ключевые слова: распределенная вычислительная система, распределение запросов, балансировка нагрузки, web-кластер, nginx, метод анализа иерархий, алгоритмы балансировки нагрузки

Annotation: The paper deals with the problem of the distribution queries among the nodes of a distributed computer system. The following load balancing algorithms are considered: Round Robin, Weighted Round Robin, Least Connections, Weighted Least Connections, Source IP Hash, Weighted Response Time. Algorithms advantages are analyzing using analytic hierarchy process. Recommendations on the application of algorithms are given.

Key words: distributed processing system, load balancing, nginx, load balancing algorithms, web- cluster, analytic hierarchy process, Round Robin, Weighted Round Robin, Least Connections, Weighted Least Connections, Source IP Hash, Weighted Response Time

В работе рассматривается задача распределения поступающих запросов по узлам распределенной вычислительной системы. Распределенная вычислительная система в общем случае представляет собой программно-аппаратный комплекс осуществляющий параллельную обработку конкурирующих запросов на двух и более вычислительных узлах. Наиболее распространенным примером распределенной вычислительной системы является web-приложение работающее на web-кластере. В данном контексте задачу выбора вычислительного узла решает так называемый балансировщик нагрузки. Эта проблема достаточно хорошо известна, она рассматривалась в работах [1-3]. В работе [1] с точки зрения безопасности рекомендуется использовать алгоритм взвешенной круговой очереди (*Weighted Round Robin*). В работе [2] приведена подробная классификация стратегий балансировки нагрузки. Авторы работы [2] для балансировки нагрузки предлагают объединить классический и фрактально-синергетический подход. Автор работы [3] предлагает использовать алгоритм *feedback node* на основе протокола *HDP*. В web-сервере *Nginx* реализованы такие алгоритмы балансировки, как *Round Robin* (является алгоритмом по умолчанию), *Least Connections*, *IP Hash* [4]. Несмотря на большое число фундаментальных и прикладных решений в данной области, проблема остается актуальной в связи с большим разнообразием архитектур вычислительных систем и существенной неоднородностью потоков задач, поступающих на обработку в такие системы.

Рассмотрим концептуальное описание задачи. Общая постановка задачи выбора вычислительного узла:

Имеется N вычислительных узлов, и M запросов, которые необходимо обработать на этих узлах. Для очередного запроса $M[i]$ необходимо выбрать узел $N[i]$ таким образом, чтобы удовлетворять следующим критериям эффективности распределения запросов:

1. **Справедливость.** Необходимо гарантировать что на обработку каждого запроса были выделены вычислительные ресурсы, и не допускать ситуации, когда один запрос обрабатывается, а остальные ждут своей очереди.
2. **Равномерность.** Необходимо распределять запросы таким образом, чтобы все узлы вычислительной системы были загружены примерно одинаково.
3. **Время отклика.** Распределению запросов необходимо осуществлять за адекватное время.

4. **Время обработки.** При выборе вычислительного узла для очередного запроса нужно стремиться выбрать узел, который сможет обработать этот запрос за минимальное время.

Также для эффективного распределения запросов необходимо учитывать следующие факторы: вычислительные характеристики узлов, такие как мощность процессора, объем оперативной памяти; сложность обработки запроса; геолокация пользователя или клиента, от которого получен запрос; геолокация вычислительного узла (при условии, если узлы расположены в разных центрах обработки данных (ЦОД)).

Подсистему балансировки нагрузки можно представить в виде структурной схемы, представленной на рисунке 1:

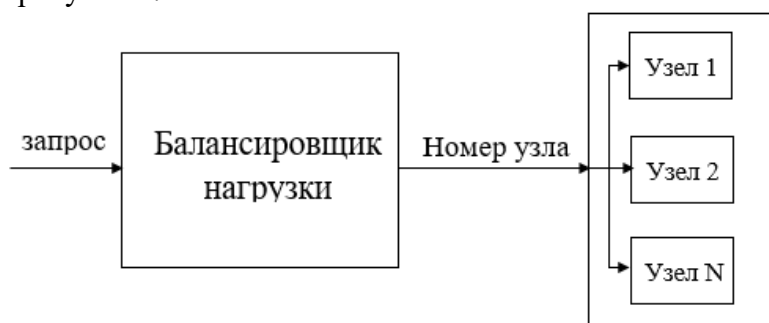


Рис. 1. Балансировщик нагрузки

Цель данной работы – провести обзор наиболее популярных алгоритмов балансировки нагрузки, на основе которого провести исследование методом анализа иерархий и выделить наиболее привлекательный, с точки зрения выбранных критериев эффективности алгоритм.

Существует множество различных алгоритмов балансировки нагрузки. Каждый из них обладает рядом преимуществ и недостатков в определенных условиях использования, рассмотрим некоторые из них.

Round-robin. Согласно данному алгоритму происходит циклический перебор вычислительных узлов для обработки каждого следующего запроса. Таким образом первый запрос ($m = 1$) назначается для обработки первому узлу ($n = 1$), второй — второму, до достижения последнего узла ($m = N$). Тогда следующий запрос ($m = N+1$) будет назначена первому узлу. Проще говоря, происходит перебор выполняющих задания объектов по циклу, или по кругу (*round*), и по достижении последнего объекта следующая задача будет также назначена первому объекту. Данный алгоритм наилучшим образом подходит в тех случаях, когда мощность всех узлов примерно одинаковая и трудоемкость задач также не сильно различается.

Weighted Round Robin. Данный алгоритм работает по такому же принципу как и Round-robin и учитывает при этом вычислительную мощность узлов. Учет мощности осуществляется за счет статических весов. Веса вычисляются для каждого узла в зависимости от мощности его процессора и объема оперативной памяти. Вычислительный узел с наибольшим весом будет обрабатывать наибольшее число запросов. Пусть $W = [1, 2, 3]$ - множество весов вычислительных узлов из множества N ($W[i]$ — вес узла $N[i]$). Тогда на узел $N[2]$ поступит в 3 раза больше запросов чем на $N[0]$. Таким образом число запросов, обрабатываемых i -тым узлом прямо пропорционально его весу. Данный алгоритм возможно применять для балансировки нагрузки на кластерах из вычислительных узлов с разной вычислительной мощностью.

Least Connection. Данный алгоритм распределяет запросы по вычислительным узлам основываясь на их текущей загрузке. Согласно этому алгоритму для обработки следующего запроса выбирается узел с наименьшим числом активных сессий в данный момент времени. Алгоритм предназначен для кластеров с одинаковыми по мощности вычислительными узлами.

Weighted Least Connection. В основе данного алгоритма лежит алгоритм *Least Connection*. При распределении запросов также учитывается мощность вычислительных узлов. Для каждого узла на основе его мощности вычисляется предельное количество одновременных соединений. При поступлении очередного запроса вычисляется оценка загрузки узла как процентное соотношение между количеством текущих соединений и предельным количеством соединений. Таким образом для обработки очередного запроса из 2 узлов с одинаковым числом соединений выбирается узел с меньшей оценкой загрузки.

Agent Based Adaptive Load Balancing. Согласно данному алгоритму происходит непрерывный мониторинг состояния всех вычислительных узлов. Узел выбирается на основании текущего состояния узлов. Данный алгоритм используется совместно с алгоритмами, рассмотренными выше. При использовании данного алгоритма совместно с *Weighted Round Robin*, вес каждого вычислительного узла динамический. Вес рассчитывается на основе текущего состояния узла. Такой подход на практике используется не часто из-за дополнительной нагрузки на вычислительную систему. Эффективность метода сильно зависит от потока запросов на вычислительную систему и характеристик производительности вычислительных узлов.

Source IP Hash. Данный алгоритм для выбора вычислительного узла использует специальный хэш ключ. Этот ключ вычисляется из IP адреса клиента и IP адреса вычислительного узла. Таким образом алгоритм гарантирует, что запросы одного и того же клиента будут обрабатываться одним узлом. Данный подход очень полезен если важно соединить клиента с его активной сессией на определенном сервере.

Weighted Response Time. При распределении запросов данный алгоритм осуществляет предварительную проверку работоспособности узлов. На каждый узел посылается тестовый запрос и измеряется время его ответа. Если один сервер отвечает медленнее чем другой, считается что он более загружен. И следующий запрос обрабатывается сервером, который ответил быстрее [5].

Если вычислительная система состоит из двух и более кластеров, расположенных в разных ЦОД, задачу распределения запросов можно разделить на два уровня. На первом уровне происходит выбор кластера, на который будет отправлен запрос. На втором уровне выбирается определенный вычислительный узел, который будет обрабатывать запрос. Несмотря на то что задачи идентичные, важность критериев эффективности решения этих задач различны.

Для систематизации экспертных знаний и анализа эффективности рассмотренных алгоритмов хорошо подходит метод анализа иерархий. Метод Анализа Иерархий (МАИ) – математический инструмент системного подхода к решению проблем принятия решений. МАИ не дает однозначного правильного решения, однако позволяет в интерактивном режиме найти такой вариант, который наилучшим образом подходит для решения конкретной задачи с заданными ограничениями [6].

Для сравнения алгоритмов были выбраны следующие критерии: K1 -справедливость; K2 - равномерность; K3 - время обработки; K4 - время отклика; K5 - масштабируемость - алгоритм должен сохранять работоспособность при росте нагрузки, изменении числа и характеристик вычислительных узлов; K6 - сложность реализации.

Согласно методу анализа иерархий в первую очередь необходимо определить относительную важность критериев по принципу дискриминации и сравнительных суждений. Матрица попарных сравнений критериев приведена в таблице 1.

Таблица 1 – Матрица попарных сравнений критериев

	K1	K2	K3	K4	K5	K6
K1 - справедливость	1	7	1	1	3	3

Таблица 1. Продолжение

	K1	K2	K3	K4	K5	K6
K2 – сложность	1/7	1	1/5	1/7	1/3	1
K3 - сокращение времени выполнения запроса	1	5	1	1/3	3	5
K4 - сокращение времени отклика	1	7	3	1	3	7
K5 - равномерная загрузка ресурсов системы	1/3	3	1/3	1/3	1	3
K6 - масштабируемость	1/3	1	1/5	1/7	1/3	1

Для определения относительной важности критериев необходимо рассчитать оценки компонент собственного вектора по следующему алгоритму:

1. Вычислить геометрическое среднее в каждой строке таблицы попарных сравнений по формуле:

$$b_i = \sqrt[n]{\prod_{k=1}^n a_{ik}}, i=1,n \quad (1)$$

Результаты вычислений по формуле (1) приведены в сводной таблице 2.

Таблица 2 – Среднее геометрическое в каждой строке

b ₁	b ₂	b ₃	b ₄	b ₅	b ₆
1,9947574	0,3328783	1,7099759	2,7589242	0,8326832	0,3833672

2. Вычислить сумму:

$$B = \sum_{i=1}^n b_i, B = 8,01258616 \quad (2)$$

3. Выполнить нормализацию вектора b:

$$x_i = \frac{b_i}{B} \quad (3)$$

Результаты вычислений по формуле (3) приведены в сводной таблице 2.

Таблица 3 - Нормализация вектора b

x ₁	x ₂	x ₃	x ₄	x ₅	x ₆
0,248953	0,0415444	0,2134112	0,3443238	0,1039219	0,0478456

4. Провести проверку нормализации, согласно формуле 4.

$$\sum_{i=1}^n x_i = 1 \quad (4)$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 0,248953009 + 0,041544423 + 0,21341124 + 0,344323808 + 0,1039219 + 0,047845621 = 1$$

После проверки нормализации можно сделать вывод, что погрешность равна 0.

После вычисления оценок значимости критериев распределения запросов второго уровня можно сделать выводы об их значимости, эти результаты приведены в таблице 4.

На этом первый этап алгоритма закончен, можно сделать выводы, что наиболее существенный вклад в анализируемой системе сделан критерием K4.

Таблица 4 – Значимость критериев

Критерий	Место	Вес	Критерий	Место	Вес
К1 - справедливость	2	0,248953009	К4 - время отклика	1	0,344323808
К2 - сложность	6	0,041544423	К5 - равномерность	4	0,1039219
К3 - время обработки	3	0,21341124	К6 - масштабируемость	5	0,047845621

Далее необходимо вычислить отношение согласованности, для подтверждения корректности суждений экспертов:

1. Вычислить сумму элементов матрицы для каждого столбца по формуле (5). Результаты вычислений приведены в таблице 5.

$$y_i = \sum_{k=1}^n a_{ik} \quad (5)$$

Таблица 5 – Суммы элементов матрицы для каждого столбца

y1	y2	y3	y4	y5	y6
3,8095238	24	5,7333333	2,952381	10,666667	20

2. Вычислить наибольшее собственное значение:

$$y_{\max} = \sum_{i=1}^n x_i y_i \quad (6)$$

$$\lambda_{\max} = 6,251004076$$

3. Вычислить индекс согласованности:

$$ИС = \frac{\lambda_{\max} - n}{n - 1}, \quad (7)$$

$$ИС = 0,050200815.$$

4. Вычислить отношение согласованности:

$$ОС = \frac{ИС}{СС} * 100\%, \quad (8)$$

$$ОС = 4\%$$

Величина отношения согласованности не превышает 10%, что является приемлемым значением и свидетельствует о корректности суждений.

По описанному выше алгоритму проводится сравнение алгоритмов по каждому из критериев. В результате попарных сравнений алгоритмов были получены суммарные количества 1-5 мест, занятых каждым из алгоритмов (Таблица 6).

Таблица 6 – Значимость критериев

	1	2	3	4	5
A1 - Round Robin	1	2	1	2	
A2 - Least Connections	2	3	1		
A3 - Weighted Least Connections	4	2			
A4 - Weighted Round Robin	4	2			
A5 - Source IP Hash	3	2	1		

Результаты вклада критериев в значение функции полезности сведены в таблице 7.

Таблица 7 – Вклад критериев в значение функции полезности

	K1	K2	K3	K4	K5	K6
A1 - Round Robin	11%	2%	17%	31%	15%	25%
A2 - Least Connections	19%	17%	21%	19%	9%	15%
A3 - Weighted Least Connections	31%	3%	20%	35%	10%	2%
A4 - Weighted Round Robin	31%	3%	20%	35%	10%	2%
A5 - Source IP Hash	17%	3%	25%	42%	12%	2%

На основе полученных результатов можно сделать вывод, что в общем случае, для обеспечения балансировки второго уровня при заданной важности критериев, наиболее подходящими альтернативами являются алгоритмы *Weighted Least Connections* и *Weighted Round Robin*. Решение задачи распределения запросов первого уровня как правило предоставляется системе доменных имен (DNS). Однако большей эффективности можно достичь используя один из выше рассмотренных алгоритмов в сочетании с постоянным мониторингом ресурсов кластеров.

При разработке системы балансировки нагрузки необходимо также учитывать, что ни один из рассмотренных алгоритмов в отдельности не обеспечивает отказоустойчивость вычислительной системы в целом. Таким образом необходимо дополнительно разрабатывать стратегию распределения запросов в случае отказа вычислительных узлов.

В рамках дальнейших исследований запланирована разработка имитационной модели балансировщика нагрузки на распределенную вычислительную систему. Имитационная модель позволит проанализировать: эффективность применения рассмотренных алгоритмов в условиях, приближенных к реальным; сравнить влияние роста нагрузки на работоспособность вычислительной системы; возможность масштабирования вычислительной системы за счет добавления новых узлов и наращивания ресурсов существующих узлов.

Библиографический список

1. Н. О. Куролесова. Выбор технологии качества обслуживания для сетей со смешанным трафиком. [Электронный ресурс]. URL: <https://cyberleninka.ru/article/v/vybor-tehnologii-kachestva-obsluzhivaniya-dlya-setey-so-smeshannym-trafikom> (дата обращения: 19.03.2018).
2. А. М. Бершадский, Л. С. Курилов, А. Г. Финогеев. Исследование стратегий балансировки нагрузки в системах распределенной обработки данных. [Электронный ресурс]. URL: <https://cyberleninka.ru/article/v/issledovanie-strategiy-balansirovki-nagruzki-v-sistemah-raspredelennoy-obrabotki-dannyh> (дата обращения: 19.03.2018).
3. М. К. Павликов. Алгоритм распределения нагрузки в программной системе, построенной на основе протокола HDP. [Электронный ресурс]. URL: <https://cyberleninka.ru/article/v/algoritm-raspredeleniya-nagruzki-v-programmnoy-sisteme-postroennoy-na-osnove-protokola-hdp> (дата обращения: 19.03.2018).
4. What is load balancing? [Электронный ресурс]. URL: <https://www.nginx.com/resources/glossary/load-balancing/> (дата обращения: 19.03.2018).
5. Thomas Alva. Load Balancing Algorithms: сайт kemp. [Электронный ресурс]. URL: <https://kemptechnologies.com/load-balancer/load-balancing-algorithms-techniques/> (дата обращения: 19.03.2018).
6. Саати Т.Л. 1989. Принятие решений. Метод анализа иерархий. М.: Радио и связь, 316 с.

**Н.А. Ковальчук, магистрант,
О.В. Ченгарь, канд. техн. наук, доцент**

Севастопольский государственный университет

ПОДХОД К ПРОЕКТИРОВАНИЮ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В ЗАДАЧАХ СОЦИАЛЬНОЙ СЛУЖБЫ

Аннотация: В статье рассматривается возможность проектирования системы поддержки принятия решений, основанной на базе знаний и опыта в социальной сфере. Целью такой системы является поиск наилучшего решения, для оказания специалистами ГБУ РК «Сакский районный ЦСССДМ» помощи людям оказавшимся в тяжелой жизненной ситуации и социально опасном положении. Проводится исследование возможности практического использования компьютерной автоматизированной системы, демонстрируется структурная схема системы поддержки принятия решения.

Ключевые слова: система поддержки принятия решений, подсистема оперативного анализа, подсистема хранения информации, база знаний.

Annotation: the article considers the possibility of designing a decision support system based on knowledge base and experience in the social field. The purpose of this system is to find the best solution for providing experts SBI RK "Saki district CSSFCY" helping people trapped in a difficult life situation and socially dangerous situation. Study the feasibility of computer automated system shows a block diagram of the decision support system solutions.

Key words: decision support system, operational analysis subsystem, storage subsystem information, knowledge base.

Современная жизнь немыслима без информационных технологий, которые играют активную роль практически во всех сферах нашей жизнедеятельности. В данной статье проведено исследование применения современных компьютерных технологий в социальной сфере и рассмотрена возможность проектирования системы поддержки принятия решений в задачах, которые стоят перед специалистами по социальной работе для оказания помощи семьям, несовершеннолетним и молодежи, оказавшимся в тяжелой жизненной ситуации [1].

Задачи Государственного бюджетного учреждения Республики Крым «Сакский районный центр социальных служб для семьи, детей и молодёжи»:

- содействие в направлении в специализированные лечебные учреждения здравоохранения лиц, нуждающихся в лечении в таких учреждениях;
- психологическое консультирование;
- содействие в оказании материальной помощи;
- помощь в оформлении документов, удостоверяющих личность;
- оказание помощи в оформлении документов на осуществление положенных мер социальной поддержки;
- оказание помощи в оформлении документов для трудоустройства;
- содействие в оформлении документов для направления детей в учреждения социального обслуживания на временное пребывание;
- содействие в решении вопросов возвращения ребенка в семью;
- содействие в решении вопроса сохранения ребенка в семье;
- содействие в организации обучения детей;
- содействие в предоставлении жилья;
- содействие в предоставлении детям, нуждающимся в оздоровлении, соответствующих услуг и направлении их на санаторно-курортное лечение;
- содействие в предоставлении временного помещения для проживания
- консультирование по социально-правовым вопросам (семейное, жилищное, трудовое законодательство, права детей). За 2017 год оказано около 6000 социальных услуг более 4000 клиентам.

Система поддержки принятия решений (СППР) – это компьютерная интерактивная система, разработанная в помощь специалисту (или руководителю) при принятии решений. СППР включают данные, и модели, чтобы помочь лицу, принимающему решения решить проблемы, особенно те, которые плохо формализованы.

Приоритет персонального компьютера состоит в его значительном быстродействии и объеме памяти, что делает его необходимым для выше указанных задач. Преимущество применения информационных технологий для решения поставленных задач:

- мгновенный доступ к информации, накопленной в базе данных компьютера или в компьютерной сети человека, принимающего решение;
- осуществление интерактивной имитации, при помощи имитационных моделей;
- использование накопленного опыта, нахождение в базах данных принятых ранее решений в ситуациях, подобных исследуемым;
- использование знаний лучших в своей области специалистов, включенных в базы знаний экспертных систем;
- представление результатов в наиболее подходящей форме [3].

Для выполнения анализа в СППР должна накапливаться информация, а следовательно система должна обладать средствами ее ввода и хранения. Таким образом, можно выделить три основные задачи, решаемые в СППР:

- ввод данных;
- хранение данных;
- анализ данных.

По степени "интеллектуальности" обработки данных при анализе выделяют три класса задач анализа:

- информационно-поисковый — СППР осуществляет поиск необходимых данных (характерной чертой такого анализа является выполнение заранее определенных запросов);
- оперативно-аналитический — СППР производит группирование и обобщение данных в любом виде, необходимом аналитику (в отличие от информационно-поискового анализа в данном случае невозможно заранее предсказать необходимые аналитику запросы);
- интеллектуальный — СППР осуществляет поиск функциональных и логических закономерностей в накопленных данных, построение моделей и правил, которые объясняют найденные закономерности и/или (с определенной вероятностью) прогнозируют развитие некоторых процессов [4].

Обобщенная архитектура СППР может быть представлена следующим образом (рис. 1).



Рис. 1. Обобщенная архитектура СППР

Упрощенную структуру системы поддержки принятия решения, далее (СППР), можно представить в виде основных блоков: базы данных, базы моделей, базы знаний, которые входят в систему управления базами данных и базами знаний, переходящей в интерфейс пользователя, пользователем является лицо, принимающее решение (ЛПР) (рис. 2) [5].

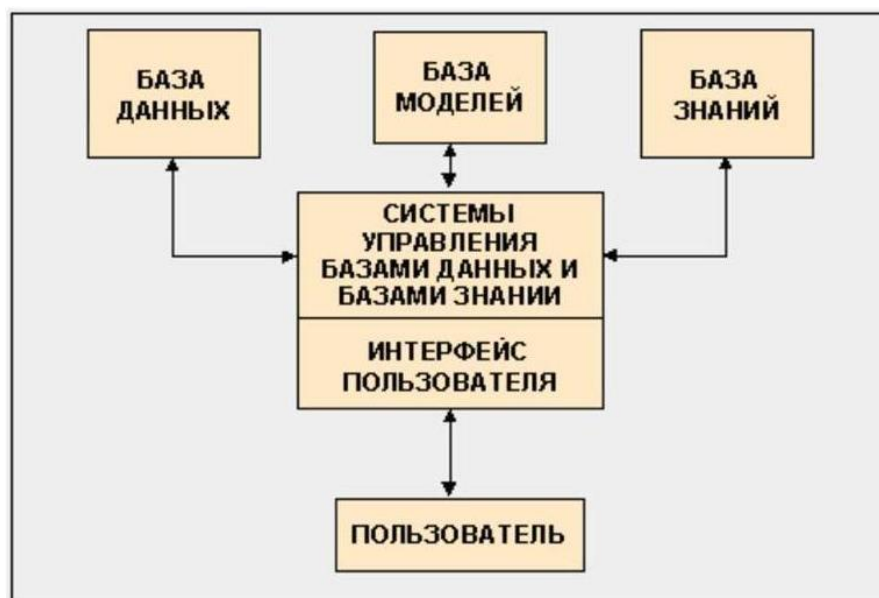


Рис. 2. Структурная схема СППР

Особенностью информационной технологии поддержки принятия решений является качественно новый метод организации взаимодействия человека и компьютера. Выработка решения, что является основной целью этой технологии, происходит в результате итерационного процесса (рисунок 3), в котором участвуют:

- система поддержки принятия решений в роли вычислительного звена и объекта управления;
- человек как управляющее звено, задающее входные данные и оценивающее полученный результат вычислений на компьютере.



Рис. 3. Информационная технология поддержки принятия решений как итерационный процесс

Окончание итерационного процесса происходит по воле человека. В этом случае можно говорить о способности информационной системы совместно с пользователем создавать новую информацию для принятия решений.

Дополнительно к этой особенности информационной технологии поддержки принятия решений можно указать еще ряд ее отличительных характеристик:

- ориентация на решение плохо структурированных (формализованных) задач;
- сочетание традиционных методов доступа и обработки компьютерных данных с возможностями математических моделей и методами решения задач на их основе;
- направленность на непрофессионального пользователя компьютера;
- высокая адаптивность, обеспечивающая возможность приспосабливаться к особенностям имеющегося технического и программного обеспечения, а также требованиям пользователя.

По взаимодействию с пользователем выделяют три вида СППР:

1. пассивные – помогают в процессе принятия решений, но не могут выдвинуть конкретного предложения;

2. активные – непосредственно участвуют в разработке правильного решения;
3. кооперативные – предполагают взаимодействие СППР с пользователем.

Выдвинутое системой предложение пользователь может доработать, усовершенствовать, а затем отправить обратно в систему для проверки. После этого предложение вновь представляется пользователю, и так до тех пор, пока он не одобрит решение.

По способу поддержки различают:

- модельно-ориентированные СППР, используют в работе доступ к статистическим, финансовым или иным моделям;
- СППР, основанные на коммуникациях, поддерживают работу двух и более пользователей, занимающихся общей задачей;
- СППР, ориентированные на данные, имеют доступ к временным рядам организации. Они используют в работе не только внутренние, но и внешние данные;
- СППР, ориентированные на документы, манипулируют неструктурированной информацией, заключенной в различных электронных форматах [6];
- СППР, ориентированные на знания, предоставляют специализированные решения проблем, основанные на фактах.

По сфере использования выделяют общесистемные и настольные СППР. Общесистемные работают с большими системами хранения данных и применяются многими пользователями. Настольные являются небольшими системами и подходят для управления с персонального компьютера одного пользователя.

Данная компьютерная технология СППР применима в ГБУ РК «Сакский районный ЦСССДМ», которая призвана помочь организации в решении трудных жизненных вопросов людей. Внедрение СППР значительно улучшит качество оказания услуг населению, будут использованы следующие компоненты, методы и технологии:

- Аналитическое хранилище данных;
- Инструменты загрузки и обработки информации;
- Автоматизация отчетности;
- Визуализация отчетности;
- Статистика и моделирование данных;
- Прогнозирование, Data Mining;
- Облачные технологии;
- Анализ иерархий для принятия решения;
- Обмен данными между компьютерами сети [6].

Для решения выше поставленных задач изучается возможность применения следующего программного обеспечения: TIBCO Spotfire, MPRIORITY, Ramus, Agro UML.

Библиографический список.

1. Научный журнал «Молодой учёный». Июнь, № 11 (91) / 2015.
2. Федеральный закон №442-ФЗ от 28 декабря 2013 г. Об основах социального обслуживания граждан в Российской Федерации.
3. Назначение систем поддержки принятия решений. [Электронный ресурс] Режим доступа: <https://studopedia.org/4-158624.html>. Дата доступа: 20.03.2018.
4. А. В. Вишняков, Н. С. Курилова, И. Е. Сафонова, В. И. Штейнберг. Многоцелевые задачи принятия проектных решений: Учебное пособие. - МГИЭМ. - М., 2002. - 101 с.
5. Барсегян А. А., Куприянов М. С., Степаненко В. В., Холод И. И.Б. Методы и модели анализа данных: OLAP и Data Mining. Учебное пособие. — СПб.: БХВ-Петербург, 2004. — 336 с.
6. Data mining, или Интеллектуальный анализ данных для занятых. Практический курс / В. Рафалович — «И-трейд», 2014, —19 с.

УДК 004.93'11

И.В. Макаров, магистрант**О.В. Ченгарь, канд. техн. наук, доцент***Севастопольский государственный университет***ПРИМЕНЕНИЕ ФУНКЦИИ ПРИНАДЛЕЖНОСТИ ДЛЯ СТРУКТУРНОГО
РАСПОЗНАВАНИЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ***Аннотация: В данной статье рассмотрена возможность применения технологии SURF для распознавания объектов в видеопотоке.**Ключевые слова: функция принадлежности, видеопоток, векторное представление**Annotation: Article explores the possibility of using SURF technology for object recognition in a video stream.**Key words: membership function, video stream, vector representation*

Задача распознавания образов до сих пор не решена в полном объеме. Однако, существуют методы, позволяющие приблизиться к ее решению.

В информационных системах необходимо решать прикладные задачи компьютерного зрения, связанные с распознаванием документов, подписей, человеческих лиц и других визуальных объектов. Такие задачи решаются путем сравнения измененного образа с эталонным описанием. Сравнение происходит в условиях геометрических преобразований, поэтому на объекте могут быть шумы, он может быть виден не полностью, иметь разный масштаб и тому подобное.

Зачастую используются структурные методы, основанные на выделении характерных признаков изображения, отражающих локальные изменения функции яркости [1,2]. Одним из эффективных современных методов структурного анализа визуальных объектов является технология SURF [2]. Она решает задачи поиска особых точек изображения, а также создание их дескрипторов, инвариантных к смещениям и поворотам. Это означает, что описание ключевой точки будет одинаковым, если образец смещен, и будет повернут.

В структурных методах распознавания изображений в качестве описания, по которому принимается решение о классе объекта, задают множество числовых векторов (дескрипторов) [2]. Дескрипторы – это характерные признаки изображения, которые вычисляются путем применения соответствующих методов (например, SURF) и содержат инвариантную к геометрическим преобразованиям информацию о функциях яркости фрагментов объекта. Структурное описание объекта представляется как конечное мультимножество. Метод SURF формирует описание по формуле (1).

$$Z \subset R_1^n, R_1^n = \{z \mid z \in R^n, \|z\| \approx 1\}, R_1^n \subset R^n, R_1^n, \quad (1)$$

Пусть $Z = \{z_i\}$ – конечное множество элементов, образующих описание, $z_i, z_k \in Z$. Зададим порог $\delta > 0$ и функцию расстояния $\rho(z_i, z_k)$ для пары $z_i, z_k \in Z$. Рассмотрим на множестве Z бинарное отношение $\Theta \subseteq Z \times Z$:

$$\Theta = \{(z_i, z_k); z_i, z_k \in Z \mid \rho(z_i, z_k) \leq \delta\}, \quad (2)$$

Оно определяет близость элементов $z_i, z_k \in Z$ с точностью δ . На основе анализа всех расстояний для элементов Z можно выделить кластеры «одинаковых» элементов в рамках параметра δ [1]. Условие $\rho(z_i, z_k) \leq \delta$ (замкнутый шар с радиусом δ с центром в z_i или z_k) в формуле (2) определяет бинарное отношение подобия Θ . Отношение в формуле (2) для произвольного элемента задает его область R_1^n , к которой принадлежат эквивалентные ему элементы.

Можно увидеть, что соотношение в формуле (2) имеет свойства рефлексивности, так как $\forall z_i$ выполнено $\rho(z_i, z_i) \leq \delta$, и симметричности: $\rho(z_i, z_k) = \rho(z_k, z_i) \leq \delta$. В то же время, это условие не задает разбиение Z , так как из $\rho(z_i, z_k) \leq \delta$ и $\rho(z_i, z_*) \leq \delta$ не следует исполнение (отсутствует транзитивность). Из условий $z_i \Theta z_k$ и $z_i \Theta z_*$ не значит, что $z_* \Theta z_k$, то есть отношение в строгом математическом смысле не является отношением эквивалентности.

Отношения, обладающие рефлексивностью и симметричностью, но не транзитивностью, получили название отношений толерантности [2]. Два элемента толерантны, если существует подмножество (класс), содержащий эти элементы. Система классов толерантности образует покрытие множества. Эквивалентность – частный случай отношения толерантности. Когда толерантность оказывается транзитивной, т.е. превращается в свой частный случай – эквивалентность, то классы толерантности становятся классами эквивалентности. Структура, обусловленная двухместной функцией принадлежности для четких множеств, определяет транзитивное отношение между парами точек следующим образом: из высказывания « X принадлежит тому же кластеру, что и Y » и « Y принадлежит тому же кластеру, что и Z » следует « X принадлежит тому же кластеру, что и Z » [2].

Наряду с формулой (1) для каждого $z_i \in Z$ целесообразно рассмотреть также унарное нечеткое отношение A принадлежности множеству [2], которое характеризуется значением функции принадлежности $a = A(z_i) \in [0, 1]$. Одноместное отношение A при фиксированном значении $A(z_i)$ определяет четкое подмножество $A \subseteq Z$.

Представляется целесообразным развитие методов структурного распознавания на основе использования значений функции принадлежности признакам эталона, которая учитывает индивидуальные особенности признаков по описанию. Для учета влияния признака на результат распознавания логично построить функцию принадлежности $A(z_i)$, основываясь на свойствах сходства элементов в рамках бинарного отношения толерантности.

Пусть $Z = \{z_k\}_{k=1}^s$ – конечное множество элементов, образующих структурное описание. Для каждого $z_k \in Z$ в соответствии с аппаратом нечетких множеств рассмотрим величину $A(z_k) = a_k$, $a_k \in [0, 1]$, характеризующую степень принадлежности ко множеству Z , т.е. нечеткое подмножество A множества Z . Как правило, степень принадлежности – это некоторая субъективная оценка, которая определяет нечеткое множество.

Если провести индексацию элементов A по правилу $A(z_i) \geq A(z_j)$ при $i > j$, то полученному нечеткому множеству на основе фиксированного значения $\alpha \in [0, 1]$ можно поставить в соответствие кортеж A_α четких подмножеств Z , называемых уровнями множества нечеткого множества A [2]. Подмножество $A_\alpha \subseteq Z$ α -уровня определяется как

$$A_\alpha = \{z \in Z \mid A(z) \geq \alpha\}. \quad (3)$$

Другими словами, A_α – это четкое подмножество Z , которое содержит элементы, степень принадлежности которых выше α . Каждое нечеткое множество может быть составлено из его уровней множеств [1]. В частном случае допускается $A_\alpha = \emptyset$.

Конструкция в формуле (3) может послужить основой для осуществления редукции описания Z , вкладывать, например, содержание уровня «важности» признака. За счет построения по формуле (3) может быть отобрано подмножество значимых признаков с высоким уровнем α .

Рассмотрим процедуру построения нечеткого множества A на основе множества для базы кадров, то есть прямой способ вычисления значения a_k функции принадлежности. Для элемента z_k из описания $Z = \{z_k\}_{k=1}^s$ и заданного значения δ_z определим число c_k его «повторений» в Z [1] по формуле:

$$c_k = \sum_{v=1, s} 1(\rho(z_k, z_v) \leq \delta_z), \quad (4)$$

где $\rho(z_k, z_v)$ – метрика для $z_k, z_v \in Z$;

Таким образом, $c_k \in C_+$ – это число элементов Z , эквивалентных $z_k \in Z$ с точностью до δ_z в соответствии с предикатом $1(\rho(z_k, z_v) \leq \delta_z)$, C_+ – множество целых неотрицательных чисел. Параметр δ_z определяет порог эквивалентности элементов (допустимая точность при их сопоставлении) [2]. Рассмотрим базу изображений из J эталонов и соответствующую ей множество $\{Z^i\}_{i=1}^J$ эталонных описаний. Для каждого из элементов $z_{ik} \in Z^i$ подсчитаем, сколько раз он встречается в эталоне Z^i , а также полной базе J эталонов. В результате вычислим параметр β_{ik} :

$$\beta_{ik} = c_{ik} / c_{ik}^J; \beta_{ik} \in [0, 1], \quad (5)$$

где c_{ik} – число эквивалентных z_{ik} элементов в i -м эталоне Z^i , c_{ik}^J – значение в формуле (4) для элемента z_{ik} в полной базе $\{Z^i\}_{i=1}^J$. Таким образом, значение в формуле (5) является частью элементов, подобных элементу z_{ik} в i -том эталоне среди общего числа таких элементов всей базы. Этапом процесса вычислений является определение значения по формуле (4) для отдельных стандартов, поэтому величина по формуле (5) может быть получена и проанализирована для каждого из эталонов базы отдельно. В частности, это можно использовать для выявления двух наиболее похожих эталонов. Величина β_{ik} , судя по формуле (5), меняется на отрезке $[0, 1]$, при $\beta_{ik} \rightarrow 1$ степень уникальности z_{ik} в базе наибольшая, при $\beta_{ik} \rightarrow 0$ элемент z_{ik} теряет уникальность и фактически становится непригодным для распознавания. Например, если элемент встречается исключительно в «своем» эталоне, то $\beta_{ik} = 1$. Если в базе все элементы равны, то $\beta_{ik} = s_i / s$, где s_i, s – множеств Z^i, Z . Величины по формулам (4) и (5) полностью определены параметром δ_z .

Отметим, что коэффициенты β_{ik} , кроме вычисления по формуле (5), могут также устанавливаться экспертом в интерактивном режиме, исходя из принципа их визуальной значимости в задаче распознавания. В таком варианте анализ обсуждаемых моделей наиболее адекватен нечеткому представлению.

Процедуру вычисления значений функции принадлежности для компонентов базы эталонов формулируем как последовательность шагов:

- 1) формирование множества $\{Z^i\}_{i=1}^J$ элементов описаний базы объектов;
- 2) вычисления для каждого элемента значение по формуле (4) при заданном δ_z ;
- 3) определение значений функции принадлежности по формуле (5) для каждого признака.

Отметим, что вычисление характеристик принадлежности и трансформация пространства признаков осуществляются на предыдущем этапе формирования описаний базы и в конце концов никак не влияют на быстрдействие распознавания.

Процедура распознавания теперь может быть построена с учетом степени принадлежности β_{ik} эталонных элементов Z^{i*} , что обеспечивает углубленный индивидуальный анализ значений и важность признаков, а это в целом способствует повышению достоверности принятия решения о классе объекта. С другой стороны, может быть сохранена и традиционная схема подсчета голосов для множества Z^{i*} сокращенных признаков [2].

Для объекта распознается, построить нечеткое описание предложенным способом трудно через препятствия и неточности представление, поэтому классификация на основе прямого вычисления сходства двух нечетких описаний затруднена. В таком случае распознавания традиционно строим на установлении оптимальной подобия рассматриваемых и эталонных элементов. Вместо подсчета числа голосов, в модифицированном методе суммируем значение для эталонных признаков, которым соответствует наибольшая сходство.

На основе отношения сходства по формуле 1 является возможность построить разбиение множества $\{Z^i\}_{i=1}^J$ на классы эквивалентных элементов, которые определяются порогом δ_z и метрикой $\rho(z_k, z_v)$. Вариантом построения может быть кластеризация на множестве. Другим вариантом может быть формирование нечетких классов.

Преобразование значительно упрощает сопоставление описаний, так как вычисление меры для множеств трансформируется в определение меры векторов, за счет чего вычислительные затраты на распознавание снижаются в десятки раз [2]. Конструкция преобразования базы представлена на рисунке 1.

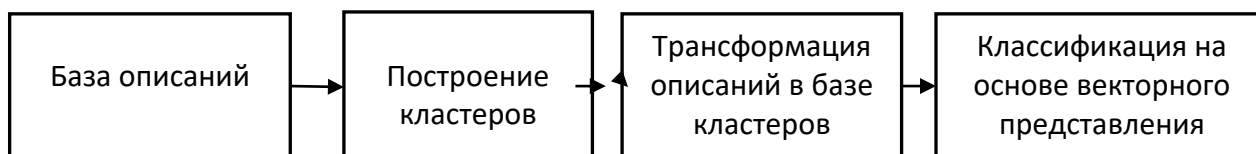


Рис. 1. Схема преобразования описаний

Библиографический список

1. Нечеткие множества и теория возможностей. Последние достижения [под ред. Р. Ягера]; пер. с англ. – М.: Радио и связь, 1986. – 408с.
2. Гороховатский В.А. Распознавание изображений в условиях неполной информации. – Харьков: ХНУРЭ, 2003. – 112с.

УДК 681.3

Н.А. Панасюк**Е.Н. Машенко, канд. техн. наук, доцент***Севастопольский государственный университет***ИССЛЕДОВАНИЕ ПРОЦЕССОВ АНАЛИЗА ТЕКСТОВЫХ ДАННЫХ***Аннотация: в статье проведен обзор процессов анализа текстовых данных, методов технологии Text Mining, а также представлена имитационная модель системы распознавания языка текста.**Ключевые слова: Интеллектуальный анализ данных, текстовый анализ, обработка, символы.**Annotation: The article contains overview of text data analysis processes, Text mining technology and a simulation model of the recognition system of text language.**Key words: Text Mining, text analysis, processing, symbols.*

Введение. Существенный скачок в развитии компьютерных технологий в 20-м веке оказал большое влияние практически на все сферы деятельности человека. Области применения компьютеров и мобильных устройств расширяются с каждым годом, что свидетельствует о прогрессе в развитии IT-технологий.

Значительно увеличивается и количество информации. Совокупность данных (информации) и средств ее хранения представляют собой информационные ресурсы. Очевидно, что объем данных со временем только растет. В связи с этим требуется использовать интеллектуальные методы обработки большого количества данных. Применяются технологии, призванные повысить эффективность решения такого рода задач.

Интеллектуальный анализ данных (ИАД) представляет собой направление в области информационных технологий, в котором рассматриваются, сравниваются и анализируются большие массивы информации. Различные методы анализа данных позволяют получать ценные результаты для исследователей и аналитиков, что непосредственно сказывается на решениях менеджеров и должностных лиц корпораций.

В области интеллектуального анализа текстовых данных наиболее известной является технология Text Mining.

1. Обзор методов технологии Text Mining.

На данный момент, технология Text Mining представляет широкий набор инструментов, с помощью которого возможно осуществлять поиск и формирование шаблонов, взаимосвязей и закономерностей.

Применение данной технологии широко используется в системах электронного документооборота, а также для бизнеса в задачах привлечения и удержания клиентов. Статистика в Web Mining свидетельствует о том, что, зная о предпочтениях пользователя, система способна выдавать рекомендации, которые заинтересуют клиента и побудят детальнее узнать информацию о продукте.

Применение Text Mining актуально и в области обработки больших массивов данных. Очевидно, что в хранилищах данных находятся огромнейшие объемы информации, как правило, в структурированном виде. Однако, далеко не любые данные можно привести к структурированному виду без потерь полезной информации. Совокупность текстовой информации, необходимой для дальнейшего использования, не может быть обработана алгоритмами Data Mining, если она неструктурированная. В таком случае, проблема решается методами Text Mining.

Text Mining также можно использовать для решения задач автоматического аннотирования, извлечения ключевых понятий и др.

Автоматическое аннотирование представляет собой процесс сокращения числа символов и слов без утери смысловой информации [1]. Данный метод позволяет сократить текст, сохраняя его смысл. Решение этой задачи обычно регулируется пользователем при помощи количества извлекаемых предложений или процентом извлекаемого текста по отношению ко всему тексту. Результат включает в себя наиболее значимые предложения в тексте.

Извлечение ключевых понятий подразумевает анализ определенных взаимосвязей, фактов и отношений для их последующей замены на новые понятия [1]. Такой подход позволяет отобразить полученные ключевые понятия в виде структур, которые можно хранить в базах данных.

Согласно определению [2], Text Mining (обнаружение знаний в тексте) - нетривиальный процесс обнаружения действительно новых, потенциально полезных и понятных шаблонов в неструктурированных текстовых данных. Следует отметить, что шаблоны в Text Mining представляют собой совокупность взаимосвязанных слов, вне зависимости от самой структуры документа. Примерами таких документов могут являться исходники web-ресурсов, web-страницы, электронная почта и другие. Особенностью таких документов является наличие не только текстовой информации, но и графической.

Например, такие документы, как XML (eXtensible Markup Language), по своей структуре принято считать полуструктурированными. Однако, методы Text Mining способны обрабатывать и такие виды документов.

Этапы Text Mining.

Последовательность действий при анализе текстовых документов [3] приведена на рисунке 1.

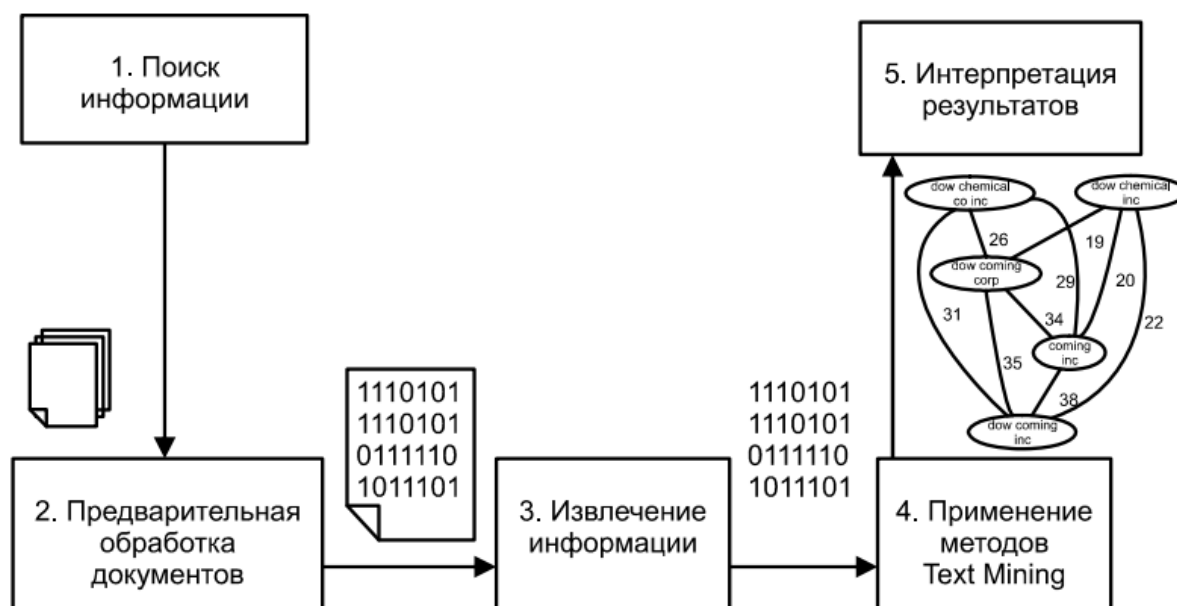


Рис. 1. Схема последовательности этапов Text Mining

Этап №1. Поиск информации.

Первым шагом является выявление критериев, по которым будет происходить выбор необходимой информации. Данные должны быть доступны для анализа и для последующей обработки. В случае большого объема поступающей информации необходимо реализовать автоматизацию обработки информации по принятым критериям.

Этап №2. Предварительная обработка документов.

Известно, что не все слова содержимого в документе несут ключевую смысловую нагрузку. Вследствие этого, документ содержит определенное количество неинформативной информации, что значительно повышает время обработки данных. Поэтому необходим этап предварительной обработки текста, в ходе которого повышается строгость текста. В ходе данного этапа применяются такие приемы, как: удаление вспомогательных слов и артиклей; приведение регистра; морфологический поиск и разбор.

Этап №3. Извлечение информации.

Является крайне важным этапом. Целью является идентификация терминов (фактов) и шаблонов, по которым будет проводиться поиск и обработка дальнейших данных. В качестве таких шаблонов можно выделить конкретные сущности, даты, имена нарицательные. Поскольку данный процесс является сложным в реализации на практике, его необходимо разбить на части, такие как локальный анализ и анализ понятий. В ходе анализа всего содержимого данные сопоставляются согласно образцам. На выходе полученные данные представлены зависимостями: связи, сущности и так далее.

Этап №4. Применение методов Text Mining.

Основу для всех методов составляет процесс извлечения ключевых слов и фраз, при котором ранжируются словоформы или лексемы по частоте. В результате задается определенный уровень ключевых слов, после чего выбираются наиболее часто появляющиеся лексемы. Впервые этот метод применил в своих работах Г.П. Лун [3]. Метод достаточно распространен, поскольку прост в реализации.

Этап №5. Интерпретация результатов.

В результате анализа и обработки данных необходимо правильно интерпретировать полученный результат. Интерпретация возможна как на естественном языке, так и в графическом варианте. Визуализация информации нагляднее отображает результаты анализа.

2. Имитационная модель системы распознавания языка текста с использованием нейронных сетей.

2.1 Применение нейронных сетей в распознавании образов и классификации.

Одним из методов, применяемых для решения задач Data Mining являются нейронные сети. Искусственные нейронные сети представляют собой вычислительную структуру, моделирующую работу человеческого мозга. Данную структуру можно представить, как направленный граф, отображающий восприятие и обработку входных данных. Искусственные нейронные сети (ИНС) применяются в различных сферах науки: от распознавания информации до классификации разных видов тяжелых болезней, а также генной инженерии.

В науке существует большое количество задач, в которых необходимо принимать решение в зависимости от наличия конкретных объектов на изображении. Распознаванием считается способность того ли иного объекта классифицировать образ по определенным характерным признакам.

Структура системы распознавания изображена на рисунке 2.

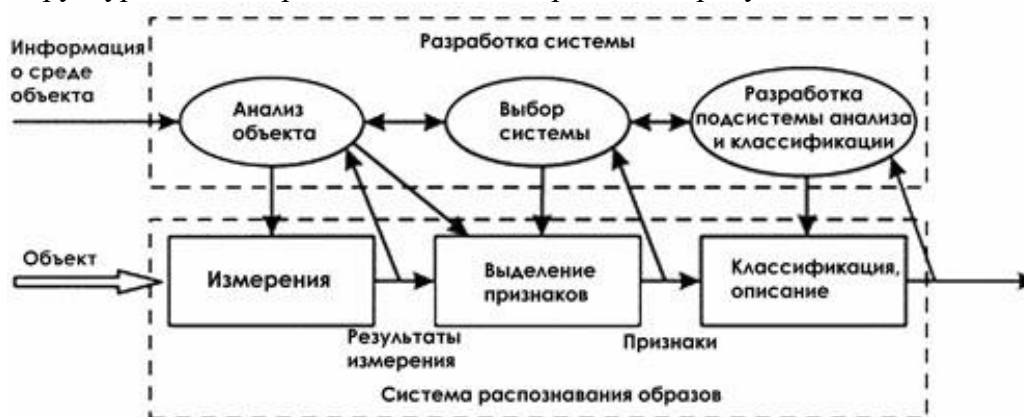


Рис. 2. Схема системы распознавания образов

Распознавание образов – это научная задача, которая применяет методы и принципы построения систем классификации данных. Долгое время распознаванием занимались лишь науки, связанные с биологией и психологией. Именно поэтому точно определить механизм

функционирования распознавателя было невозможно. В процессе эволюции большинство особей живого мира научились решать задачу распознавания при помощи слухового и зрительного аппарата. Однако создание систем искусственного интеллекта, владеющих функциями распознавания, до сих пор остается сложной технической задачей, требующей затрат.

Проблема распознавания включает две части: обучение и распознавание. Обучение – это процесс, в ходе которого система все больше развивает способность реагировать на соответствующие внешние воздействия [4], а также приспосабливается к изменениям внешних условий и при этом не теряет качества. Результатом обучения распознающей системы является приобретение одинаково реагировать на одинаковые образы, а на различные образы – реагировать иначе. Стоит отметить, что в процессе обучения указывается лишь объекты и их классификация.

2.2 Описание системы распознавания языка текста.

Рассмотрим реализацию системы, решающую задачу распознавания пяти языков: английский, испанский, польский, немецкий, турецкий. Разрабатываемая система осуществляет решение задачи классификации при помощи нейронных сетей. На выходе мы должны получить язык текста. Неструктурированный текст задается входным файлом с расширением txt.

Поскольку проектируемая система представляется сложной системой, то для ее синтеза необходимо провести системный анализ. Необходимо обеспечить стойкость системы, свести к минимуму потери и повысить эффективность.

Согласно принципу связности спроектирована структура проектируемой системы (рисунок 3). Вся система разделяется на несколько подсистем.

Первая подсистема проверяет корректность исходных данных, которые обязаны соответствовать заданным требованиям. После этого данные подаются на вход классификатора, построенного на основе нейронной сети.

Принцип распознавания текста нейронной сетью - подсчет частот символов в тексте и сравнение их с эталонами (известными частотами) для разных языков.

Нейронной сетью проводится распознавание языка текста и формируется выходной результат, который в подсистеме вывода представляется для пользователя с помощью интерфейса.

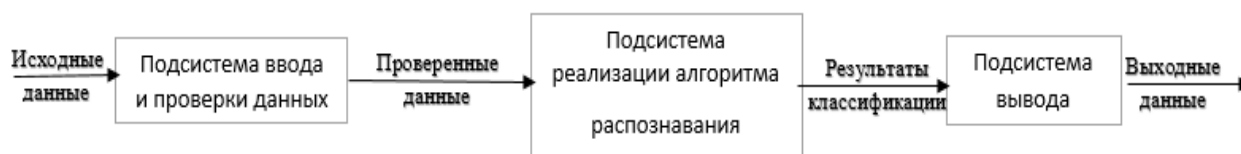


Рис. 3 - Схема разбиения на подсистемы согласно принципу системного анализа

2.3 Моделирование системы распознавания в среде Anylogic.

В ходе разработки имитационной модели были реализованы такие блоки как:

- 1) Генерирование заявок (sourcevh);
- 2) Очереди на (обработку – queueobr, обучение – queueobuch, классификацию – queueclass);
- 3) Задержки на (обработку – delayobr, обучение – delayobuch, классификацию – delayclass);
- 4) Вывод результата распознавания - количества обработанных заявок (sinkresult).

Параметры блоков заданы по распределению (равномерное или треугольное). Значения взяты в секундах. Delay классификатора – треугольное (2, 4, 6). Время моделирования – 50000 секунд. Входные параметры модели представлены в таблице 1.

Табл. 1 - Значения распределений для блоков модели

№языка/ Распределение	Sourcevh равномерное	Delayobr треугольное	Delayobuch треугольное
1	600, 720	30,60,90	10,12,14
2	780,900	30,60,90	11,14,17
3	720,840	30,60,90	8,10,12

Схема имитационной модели и результирующая статистика представлена на рисунке 4.

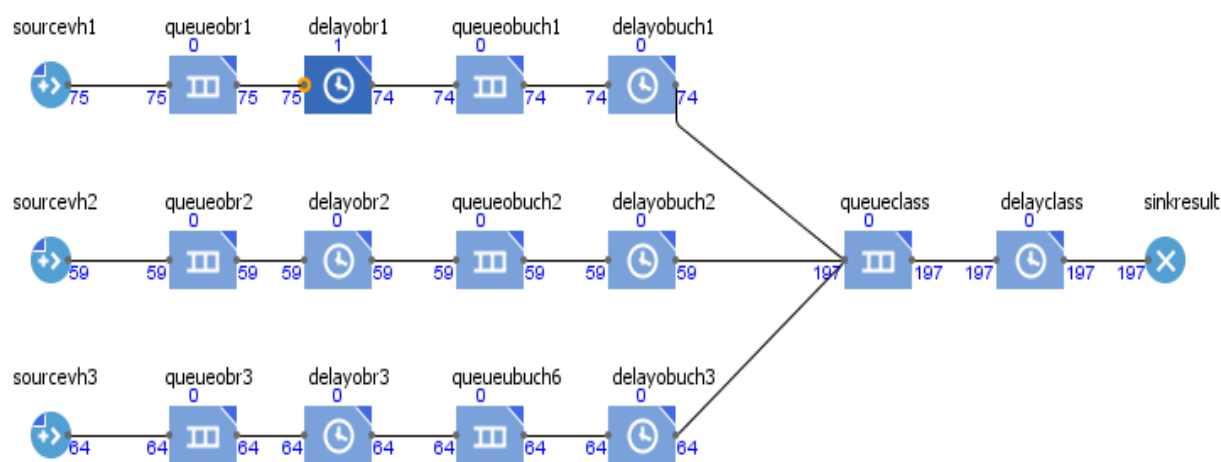


Рис. 4 – Имитационная модель системы распознавания языка текста

Закключение. В данной статье был проведен обзор процессов анализа текстовых данных, рассмотрено применение технологии Text Mining в области обработки больших массивов данных, а также представлена имитационная модель системы распознавания языка текста с использованием нейронных сетей. Перспективами дальнейших исследований является расширение комплекса имитационных моделей процессов анализа текстовых данных.

Библиографический список

1. Горковенко Д. К. Применение методов text mining для классификации информации, распространяемой в социальных сетях // Молодой ученый. — 2016. — №14. — С. 66-72
2. Чубукова И.А. Data Mining. - М.: Изд-во «Интернет-университет информационных технологий - ИНТУИТ.ру», 2008. - 384 с.
3. Алексеев А. А., Катасёв А. С., Кириллов А. Е., Кирпичников А. П. Классификация текстовых документов на основе технологий Text Mining // Вестник технологического университета, 2016. Т. 19 №18. С. 110-116.
4. А. А. Барсегян и др. Анализ данных и процессов: учеб. пособие / А. А. Барсегян, М. С. Куприянов, И. И. Холод, М. Д. Тесс, С. И. Елизаров. — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2009. — 512 с

УДК 004.6

С.С. Поволоцкий, студент 4-го курса,**М.В. Заморёнов, канд. техн. наук, доцент***Севастопольский государственный университет***СИНЕРГИЯ BIG DATA И BLOKCHAIN**

Аннотация: Натан Ротшильд, основатель огромной банковской династии Ротшильдов, сказал следующую фразу, которая сразу стала крылатой: «Кто владеет информацией, тот владеет миром». Своим высказыванием автор хотел затронуть проблему значимости информации в бизнесе. Он считал, что тот человек, кто первым владеет информацией, имеет огромное преимущество перед другими людьми. Тот, кто владеет информацией, может использовать ее в своих целях для решения даже самых больших задач. В данной статье рассматриваются методы и механизмы управления большим количеством информации современными средствами и технологиями.

Ключевые слова: Блокчейн, большие данные, прикладные технологии, производственная оптимизация.

Annotation: Nathan Mayer Rothschild said: "who owns the information that owns the world". By these words he wanted to pay attention on information wealth problem. He was confident concerning the idea of being the first of getting and managing information for getting the advantage among competitors. This article will get you acknowledged about the methods and mechanisms of managing big data amount by modern technologies.

Key words: Blockchain, Big Data, application technologies, optimization of production process.

Сегодня никого не удивит большими объемами данных и стремительным ростом их накоплений. Социальные сети, измерительные устройства, метеорологические данные – это всего лишь малая часть всех источников информации, способных генерировать все больше и больше данных. Сбор и анализ полученной информации представляет собой бесценный объект пристального внимания не только со стороны науки и бизнеса, но и государства. Одними из поводов заинтересованности всех сторон в сборе и обработке данных могут быть возможность прогнозирования грядущих событий, возможность принятия истинно правильного решения, опираясь на предыдущий опыт, возможность объективного анализа рынка с последующим более выгодным расставлением приоритетов. Тем не менее, традиционные методы анализа полученной информации оказались неэффективными в виду того, что в большинстве случаев собранная информация была представлена в формате (аудио/видео-записи, фотографии, веб-журнал и т.д.), не позволяющим провести их эффективный разбор. Итогом стал тот факт, что организации имели данные, но не имели средств для установления взаимосвязи между ними и не могли сделать значимые выводы на их основе. Не стоит забывать про стремительно растущий поток информации, в результате чего традиционные методы анализа не были способны эффективно и корректно обрабатывать новые данные. Для решения сложившихся проблем была создана Big Data [1].

Big Data представляет собой набор инструментов, подходов и методов обработки структурированной и неструктурированной информации огромного объема значительного многообразия. Причиной проникновения Big Data во все сферы деятельности человека стали стремительное развитие IT-технологий и принципиально новые методы их использования. Согласно рекрутерской компании Robert Half International Inc, многие компании обеспокоены обработкой больших объемов данных.

Основные принципы Big Data:

- Объем. Большое число источников информации с большим размером данных провоцирует рост объема данных, необходимых для анализа.
- Скорость. Результат полезен только при минимальных задержках обработки информации.
- Вариативность. Информация поступает во всех возможных форматах (звук, видео, текст и т.д.)
- Надежность. В дополнении к растущей скорости и вариативности данных, процесс сбора и обработки должен иметь погрешность, стремящуюся к нулю.
- Сложность. Информация поступает из разнообразных источников, что затрудняет процесс преобразования и сопоставления информации между системами.

Работа с большими объемами данных имеет следующие принципы:

- Горизонтальная масштабируемость. Должна быть расширяемость обрабатывающей системы в зависимости от увеличения обрабатываемых данных. К примеру, в случае увеличения объема данных в 2 раза увеличится количество железа в кластере в 2 раза.
- Отказоустойчивость. Согласно горизонтальной масштабируемости, кластер может содержать много машин. Это подразумевает выход их из строя. Методы работы с данными должны допускать такие сбои и продолжать корректно работать.
- Локальность данных. В больших распределенных системах данные хранятся на различных машинах. В случае нахождения данных на одном устройстве, а обработки на другом – расходы на передачу информации может превысить расходы на их непосредственную обработку. Во избежание этого, целесообразно хранить и обрабатывать данные на одной и той же машине.

Основываясь на последних наработках Accenture, GE и IBM можно говорить о значительном влиянии аналитики больших данных на бизнес. Так, 92% руководителей были полностью удовлетворены результатами исследований и их влиянием на свой бизнес, 89% назвали аналитику очень важной составляющей в налаживании бизнес-процессов и прогнозировании. Если компания завершила хотя бы один проект изучения больших данных и отметила его впечатляющие результаты и революционное влияние на дальнейшую стратегию ведения дел, она вряд ли откажется от такого анализа в дальнейшем в пользу устаревших методов. Это, без сомнений, революция в мире бизнеса.

В аналитическом отчете, посвященном последним Интернет трендам и технологиям 2015 года от венчурного фонда KPCB, аналитик Мэри Микер рассказала, как изменился Интернет за последние десятки лет. Так, стоимость компьютерных вычислений и хранения данных заметно снизились, а Big Data приносят компаниям невероятную пользу за счет уменьшения расходов на многие бизнес-операции и инвестиции.

Выделяются 10 основных сфер применения технологии Big Data:

- Маркетинг. Сейчас это наиболее популярное место для реализации Big Data. Big Data применяется с целью лучше понять клиентов, их поведение и предпочтения. Одним из наиболее значимых результатов применения Big Data стало появление контекстной рекламы.
- Бизнес. Продавцы способны оптимизировать сбыт продукции, опираясь на прогнозы, сделанные с учетом трендов среди поисковых запросов, погодных прогнозов и др.
- Личностное развитие. Каждый из нас может получить пользу от информации, полученной от носимых устройств (к примеру, умные часы или фитнес-браслеты). Таким образом, мы можем, используя полученную информацию, оптимизировать наш режим дня, рацион и даже физическую нагрузку.
- Здравоохранение. Опираясь на анализ информации о пациенте, можно выявить как наиболее подходящее лекарство, так и с высокой точностью определить причину недуга. Также есть возможность определить реакцию организма на тот или иной препарат и сделать соответствующие выводы. Так, Kaiser Permanente детально изучил образования тромбов у женщин, которые принимали оральную контрацепцию. В процессе удалось выяснить, что одно из лекарств повышает риск появления тромбов на 77%.
- Спорт. Big Data применяется для наблюдения и изучения спортсмена с целью повышения его показателей.
- Наука. Эксперименты производят огромные количества данных, которые в дальнейшем будут детально разобраны.
- Промышленность. Big Data помогает машинам и устройствам стать “умнее” и более автономными. Одним из множества примеров может быть беспилотный

автомобиль, который оборудован камерами, GPS, различными сенсорами и мощным оборудованием для анализа получаемой информации и быстрым реагированием на изменяющиеся условия.

- Безопасность. Big Data применяется не только для предотвращения преступления, но и для поимки преступника.
- Администрирование. Big Data применяется для оптимизирования тех или иных ресурсов. К примеру, в случае с организацией транспортного движения, становится возможным оптимизировать трафик, опираясь на получаемую информацию в режиме реального времени.
- Торговля. В этой сфере деятельности алгоритмы Big Data призваны делать решения относительно торговли.

Несмотря на все очевидные преимущества, Big Data имеет также и недостатки:

- Высокая стоимость.
- Этический характер (в случае сбора информации о пользователе без его ведома).
- Проблема выбора обрабатываемых данных.
- Достоверность данных.

Последние два недостатка исправить может связка Big Data + Blockchain [2,3]. Blockchain представляет собой способ хранения данных или цифровой реестр транзакций, сделок, контрактов. Всего что нуждается в отдельной независимой записи и, при необходимости, в проверке. В блокчейне можно хранить данные о выданных кредитах [4], правах на собственность, нарушении правил дорожного движения, бракосочетаниях. То есть практически обо всем. Главным его отличием и неоспоримым преимуществом является то, что этот реестр не хранится в каком-то одном месте. Он распределён среди нескольких сотен и даже тысяч компьютеров во всем мире. Любой Пользователь этой сети может иметь свободный доступ к актуальной версии реестра, что делает его прозрачным абсолютно для всех участников. Иными словами, это выстроенная по определённым правилам непрерывная последовательная цепочка блоков (связный список), содержащих информацию, копии которой хранятся и независимо друг от друга обрабатываются на множестве других компьютеров, подключенных к системе.

Блокчейн содержит то, что хотели видеть все специалисты в области управления данными: информацию, которая поставляется достоверным источником, о том, кто, когда и какие действия совершил вплоть до самого первого дня. Информацию, проверенную множественными третьими лицами и полностью соответствующую последним требованиям криптографии. Блокчейн открывает новые возможности не только для бизнеса, но и для IT-индустрии, которая традиционно поддерживает и автоматизирует бизнес. По словам гендиректора компании SAP Labs в СНГ Андрея Биветски, блокчейн действует в комбинации с аппаратным обеспечением, такими как облака и физическая память без данных. Это позволяет развить программные платформы для управления данными, обрабатываемыми блокчейном, а также для их анализа по запросу бизнес-приложения. Уверенность в информации на выходе при управлении данными станет расти по мере распространения блокчейна. Не исключено, что лет через пять технологию блокчейн будут применять для распределенного облачного хранения данных повсеместно. Пока еще облачные хранилища централизованы, что заставляет пользователя доверять всю его онлайн-информацию единственному поставщику услуг.

Блокчейн способен помочь в борьбе с преступностью. С его применением можно создать децентрализованные сети связи, сохранив определенный уровень прозрачности. Это позволяет пользоваться сетями, как и прежде, но любая подозрительная передача данных или действий в сети будет вполне отслеживаема.

Блокчейн может быть двух видов – открытый и частный. Децентрализованные открытые сети на блокчейне сами отслеживают важные данные и защищают их от

вредоносного или мошеннического вмешательства. Открытые сети на основе блокчейна используют механизм всеобщего консенсуса, т.е. все компьютеры в сети должны прийти к согласию в том, какие данные являются подлинными и готовыми к включению в блокчейн, а какие являются попыткой вторжения и их следует отбросить. Частные же сети на основе блокчейн используют специфический механизм консенсуса, позволяя внести новые данные только идентифицированным пользователям.

По мнению Оливера Буссмана, IT-менеджера швейцарского финансового холдинга UBS, технология блокчейн способна «сократить время обработки транзакций от нескольких дней до нескольких минут».

Тандем Blockchain и Big Data можно успешно использовать в здравоохранении. Как известно, несовершенные и неполные данные о здоровье пациента в разы увеличивают риск постановки неверного диагноза и неправильно назначенного лечения. Критически важные данные о здоровье клиентов медучреждений должны быть максимально защищенными, обладать свойствами неизменности, быть проверяемыми и не должны быть подвержены каким-либо манипуляциям.

Информация в блокчейне соответствует всем перечисленным требованиям и может служить в роли качественных и надежных исходных данных для глубокого анализа при помощи новых технологий Big Data. Помимо этого, при помощи блокчейна медицинские учреждения смогли бы обмениваться достоверными данными со страховыми компаниями, органами правосудия, работодателями, научными учреждениями и другими организациями, нуждающимися в медицинской информации.

Одна из главных проблем больших данных, которую призван решить блокчейн, лежит в сфере информационной безопасности. Обеспечивая соблюдение всех основных ее принципов, технология распределенного реестра может гарантировать целостность и достоверность данных, а благодаря отсутствию единой точки отказа, блокчейн делает стабильной работу информационных систем. Технология распределенного реестра может помочь решить проблему доверия к данным, а также предоставить возможность универсального обмена ими.

Информация – ценный актив, а это значит, что на первом плане должен стоять вопрос обеспечения основных аспектов информационной безопасности. Для того, чтобы выстоять в конкурентной борьбе, компании должны идти в ногу со временем, а это значит, что им нельзя игнорировать те потенциальные возможности и преимущества, которые заключают в себе технология Blockchain и инструменты Big Data.

Библиографический список:

1. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим / Виктор Майер-Шенбергер, Кеннет Кукер ; пер. с англ. Инны Гайдюк. — М. : Издательство «Манн, Иванов и Фербер», 2014. — 240 с.
2. Децентрализованные приложения. Технология Blockchain в действии / Равал Сираж. — СПб.: Издательство «Питер», 2017. — 240 с.
3. Блокчейн: Как это работает и что ждет нас завтра / Артем Генкин, Алексей Михеев — Москва : Издательство «Альпина Паблишер», 2018. — 592 с.
4. Блокчейн: Схема новой экономики / Мелани Свон : [перевод с английского]. — Москва : Издательство «Олимп-Бизнес», 2017. — 240 с.

УДК

Ю.А. Цокало, студентка 4-го курса

С.Г. Лелеков, канд. ф. м. наук., доцент

Севастопольский Государственный университет

ПРОГРАММНАЯ СИСТЕМА АВТОМАТИЧЕСКОГО ВОССТАНОВЛЕНИЯ ГРАММАТИКИ ПО ЦЕПНОМУ КОДУ ОБЪЕКТОВ В СИСТЕМАХ РАСПОЗНАВАНИЯ ОБРАЗОВ

Аннотация: рассматриваются методы обработки изображений, вопросы их сегментации и интерпретации на основе синтаксического подхода. Приводятся эффективные алгоритмы автоматического восстановления грамматики и вычисление степени отличия распознанного и эталонного объекта. Ключевые слова: обработка изображений, цепной код Фримена, автоматическое восстановление грамматики, последовательный код.

Annotation: considers image processing methods, their segmentation and interpretation based on the syntactic approach. Effective algorithms for automatically recovering grammar and calculating the degree of difference between the recognized object and the reference object. Key words: image processing, freeman's chain code, automatic grammar recovery, sequential code.

В настоящее время компьютерные системы анализа изображений становятся стандартным инструментом специалистов, работающих в разных областях, например биологии, медицине, фармакологии, материаловедении, географии и др. В фармакологии при изучении морфологических изменений клеток необходима не только визуальная информация в виде наблюдений и сравнений, а также обязательным становится подсчет размера клеток, глубины свечения и статистический анализ формы клетки. Для решения этих задач все чаще используются методы анализа изображений.

Существует несколько основных методов анализа изображений:

1. Кластерный анализ (или кластеризация) – это задача разбиения множества объектов на группы, называемые кластерами. Внутри каждой группы должны оказаться «похожие» объекты, а объекты разных группы должны быть как можно более различны. При кластеризации перечень групп четко не задан и определяется в процессе работы алгоритма. [1].

2. Корреляция цифровых изображений (англ. *Digital Image Correlation and Tracking*, (DIC/DDIT)) — оптический метод, используемый при отслеживании и идентификации изображения для точных плоских и объемных измерений изменений на изображении. Например, для измерения деформаций, полей перемещений и оптических потоков. Одно из наиболее широко известных применений данного метода — идентификация перемещений оптической мышки.[3]

3. Спектральный анализ – это один из инструментов обработки экспериментальных данных. В частности, он используется для анализа данных, выявления характерных частот, в целях подавления шума и др. [3].

4. Контурный анализ изображения – позволяет эффективно решать многие проблемы распознавания образов. Анализ изображения включает в себя получение внешнего контура изображенных объектов и запись координат точек этого контура. Чаще всего требуется получить внешний контур в виде замкнутой кривой или совокупности отрезков дуг [6].

5. Порогового преобразования (рисунок 1), выделение объектов по цвету. В этих случаях получают бинарное изображение, в котором явным образом задаются границы объекта. Эта совокупность пикселей, которая составляет границу объекта и есть контур объекта. Чтобы оперировать контуром, который был получен, его необходимо закодировать, например, отметить вершины отрезков, составляющих контур [4].

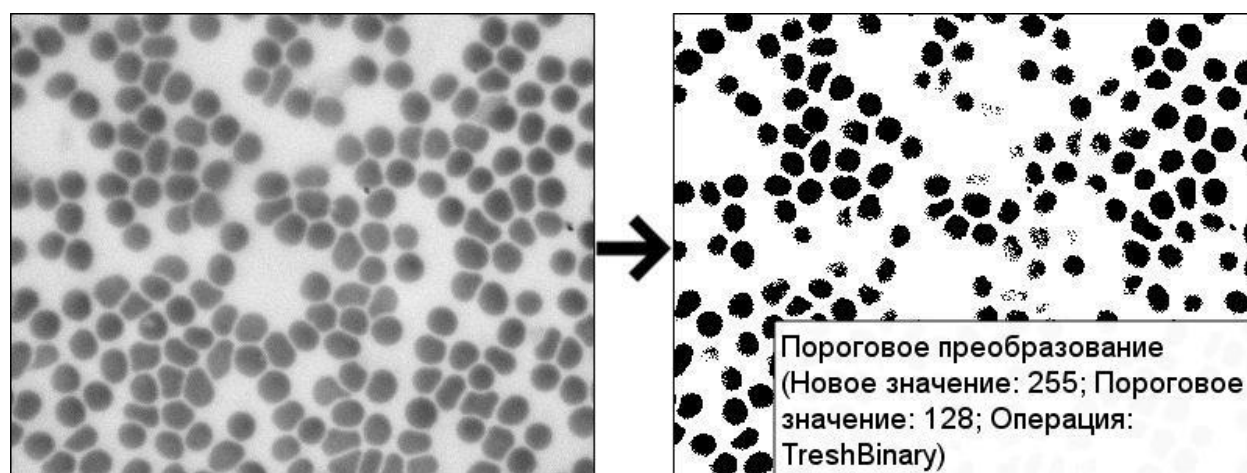


Рис.1. Обработка изображения методом порогового преобразования.

Рассмотрим более подробно контурный анализ изображения. При проведении контурного анализа необходимо полагать, что контур содержит достаточную информацию о форме объекта. Контур – это внешние очертания предмета/объекта. Преимущества метода: 1) простой алгоритм обработки изображения, 2) несложные математические вычисления и 3) высокое быстродействие. Недостатки - слабая устойчивость к помехам и зашумленности[4]. Получить контур можно несколькими методами, например, такими, как детектор границ Кенне [7]. С помощью этого метода можно уменьшить толщину контура изображения. Пример использования метода представлен на рисунке 2.

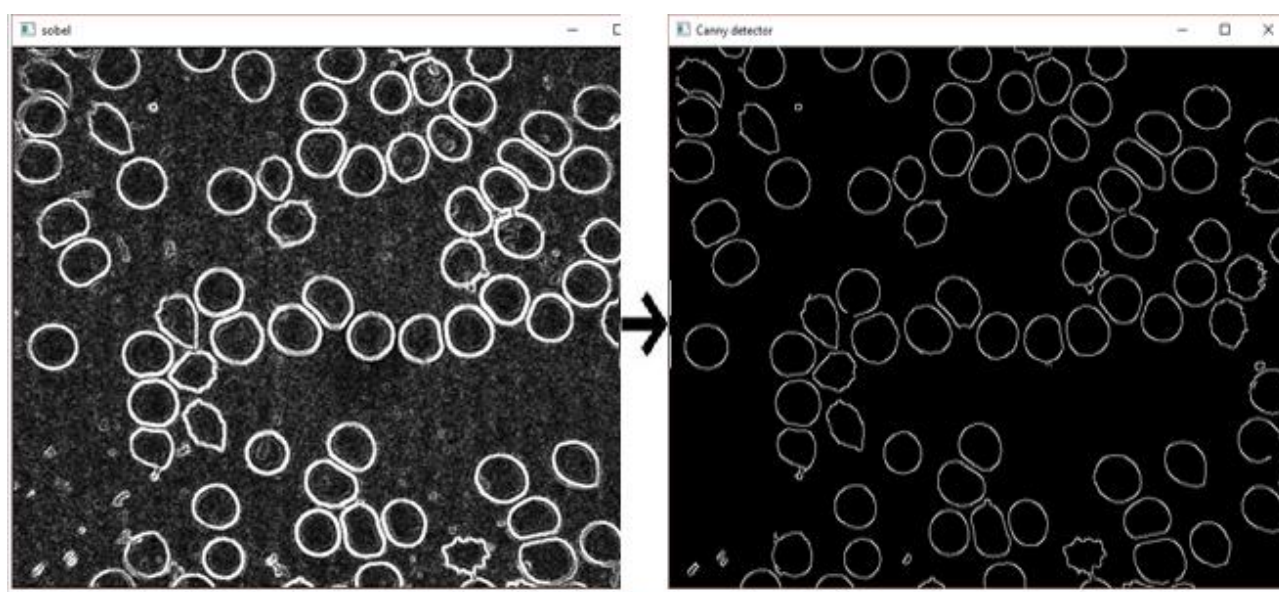


Рис. 2. Обработка изображения методом детектора границ Кенне.

Другой известный способ кодирования контура – это цепной код Фримена. Г.Фрименом была предложена схема цепного кодирования, которая обычно используется для описания границ и остовов множеств произвольных элементов [5]. На двумерное изображение накладывают прямоугольную сетку. Узлы сетки, которые наиболее близки к точкам изображения, соединяют отрезками прямых. Каждому такому отрезку в соответствии с его наклоном присваивают восьмеричное число. Таким образом изображение представляется цепью (последовательностью) или цепями восьмеричных чисел. На рисунке 3 показано множество начальных элементов и кодовая цепочка, описывающая кривую. Эта дескриптивная схема обладает рядом полезных свойств. Например, поворот изображения на угол, кратный 45° , сводится к прибавлению восьмеричного числа (сложению по модулю 8) к каждому числу цепочки. Конечно, при этом изображение может

искажаться. Только поворот на угол, кратный 90° , никогда не приводит к искажениям изображения. Легко выполнимы и некоторые другие простые манипуляции с изображением, такие, как растяжение, измерение длины кривой, определение самопересечений. Изменяя зернистость сетки, накладываемой на изображение, можно получить любое желаемое разрешение. Этот метод не ограничен изображениями с односвязными замкнутыми границами. Его можно применять для произвольных двумерных фигур, составленных из прямых и кривых линий и отрезков [5].

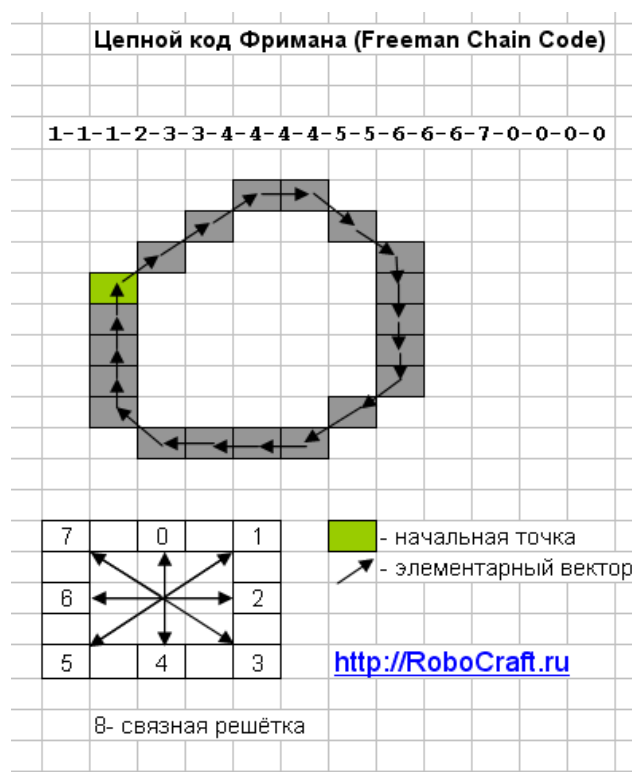


Рис.3. Цепной код Фримана

Если последовательно объединить детектор границ Кенни и цепной код Фримана, то можно получить более точное описание изображения в виде цепного кода, так как при сужении границ изображения разброс по его контуру будет меньше, тем самым цепной код Фримана будет более точным. После получения цепного кода возможна обработка этой информации, а именно составление и анализ грамматик. Это необходимо для решения задачи распознавания в реальном масштабе времени.

Описание программной системы автоматического восстановления грамматики по цепному коду объектов в системах распознавания объектов.

Следующим этапом анализа данных является сегментация и интерпретация изображения. Для этих целей может быть использован синтаксический подход на основе эффективных алгоритмов автоматического восстановления грамматики и вычисления степени отличия распознаваемого эталонного объекта.

Решение задачи синтаксического распознавания образов можно представить следующей схемой (рисунок 4):

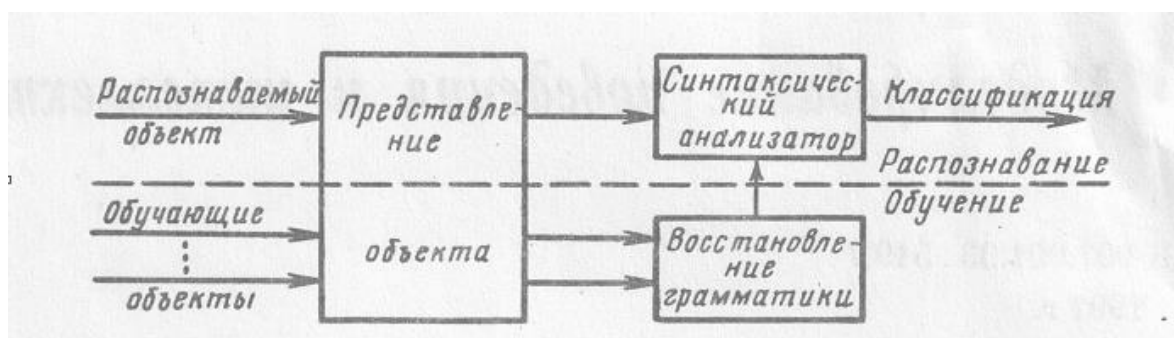


Рис.4. Схема синтаксического распознавания образов.

Решение задачи синтаксического распознавания образов в соответствии со схемой (рисунок 4) сводится к решению следующих подзадач: построение цепного года, синтаксический анализ, представление объектов и восстановление грамматики. Рассмотрим подробнее решение этих подзадач.

Построение цепного кода

- 1) Перевод каждого массива отсчетов в последовательность символьных описаний $T = (\{t_1, t_2, \dots, t_n\}, \Omega)$, где $t_i = c_1 c_2 \dots c_R | c_j \in \Omega$ (Ω – алфавит, t_i – цепочка в алфавите Ω);

Восстановление грамматики

- 2) Алгоритм данной подзадачи представлен в виде блок-схемы на рис.5. Построение по символьным описаниям обучающих объектов $T_{01}, T_{02}, \dots, T_{0l}$ (l – количество обучающих объектов) грамматики $G = (N, \Sigma, P, S)$, где N – множество нетерминальных символов, Σ – множество терминальных символов ($\Sigma = \Omega$), P – множество продукций вида: $p: \alpha \rightarrow \beta$, $\alpha \in (N \cup \Sigma)^* N (N \cup \Sigma)^*$, S – аксиома грамматики ($S \in N$). Здесь $(N \cup \Sigma)^*$ обозначает множество всех цепочек в алфавите $(N \cup \Sigma)$, включая пустую. Алгоритм первой подзадачи представлен в виде блок-схемы на рисунке 5.

Синтаксический анализ и классификация

- 3) Построение синтаксического анализа, реализующего функцию $A: (t_x \times G_i) \rightarrow \{\text{да, нет}\}$, где t_x – символьное описание распознаваемого объекта x , G_i – грамматика, порождающая язык $L(G_i)$, предложения которого интерпретируются как описания объектов i -го класса;
- 4) Собственно классификация объекта x по схеме: если $t_x \in L(G_i)$, то x классифицируется как один из объектов i -го класса, иначе x не принадлежит ни одному из известных классов.

На практике такая схема не используется, так как реальные объекты x редко удается точно отнести к тому или иному классу, например из-за наличия шумов.

Более гибкая техника предложена в [1]. В грамматику G_i , описывающую i -й класс, вводятся дополнительные продукции, обеспечивающие выводы произвольной цепочки. Это достигается допуском ряда искажений описаний эталонных объектов (удаление символов, замена символов, вставка дополнительных символов).

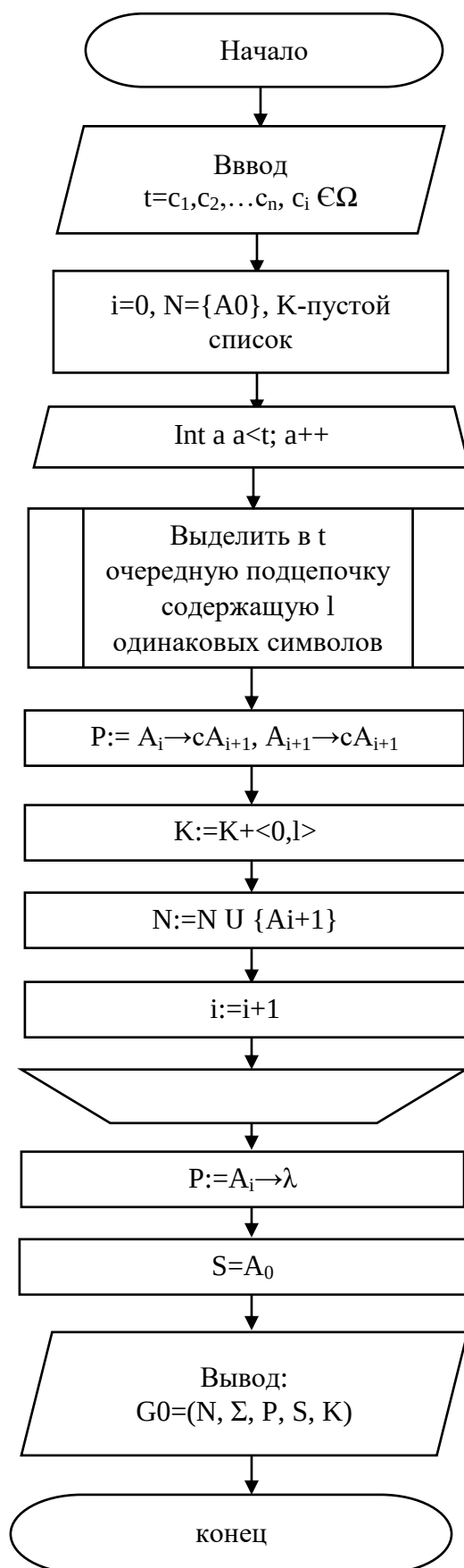


Рис. 5. Алгоритм 1, блок-схема.

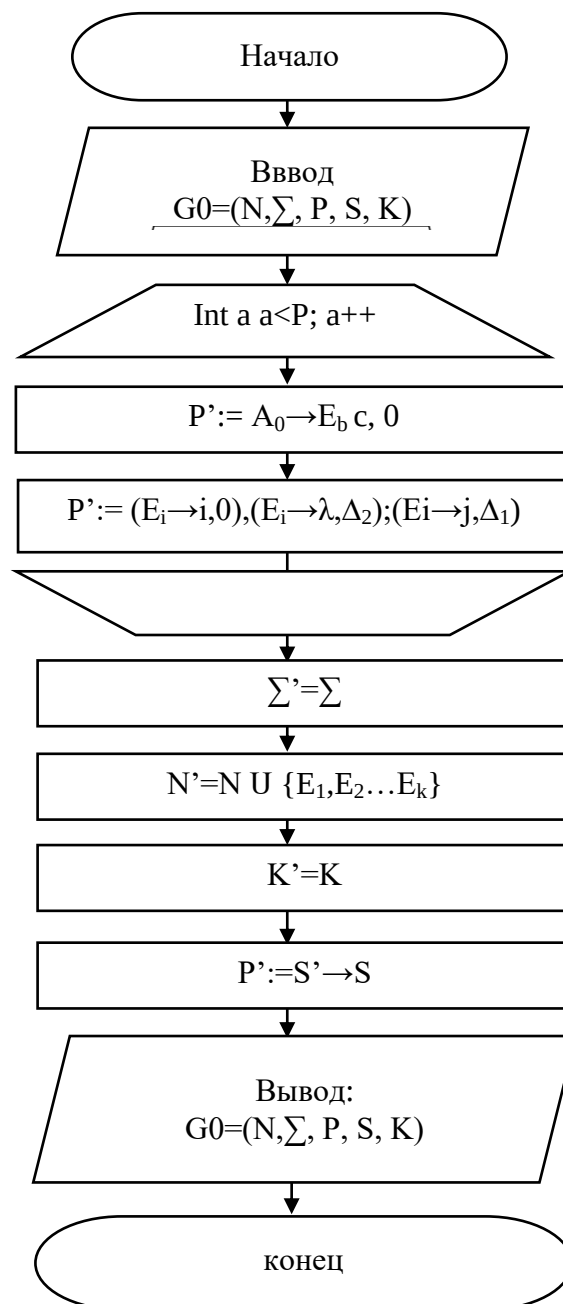


Рис. 6. Алгоритм 2, блок-схема.

Далее необходимо построить синтаксический анализатор, реализующий функцию $A: (t_x \times G_i) \rightarrow d_i(t_x, L(G_i))$.

Суть предлагаемых изменений заключается в следующем: так как продукции грамматики образов имеют простую повторяющуюся структуру, то шаги нисходящего разбора, выполняемые в соответствии с алгоритмом Эрли при анализе произвольной входной цепочки, проводят к построению наперёд заданного множества деревьев разбора, отличающихся весами вершин, характеризующих степень отличия входной цепочки от эталонной. Поэтому собственно разбор можно не выполнять, заменив его вычислением минимальной степени отличия, что существенно повышает эффективность анализа в целом. Алгоритмы модифицированного метода показаны на рисунке 6 и в алгоритме 3.

Алгоритм 3

Вход: расширенная грамматика $G_0' = (N', \Sigma', P', S', K')$, описание объекта

$t_x = c_1, c_2, \dots, c_n$

Выход: $d_i(t_x, L(G_i))$.

Метод:

1. $j:=n, p:=m$.
2. Для всех $i=1, 3, \dots, 2m-1$ выполнить
 $R_2(i):=p*\Delta_2, R_2(i+1):=p*\Delta_2, p:=p-1$.
3. Для всех $i=1, 2m$ вычислить

$$R_2(i) = \begin{cases} 0, & \text{если } A_{[i/2]} \rightarrow E_{cj} A_{[i/2]} \in P, \\ \Delta_i & \text{в противном случае,} \end{cases}$$

4. Для всех $i=2m, 2m-1, \dots, 1$ вычислить

$$R'_2(i) = \begin{cases} R_1(i), & \text{если } j = n \text{ и } i \geq 2m - 1, \\ R_2(k) + R_1(i), & \text{если } j < n \text{ и } i \geq 2m - 1 \\ \min(R_2(k), R_2(k+1)) + R_1(i) & \text{в противном случае,} \end{cases}$$

Где $k=i+i_{\text{mod}}$.

5. Для всех $i=2m, 2m-1, \dots, 1$ вычислить

$$R'_2(i) = \begin{cases} \min(\Delta_2, R'_2(i)), & \text{если } j = n \text{ и } i \geq 2m - 1, \\ \min(R'_2(k) + \Delta_2, R'_1(i)), & \text{если } j < n \text{ и } i \geq 2m - 1 \\ \min(\min(R'_2(k), R'_2(k+1)) + \Delta_2 R'_2(i) & \text{в противном случае,} \end{cases}$$

Где $k=i+i_{\text{mod}}$

6. $j:=j-1$. Если $j>0$, то перейти к п.3.
7. $d_i(t_x, L(G_i)):=R_2$. Конец.

В приведенном алгоритме изменение значений компонент векторов R_1 и R_2 отражает процесс вычисления степени отличия входной цепочки от эталонной.

Далее необходимо отнести x к одному из классов, при этом

$$t_x \in L(G_i) | d_i(t_x, L(G_i)) = \min(d_i(t_x, L(G_1)), \dots, d_k(t_x, L(G_k)))$$

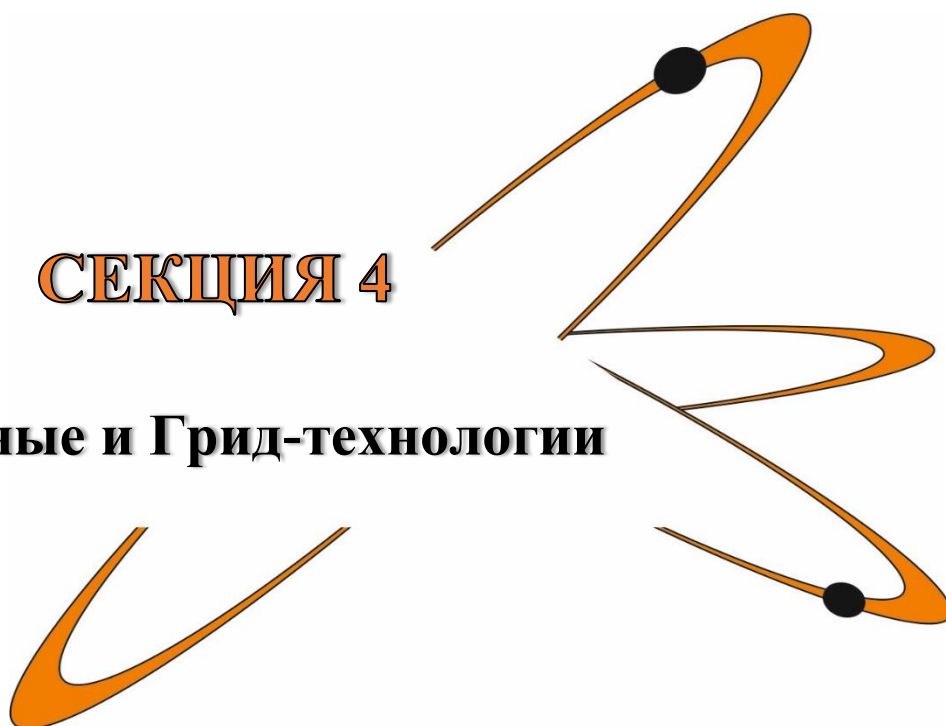
Заключение. Представленные алгоритмы были реализованы на языке java, в программной среде eclipse. Графическая обработка изображения была реализована на языке C в программной среде vim. Система распознавания образов была апробирована на микроскопических снимках лекарственного растительного сырья.

Библиографический список

1. Часовский А.А. Обзор алгоритмов кластеризации данных/ Часовский А.А. //Хабрахабр.–2010.
2. Корреляция цифровых изображений [Электронный ресурс]:–Режим доступа: [https://ru.wikipedia.org/wiki/ Корреляция_цифровых_изображений](https://ru.wikipedia.org/wiki/Корреляция_цифровых_изображений), свободный – Заглавие с экрана (дата обращения: 23.03.2018)
3. Ветров Д. В. Спектральный анализ сигналов и двумерных изображений/ Д. В. Ветров, И. А. Мизев // Пермский ГНИУ, 2009г. с.96-97.
4. OpenCV шаг за шагом. Нахождение контуров и операции с ними. [Электронный ресурс]: – Режим доступа: <http://robocraft.ru/blog/computervision/640.html>, свободный – Заглавие экрана (дата обращения: 23.03.2018)
5. Фу К. Структурные методы в распознавании образов/ К. Фу. – М.:Изд-во «Мир», 1977 – с.318
6. Сакович И. О.Обзор основных методов контурного анализа для выделения контура движущегося объекта /И.О. Сакович, Ю. С. Белов. // Инженерный журнал: наука и инновации.–2013.–с. 1-8.
7. OpenCV шаг за шагом. Обработка изображений детектором границ Кенни (Canny). [Электронный ресурс]: – Режим доступа: <http://robocraft.ru/blog/computervision/484.html>, свободный – Заглавие экрана (дата обращения: 23.03.2018)

СЕКЦИЯ 4

Облачные и Грид-технологии



УДК 004:65

А.Э. Белоусова, магистрант**В.И. Инюшин, магистрант****М.В. Потанина, канд. техн. наук, доц.***Севастопольский государственный университет***ПЕРСПЕКТИВЫ ИСПОЛЬЗОВАНИЯ SECAAS-ТЕХНОЛОГИЙ В РОССИИ**

Аннотация: В данной работе рассматривается актуальность использования «облачных» технологий в сфере информационной безопасности. Дана краткая характеристика «SECaaS»-решений, перечислены их основные и дополнительные компоненты, а также преимущества и недостатки. Приведен краткий обзор рынка «SECaaS»-решений в России, а также обозначены перспективы их использования.

Ключевые слова: «облачные» технологии, информационная безопасность, «SECaaS», антивирус.

Annotation: In this paper, the relevance of the use of "cloud" technologies in the field of information security is considered. A brief description of the SECaaS solutions is given, their main and additional components are listed, as well as advantages and disadvantages. A brief overview of the SECaaS market in Russia is given, and prospects for their use are outlined.

Keywords: "Cloud" technologies, information security, "SECaaS", antivirus.

С развитием Интернет-технологий в различных отраслях бизнеса прочно закрепились так называемые «облачные» сервисы, которые позволяют работать с большими объемами данных, осуществлять их обработку и хранение с помощью удаленного доступа, осуществляемое через клиентское приложение. Данный вид сервиса, получивший название «SaaS» (от англ. «Software as a Service» - «Программное обеспечение как услуга») обрёл широкую популярность вследствие того, что позволяет предприятию значительно сократить издержки на содержание и обслуживание аппаратного и программного обеспечения, что в реалиях современного бизнеса является веским преимуществом. В настоящее время трудно найти фирму или предпринимателя, которые бы не использовали «SaaS» в своей деятельности.

Однако, с развитием «SaaS» получила своё развитие одна из серьезных проблем – проблема информационной безопасности «облачных» сервисов. Ежедневно пользователи этих сервисов сталкиваются с проблемой конфиденциальности, утечки данных, хакерскими атаками, что влечёт за собой значительный моральный или материальный ущерб.

Решению данной проблемы, начали уделять пристальное внимание относительно недавно. Специалисты, изучавшие проблему информационной безопасности «облачных» сервисов вскоре вывели на рынки собственное решение, получившее название «SECaaS» (от англ. «Security as a Service» - «Безопасность как услуга»). Вскоре свои решения в данной сфере представили вендоры мировых масштабов, например: «Akamai», «Barracuda Networks», «Cisco», «Google/Postini», «McAfee», «Mimecast», «Qualys», «Symantec», «Symplified», «TriCipher», «Veracode», «Webroot», «WhiteHat Security», «Zscaler».

Что же из себя представляют «SECaaS»-решения?

Это «бизнес-модель построенная таким образом, что крупный сервис-провайдер интегрирует все имеющиеся сервисы безопасности в единую корпоративную инфраструктуру, что является экономически более выгодно и эффективно, нежели большинство индивидуальных фирм или корпораций предоставят их самостоятельно с учетом совокупной стоимости данных сервисов» [1].

Таким образом, «SECaaS» представляет собой ответвление «SaaS»-технологий, которое позволяет передать процесс решения возникающих проблем информационной безопасности – провайдеру или разработчику, которые методом аутсорсинга решают поставленные перед ними задачи, тем самым сняв данные обязанности с сотрудников IT-отдела предприятия.

Предприятие по своему усмотрению может также передать управление безопасностью данных – в так называемое «частное облако», обслуживанием которого полностью занимаются вендоры (разработчики и поставщики программного обеспечения) предоставляющие данную услугу. Пользователю больше не нужно приобретать

дорогостоящее программное обеспечение (ПО), т.к. «SECaaS» распространяется в виде ежемесячной или ежегодной подписки, которая позволяет на регулярной основе получать актуальные обновления черных списков и антиспам-фильтров.

Пользователь сам в праве выбирать надстройки и компоненты данного сервиса, в зависимости от сферы бизнеса и необходимого функционала. Компоненты «SECaaS»-решений могут быть разделены на две группы: основные и дополнительные.

К основным компонентам относятся:

- защита от вирусов;
- защита от спама;
- блокирование «DDoS-атак» (т.н. «отказы в обслуживании»);
- защита данных на мобильных устройствах на базе «IOs» или «Android»;
- оценка безопасности системы.

К дополнительным компонентам относятся:

- управление правами доступа и учетными записями пользователей;
- предотвращение потери данных (резервное копирование);
- шифрование данных;

Далее будут рассмотрены основные преимущества и недостатки использования «SECaaS»-решений в бизнесе:

К преимуществам данных решений можно отнести следующее:

- регулярное обновление баз данных антивирусных программ, а также своевременное внесение вредоносных URL-адресов в «черные списки» выполняется без вмешательства пользователя;

- уровень навыков специалистов, предоставляющих «SECaaS»-решения, зачастую гораздо выше IT-специалистов предприятия за счет узкоспециализированной направленности их деятельности;

- обеспечение информационной безопасности предприятия методом передачи её на аутсорсинг значительно сокращает временные и материальные затраты на содержание аппаратного и программного обеспечения, а также ответственных специалистов, что в конечном итоге позволяет максимизировать получаемую прибыль предприятия;

Недостатки:

- недостаточное правовое регулирование деятельности «SECaaS»-сервисов в силу того, что рынок «облачных» технологий появился относительно недавно и находится в постоянном развитии;

- передача вопросов информационной безопасности предприятия сторонней организации является трудным для принятия управленческим решением и не каждое предприятие соглашается на внедрение данной бизнес-модели;

- недостаточная осведомленность пользователей как о механизмах функционирования данных сервисов, так и об их существовании в целом [2].

В России, как и по всему миру, всё чаще используются облачные технологии, также растёт число пользователей SaaS-доступа, так как это удобнее и чаще всего выгодней.

Для того чтобы количество пользователей в России увеличивалось, поставщикам данной услуги необходимо выполнять следующие требования:

1) соответствие стандартным требованиям (наличие демо-версии; быстрое развёртывание; быстрое время отклика программы; своевременные и постоянные обновления программы; определённая стоимость для заказчика, вариация предоставляемых функциональных пакетов);

2) наличие соответствующих лицензий (не редкость среди SecaaS-провайдеров сертификация по ISO27001, «PCI DSS» - (от англ. «Payment Card Industry Data Security Standard» — стандарт безопасности данных индустрии платёжных карт, разработанный Советом по стандартам безопасности индустрии платёжных карт);

- 3) выполнение работы с заранее прописанными обязательствами и ответственностями заказчика;
- 4) SLA (Service Level Agreement – Соглашение об уровне предоставления услуги) по каждому виду услуг;
- 5) контракты поддержки вендоров (компаний-производителей);
- 6) обеспечение требований нормативно-правовых актов;
- 7) страхование рисков [3].

Основными потребителями Sesaas-решения являются предприятия, которые используют:

- 1) сотрудников, работающих удалённо или же мобильно с ноутбуками, планшетами или смартфонами;
- 2) SaaS-бизнес-приложения (являются активными пользователями, например, SaaS IDM и Access Management для управления учетными записями в SaaS-системах);
- 3) облачные серверы, гибридные облака (совмещено несколько видов из частных, публичных и общественных) и IaaS (инфраструктура как сервис – является одной из самых успешных интеграций с Sesaas);
- 4) динамичную и масштабируемую инфраструктуру веб-приложений [4].

Международный сервис-провайдер Orange Business Services и исследовательская компания International Data Corporation (IDC) в феврале 2018 года сообщили о проведении анализа корпоративного рынка услуг кибербезопасности в России. Было рассмотрено несколько основных сегментов, однако, основной для нас является облачные корпоративные услуги безопасности. Спрос на услуги корпоративной кибербезопасности в России наиболее высок в финансах, промышленности и энергетике, также можно видеть из рисунка 1, что ежегодно востребованность данных услуг будет возрастать.

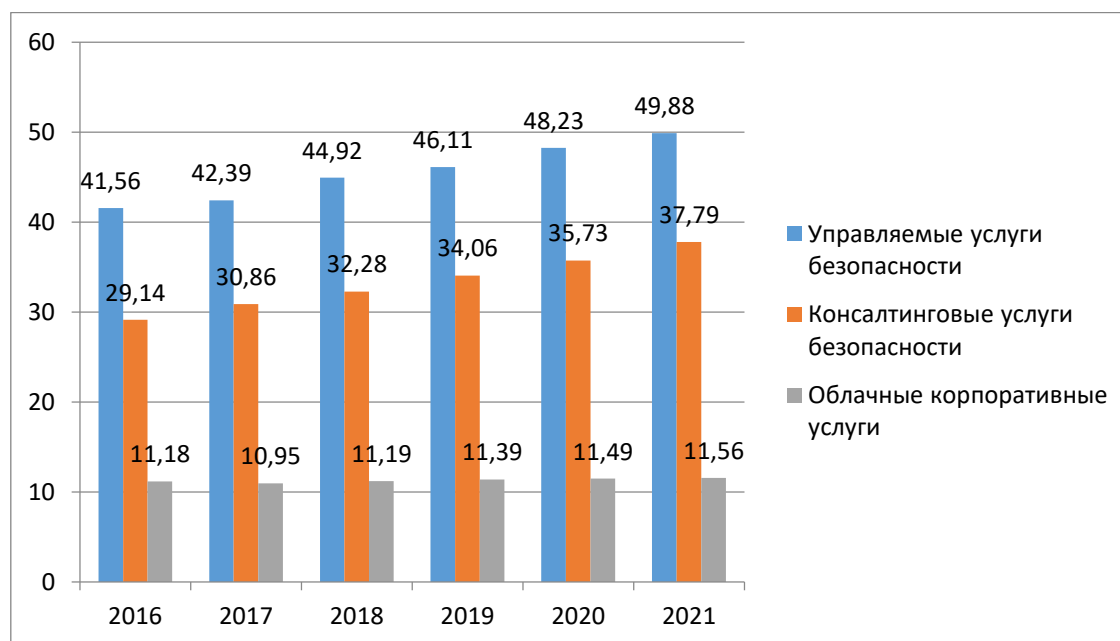


Рис. 1. Объем и прогноз развития российского рынка услуг безопасности (млн. долларов)

Корпоративная безопасность актуальна и для ритейла. Интерес к правовым услугам в оптово-розничной торговле обусловлен тенденцией этой отрасли к аутсорсингу экспертизы по ИТ-безопасности, а также растущими угрозами, связанными с применением новых технологий в торговле и развитием сети электронной коммерции.

Первым Sesaas-провайдер в России и СНГ и лидером на данный момент является компания «ProtoSecurity». Данная компания имеет опыт свыше 10 лет на рынке, а также ведёт работу по всему миру, на рисунке 2 представлены решения данной компании [5].

	ЗАЩИТА ВЕБ-САЙТОВ	Защита от DDoS: веб-сайта и сервера, DNS, инфраструктуры, IP	Web Application Firewall	Защита от ботов
		Двухфакторная аутентификация	CDN и кэширование	Балансировка нагрузки
	МОНИТОРИНГ ПРОИЗВОДИТЕЛЬНОСТИ	Мониторинг производительности и ошибок Frontend и Backend	Мониторинг взаимодействия пользователей с приложением	Мониторинг инфраструктуры
		Мониторинг мобильных приложений	Синтетический мониторинг	Аналитическая платформа
	АУДИТ БЕЗОПАСНОСТИ	Аудит безопасности сайта (DAST анализ)	Аудит исходного кода (SAST анализ)	Аудит безопасности мобильных приложений
	APPDYNAMICS	AppDynamics Application Performance Management	AppDynamics End-User Monitoring	AppDynamics Infrastructure Visibility
		AppDynamics IQ Platform		

Рис. 2. Решения компании «ProtoSecurity»

Таким образом, можно сделать вывод о том, что вопрос использования Sesaas-решения является актуальным для Российской Федерации, и в ближайшее время будет пользоваться спросом у предприятий, которые заботятся о своей информационной безопасности.

Библиографический список

1. Глухова Т.В., Горина Е.В., Ручина О.М. Актуальность использования Security as a Service в современных реалиях. // Инновации в науке: сб. ст. по матер. LI междунар. науч.-практ. конф. № 11(48). Часть II. – Новосибирск: СибАК, 2015.
2. Борисова Е.С., Долгова Т.Г. Актуальность использования бизнес-модели secaaas // Актуальные проблемы авиации и космонавтики. 2014. №10. - [Электронный ресурс] – Режим доступа: <https://cyberleninka.ru/article/n/aktualnost-ispolzovaniya-biznes-modeli-secaaas> (дата обращения: 23.03.2018).
3. Черемнов Д.В. Актуальность использования бизнес-модели «SECaaS» - [Электронный ресурс] – Режим доступа: http://sci-article.ru/stat.php?i=aktualnost_isspolzovaniya_biznes-modeli_SECaaS (Дата обращения: 22.03.2018)
4. Пудов Д. Рынок сервисов информационной безопасности - [Электронный ресурс] – Режим доступа: https://www.anti-malware.ru/analytics/Market_Analysis/market_services_information_security#part2 (Дата обращения: 22.03.2018)
5. Сайт компании «ProtoSecurity» - [Электронный ресурс] – Режим доступа: <https://protosecurity.ru/> (Дата обращения: 22.03.2018).

УДК 004:65

Д.Д.Капустина, студентка 4-го курса

М.В. Потанина, канд. техн. наук, доцент

Севастопольский государственный университет

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ОБЛАЧНЫХ ТЕХНОЛОГИЙ В ОБРАЗОВАНИИ

Аннотация: В работе анализируются особенности использования облачных технологий в образовательном процессе. Рассмотрены основные виды облачных технологий, возможности их применения в образовательном процессе, преимущества и недостатки использования.

Ключевые слова: облачные технологии, облако, сервис, компьютерная инфраструктура, платформа, образование.

Annotation: The paper analyzes the features of using cloud technologies in the educational process. The main types of cloud technologies, their application in the educational process, advantages and disadvantages of use are considered.

Key words: cloud technologies, cloud, service, computer infrastructure, platform, education.

В современном мире информационные технологии набирают все большую популярность во всех сферах деятельности человечества. Не исключением является и образование. Процесс информатизации и усовершенствования технологий в сфере обучения занимает одно из главных мест среди большого количества новых направлений развития образования.

В настоящее время имеет место тенденция развития образования за счет информатизации и компьютеризации, что подразумевает под собой внедрение информационных технологий, оснащение учебных учреждений вычислительной техникой и постоянным доступом к сети интернет.

На сегодняшний день облачные технологии находят активное применение во многих развитых странах, позволяя обеспечить новые, экономические эффективные возможности для образования, научных исследований и предпринимательской деятельности.

В связи с этим, изучение вопросов связанных с применением облачных технологий в сфере образования является достаточно актуальным.

Облачные технологии представляют собой технологии обработки данных, при этом, ресурсы предоставляются пользователю через онлайн-сервис, т.е. нет необходимости постоянно носить с собой сменные электронные носители типа «флешек», бумажные документы и т.п., т.к. все информация храниться в «облачном хранилище» [1].

Для создания такого «облака» используются ряд базовых моделей [2]:

- инфраструктура как сервис;
- платформа как сервис;
- программное обеспечение (ПО) как сервис.

Проведем краткий анализ базовых «облачных» моделей:

Первая базовая модель – инфраструктура как сервис (IaaS, infrastructure as a service), является процессом предоставления компьютерной инфраструктуры как услуги, на основе концепции облачных вычислений. На данном этапе пользователи получают базовые вычислительные ресурсы. Данная модель предоставляет возможность бесплатного хранения данных, использования функций электронной почты и функцию совместной работы, что может быть полезно для образовательного процесса.

Вторая базовая модель - платформа как сервис (PaaS, platform as a service), представляет набор определенных услуг на пользовательском уровне, доступ к программному обеспечению. В данном случае, пользователи имеют возможность самостоятельно устанавливать приложения на платформе. Данный вид модели не очень распространён в сфере образования.

Третья базовая модель – программное обеспечение как услуга (SaaS, software as a service). Данная модель дает возможность пользователю использовать прикладное ПО, которое функционирует в облачной среде и является доступным из разных устройств

(браузера или посредством интерфейса программы). Такая модель представляет большой интерес для образовательных учреждений.

На сегодняшний день на практике чаще всего используются следующие виды облачных систем:

- частное облако, которое представляет собой инфраструктуру, предназначенную для использования одной фирмой, состоящей из нескольких подразделений. Такое облако может находиться как в собственности организации, так и третьей стороны.

- публичное облако – представляет собой развертывание инфраструктуры со всем необходимым ПО и предоставление доступа к ним как в пределах организации, так и за ними.

- общественное облако – предназначено для использования конкретной группой пользователей из организации, для решения общих задач. Такое облако может находиться в собственности одной или более из организаций сообщества или третьей стороны.

- гибридное облако представляет собой комбинацию из двух и более различных видов облачных структур, которые в свою очередь остаются уникальными, но связанными между собой.

Использование облачных технологий позволяет организовать доступ к различным видам ПО, используя преимущества этого вида информационных технологий в образовательном процессе. Такие технологии могут послужить платформой для организации мобильного обучения. Так же использование «облачных» технологий может снизить затраты на обучение, повысить качество обучения за счет оперативного изменения учебных планов, формирования и предоставления общего доступа к необходимым для обучения материалам.

Исследователи выделяют ряд преимуществ применения облачных вычислений в процессе образования [3]:

- 1) Возможность резервного копирования.
- 2) Возможность хранения информации всех типов.
- 3) Высокая доступность.
- 4) Облако предоставляет возможность работы с несколькими пользователями одновременно.
- 5) Сокращение времени и ресурсов на копирование учебных материалов.
- 6) Низкая стоимость.
- 7) Высокая гибкость облачных технологий.
- 8) Высокая надежность и безопасность.

Недостатков применения облачных технологий достаточно немного, к ним можно отнести в основном проблемы, имеющие технологический характер. Например, относительно ограниченный функционал, по сравнению с локальными системами, отсутствие российских представителей облачных сервисов, а также отсутствие нормативно-правовой и законодательной базы, регламентирующих применение облачных технологий.

Развитие облачных сервисов с каждым днем ускоряется и совершенствуется. На сегодняшний день существует большое количество публичных сервисов, предоставляющих возможность работать с офисными предложениями. Одним из наиболее известных облачных продуктов является Microsoft Office 365, получивший свое распространение в 2011 году. Office 365 – это продукт, созданный по принципу «ПО + услуги» на основе модели SaaS. Перечень услуг, которые предлагает данный продукт, достаточно широк: электронная почта, облачное хранилище данных, пакет офисных программ и ряд других возможностей, состав которых зависит от вида выбранной подписки.

Одним из подвидов Office 365 является продукт, предназначенный для использования в образовательных учреждениях - Office 365 University [4].

Существует ряд особенностей применения Office 365 University в высших учебных заведениях:

- организация совместной работы преподавателей и студентов над документами. В процессе использования данного ПО появляется возможность совместной работы над курсовыми, рефератами, научными статьями и т.п. в режиме реального времени, корректировка документов преподавателем и одновременное исправление ошибок студентом.

- расписание занятий – онлайн. Возможность оценить занятость преподавателей по календарям и моментальное оповещение студентов о возможных изменениях в расписании.

- проведение совместных мероприятий. Данный сервис позволяет проводить конференции, лекции, собрания в режиме онлайн, для аудитории до 1500 человек [5].

На сегодняшний день, облачные технологии представляют собой инструмент, которым пользуется каждый человек практически каждый день. Такие сервисы как электронная почта, Интернет-торговля, сетевые игры и другие онлайн развлечения, представляют собой примеры распространённых облачных технологий. Облачные технологии с каждым днем набирают все большую популярность, стремительно развиваются и охватывают разные сферы деятельности.

Такое широкое распространение и развитие облачных технологий предоставляет возможность интегрирования облачных сервисов в систему образовательных учреждений. Использование таких технологий в этой сфере подходит для персонального обучения, группового преподавания и интерактивных занятий. Главным преимуществом внедрения и использования облачных технологий в образовательном процессе является не только снижение затрат на приобретение лицензионно программного обеспечения, повышение эффективности образовательного процесса, но и более качественная подготовка студентов и учеников к современной жизни в информационном обществе.

Библиографический список

1. Сейдаметова З.С., Сейтвелиева С.Н. Облачные сервисы в образовании // Информационные технологии в образовании, 2016, №9.
2. Соснин В.В. Облачные вычисления в образовании. - URL: http://www.intuit.ru/goods_store/ebooks/9155 (дата обращения: 22.03.2018).
4. Сорочинский М. А. Особенности использования облачных технологий в вузе на основе Office 365 University // Научно-методический электронный журнал «Концепт». – 2015. – Т. 26. – С. 141–145. – URL: <http://e-koncept.ru/2015/95313.htm> (дата обращения: 22.03.2018).
5. Емельянова О. А. Применение облачных технологий в образовании [Текст] / О. А. Емельянова // Молодой ученый. — 2014. — №3. — С. 907-909.
6. Microsoft Office [Электронный ресурс] – Режим доступа: <https://products.office.com> (дата обращения: 22.03.2018)

УДК 004.94; 378.1

М.С. Лисовский, магистрант 2-го курса**Д.Ю. Воронин, канд. техн. наук, доцент кафедры ИТиКС***Севастопольский государственный университет***ОРГАНИЗАЦИЯ УПРАВЛЕНИЯ АКТОРНЫМ РАВНОВЕСИЕМ В ОБЛАЧНЫХ СЕРВИСАХ КРИТИЧЕСКОГО ПРИМЕНЕНИЯ***Аннотация: Рассматривается задача организации управления акторным равновесием в облачных сервисах критического применения. Предлагаемая информационная технология обеспечивает необходимую точность и качество переходного процесса, что гарантирует требуемую реактивность принимаемых решений.**Ключевые слова: облачная среда, имитационное моделирование, референтная модель, системы критического применения, акторное равновесие**Annotation: The problem of organization of actor equilibrium management in cloud services of critical application is considered. The proposed information technology provides the necessary accuracy and quality of the transient process, which guarantees the required reactivity of the decision-making process.**Key words: cloud environment, simulation modeling, reference model, critical application systems, actor equilibrium*

При решении проблем эффективной организации облачного компьютеринга (ОК) использование классического системного подхода [1] весьма ограничено. Известно [2 – 6], что в развитых облачных средах появляются свои особенности, которые требуют привлечения инновационных подходов и технологий. Системообразующие факторы, безусловно, существуют, но употребление централизованных понятий для описания сценариев функционирования ОК обладает весьма слабой конструктивностью. Альтернативным вариантом описания ОК является референтная модель, которая описывает взаимодействие следующих базовых Акторов [5]: Потребитель, Провайдер, Брокер, а также расширенных [6, 7]: Регулятор, Кризисный Менеджер. Их функциональные особенности и основные сценарии взаимодействия приведены в [5 – 7]. Каждый из них имеет свой относительный характер поведения, свои цели, а качество организации ОК во многом определяется коллинеарностью этих целей – то есть эффективностью принимаемых управленческих решений по нахождению равновесия между интересами различных базовых Акторов при реализации услуг сервисов ОК. Далее для лаконичности изложения такое равновесие будем называть А-равновесием. В докладе рассматривается информационная технология управления А-равновесием, которая ориентирована на разрешение противоречий вида «Потребитель-Брокер-Провайдер», «Брокер-Регулятор». Ввиду нетривиальности подходов к разрешению этих противоречий задача синтеза информационной технологии управления А-равновесием является актуальной.

Исходными данными предлагаемой информационной технологии являются параметры, характеризующие взаимодействия Акторов в рамках рассматриваемой услуги сервиса ОК. Результат ее функционирования – найденное управление А-равновесием при реализации сервиса ОК. Различают следующие функциональные блоки (ФБ):

Φ_B – формирует цены на покупку и продажу услуги Брокером;

Φ_C – оценивает объем услуги сервиса, которую готов купить Потребитель при складывающейся цене;

Φ_P – оценивает объем услуги сервиса, которую готов продать Брокер при складывающейся цене;

Φ_R^c – формирует управляющие воздействия Регулятора по коррекции «базовой» цены Брокера;

Φ_R^z – формирует управляющие воздействия Регулятора по коррекции «наценки» Брокера.

Целевое назначение предлагаемой информационной технологии состоит в обеспечении А-равновесия для фиксированной услуги рассматриваемого сервиса ОК. Иными словами, она реализует отображение $\mathfrak{Z}^{UA} \rightarrow u^{\text{равн}}$, где $\mathfrak{Z}^{UA} = \langle c^0, z^0, \Phi_B, \Phi_C, \Phi_P, \Phi_R^c, \Phi_R^z \rangle$ – кортеж, описывающий информационное обеспечение рассматриваемой задачи;

c^0 – начальная цена, сформированная без учета интересов Брокера;

z^0 – начальное значение «маржи» Брокера;

$u^{\text{равн}} = \langle c_P^{\text{равн}}, c_C^{\text{равн}} \rangle$ – кортеж, описывающий найденное управление А-равновесием для текущего момента времени;

$c_P^{\text{равн}}$ – цена, обеспечивающая А-равновесие между Брокером и Провайдером;

$c_C^{\text{равн}}$ – цена, обеспечивающая А-равновесие между Брокером и Потребителем.

Формализация функциональных требований описывается системой (1).

$$\left\{ \begin{array}{l} \langle c_P(t), c_C(t) \rangle = \Phi_B \left(\frac{dc}{dt}, \frac{dz}{dt} \right); \\ \frac{dc}{dt} = \Phi_R^c(c^0, W(V_C, V_P, t), u_R^c); \\ \frac{dz}{dt} = \Phi_R^z(z^0, W(V_C, V_P, t), u_R^z); \\ V_C(t) = \Phi_C \left(c_C, \frac{dc}{dt}, \frac{dz}{dt} \right); \\ V_P(t) = \Phi_P \left(c_P, \frac{dc}{dt}, \frac{dz}{dt} \right); \\ u_R^c = \lambda W(V_C, V_P, t) + \beta \int_0^T W(V_C, V_P, t) dt + \gamma \frac{dW(V_C, V_P, t)}{dt}; \\ u_R^z = \begin{cases} -\alpha W(V_C, V_P, t), & \text{при } W(V_C, V_P, t) \geq W_{\text{ПОР}}; \\ \alpha(V_C, V_P, t), & \text{при } W(V_C, V_P, t) < W_{\text{ПОР}}; \end{cases} \\ \text{при начальных условиях } \langle c^0, z^0 \rangle. \end{array} \right. \quad (1)$$

где c_P – цена, сформированная Брокером для покупки услуги у Провайдера;

c_C – цена, сформированная Брокером для продажи услуги Потребителю;

$\frac{dc}{dt}$ – производная, на основе которой выполняется коррекция цены за единицу объема услуги сервиса, сформированная без учета интересов Брокера в соответствии с корректирующими управляющими воздействиями, полученными от Регулятора;

$\frac{dz}{dt}$ – производная, на основе которой формируется «маржа» Брокера за единицу объема услуги сервиса, сформированная в Регулятором и Брокером.

W – объем продаж (системная характеристика);

V_C – объем услуги сервиса, которую готов приобрести Потребитель при цене c_C ;

V_P – объем услуги сервиса, которую готов продать Провайдер при цене c_P ;

u_R^c – управляющие воздействия Регулятора по коррекции цены c ;

u_R^z – управляющие воздействия Брокера/Регулятора по коррекции маржи z ;

λ, β, γ – коэффициенты усиления пропорциональной, интегрирующей и дифференцирующей составляющих ПИД-регулятора, которые могут быть найдены при использовании процедур адаптивного выбора вариантов или применении нейронных сетей;

α – коэффициент Alfa-регулятора (задает ЛПР);

$W_{\text{ПОР}}$ – пороговое значение объема продаж W для Alfa-регулятора (задает ЛПР).

На рисунке 1 представлена организационно-функциональная структура информационной технологии управления равновесием Акторов при облачном компьютинге. Она является двухканальной: прямой канал реализуется функциональными блоками Φ_B, Φ_C, Φ_P ; обратная связь – блоками Φ_R^c и Φ_R^z . Предлагаемая информационная технология реализована с использованием системно-динамической библиотеки среды имитационного моделирования AnyLogic, причем формирование и осуществлено при помощи различных процедур поиска

управления А-равновесием (ПИД-регулятор или Alfa-регулятор). Начальные условия $\langle c^0, z^0 \rangle$ задает ЛПР.

Динамика изменения V_C и V_P описывается следующим образом:

$$V_C^{i,j}(t, \Delta t) = V_C^{i,j}(t_0, \Delta t) \cdot \alpha^{i,j}(t) + c_C^{i,j}(t_0, \Delta t) \cdot \beta^{i,j}(t);$$

$$V_P^{i,j}(t, \Delta t) = V_P^{i,j}(t_0, \Delta t) \cdot \gamma^{i,j}(t) + c_P^{i,j}(t_0, \Delta t) \cdot \mu^{i,j}(t),$$

где $\alpha^{i,j}(t)$, $\gamma^{i,j}(t)$ – коэффициенты, учитывающие "инертность" Акторов при рассмотрении j-ой услуги i-ого сервиса ОВС;

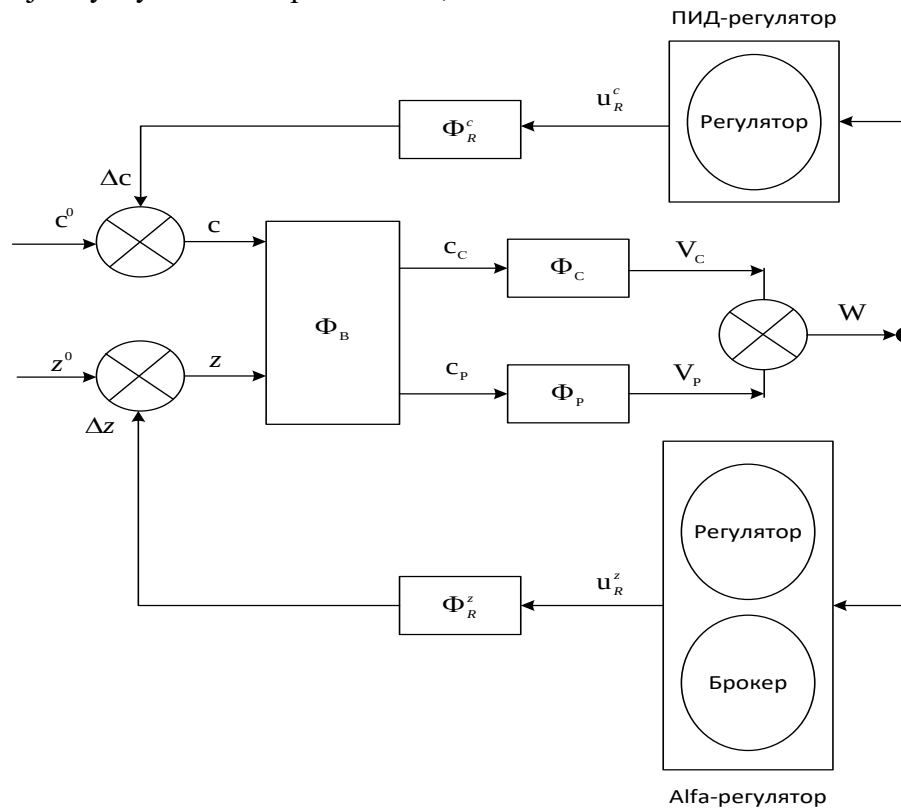


Рис. 1. Организационно-функциональная структура информационной технологии информационной технологии управления равновесием Акторов при облачном компютинге

$\beta^{i,j}(t)$, $\mu^{i,j}(t)$ – коэффициенты, учитывающие "эластичность" Акторов при рассмотрении j-ой услуги i-ого сервиса ОВС.

Очевидно, что:

$$\frac{\Delta V_C^{i,j}(t)}{\Delta t} = V_C^{i,j}(t_0) \cdot \alpha^{i,j}(t) + c_C^{i,j}(t) \cdot \beta^{i,j}(t);$$

$$\frac{\Delta V_P^{i,j}(t)}{\Delta t} = V_P^{i,j}(t_0) \cdot \gamma^{i,j}(t) + c_P^{i,j}(t) \cdot \mu^{i,j}(t).$$

Таким образом, динамика спроса Потребителя описывается системой дифференциальных уравнений (2) при начальных условиях (3). Для Провайдера предложение услуг сервисов ОВС задается системой дифференциальных уравнений (4) при начальных условиях (5).

$$\begin{cases} \frac{dV_C^{1,1}(t)}{dt} = \Phi_C(1,1, \alpha^{1,1}(t), \beta^{1,1}(t), t); \\ \dots \\ \frac{dV_C^{i,j}(t)}{dt} = \Phi_C(i, j, \alpha^{i,j}(t), \beta^{i,j}(t), t); \\ \dots \\ \frac{dV_C^{n,m}(t)}{dt} = \Phi_C(n, m, \alpha^{n,m}(t), \beta^{n,m}(t), t). \end{cases} \quad (2)$$

$$\begin{cases} V_c^{1,1}(t_0) = \eta_c^{1,1}; \\ \dots \\ V_c^{i,j}(t_0) = \eta_c^{i,j}; \\ \dots \\ V_c^{n,m}(t_0) = \eta_c^{n,m}. \end{cases} \quad (3)$$

$$\begin{cases} \frac{dV_p^{1,1}(t)}{dt} = \Phi_p(1,1,\gamma^{1,1}(t),\mu^{1,1}(t),t); \\ \dots \\ \frac{dV_p^{i,j}(t)}{dt} = \Phi_p(i,j,\gamma^{i,j}(t),\mu^{i,j}(t),t); \\ \dots \\ \frac{dV_p^{n,m}(t)}{dt} = \Phi_p(n,m,\gamma^{n,m}(t),\mu^{n,m}(t),t). \end{cases} \quad (4)$$

$$\begin{cases} V_p^{1,1}(t_0) = \eta_p^{1,1}; \\ \dots \\ V_p^{i,j}(t_0) = \eta_p^{i,j}; \\ \dots \\ V_p^{n,m}(t_0) = \eta_p^{n,m}. \end{cases} \quad (5)$$

Проведенные вычислительные эксперименты подтвердили эффективность предлагаемого подхода для решения задач управления акторным равновесием при облачном компьютинге. Планируется расширить сферу ее применения путем включения других Акторов при анализе процессов предоставления услуг по технологии облачного компьютинга.

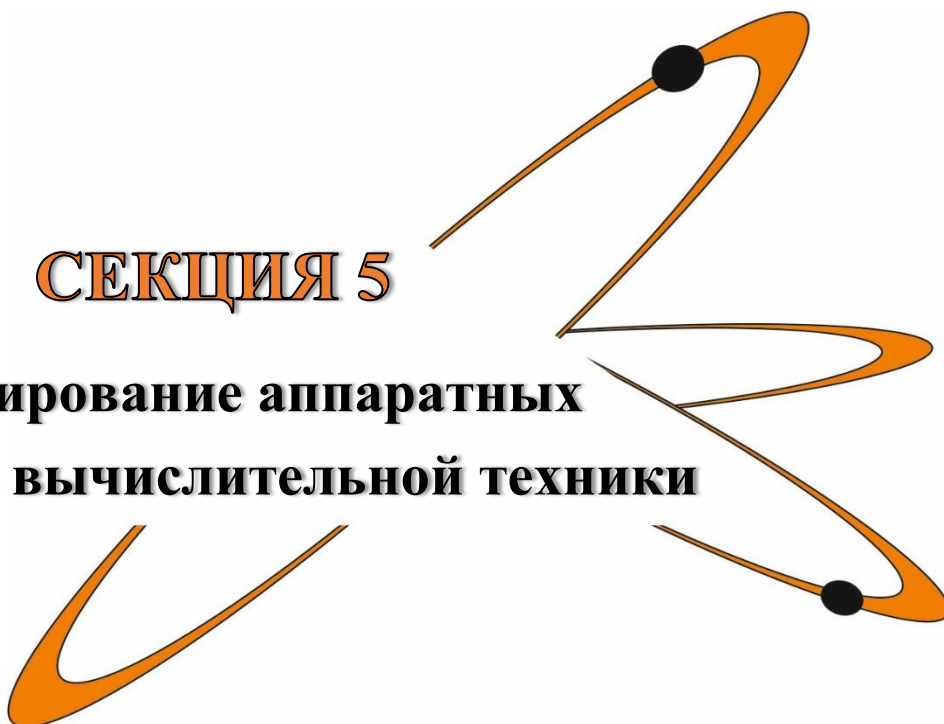
Работа выполнена при частичной поддержке Российского фонда фундаментальных исследований (грант № 15-29-07936).

Библиографический список

1. Прангишвили И. В. Системный подход и общесистемные закономерности. – М: Синтег, 2000. – 528 с.
2. Buyya R., Broberg J., Goscinski A. Cloud Computing: Principles and Paradigms. – New York: Wiley Press, 2011. – pp. 1 – 443.
3. Skatkov A. V., Shevchenko V. I. and Voronin D.Y. Game-theoretical Management Model for IT-services of ERP-systems Guaranteed Level Assurance in Cloud Environments // Proceedings of the 5th IEEE International Conference on Informatics, Electronics & Vision, 2016, Dhaka, Bangladesh, pp. 1113 – 1116. doi: 10.1109/ICIEV.2016.7760172.
4. Vecchiola C., Pandey S. and Buyya R. High-performance cloud computing: a view of scientific application// ISPAN 2009: Proceeding of 10th International Symposium on Pervasive Sytems, Algorithms, and Network, Kaohsiung, 2009, Taiwan, pp. 4 – 16.
5. Liu F., Tong J., Mao J., Bohn R., Messina J., Badger L., Leaf D. NIST Cloud Computing Reference Architecture. Recommendations of the National Institute of Standards and Technology. September 2011 (http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505).
6. Skatkov A.V., Maschenko E.N., Shevchenko V.I., Voronin D.Y. Actors Interactions Research in Cloud Computing Environments Using System Dynamics Methodology// Proceedings of the 18th FRUCT & ISPIT Conference, 2016, pp. 612 – 619.
7. Шевченко В.И. Концептуальная модель взаимодействия агентов облачного сервиса при проведении крупномасштабных научных исследований/ Шевченко В. И., Скатков А.В., Николаева Ю.П. // Материалы межд. научно-практ. конференции "Инфотех-2015". — Севастополь: изд-во СевГУ, 2015. — С. 45 – 46.

СЕКЦИЯ 5

Проектирование аппаратных средств вычислительной техники



УДК 004.3:4:5:9

А.А. Галасюк, студент 4-го курса

Ю.Л. Явкун, ст. преподаватель

Севастопольский государственный университет

ПРОГРАММНОЕ И АППАРАТНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ АВТОМАТИЗАЦИИ РАБОТЫ ИНФУЗИОННОЙ СИСТЕМЫ ПОДАЧИ ПРЕПАРАТА (КАПЕЛЬНИЦЫ)

Аннотация: значительная часть инновационных исследований и разработок в области медицинского оборудования автоматически подразумевает под собой колоссальных затрат не только со стороны финансирования, но и трудоёмких работ по проектированию, в особенности программного комплекса. В статье предлагается механизм упрощения разработки конкретного аппарата, путём анализа уже имеющихся прототипов и изменения самых высокочастотных элементов всего программно-аппаратного комплекса. Процесс представляет собой последовательное исследование и анализ сегментов аппарата, которые нуждаются в автоматизации и предложения упрощённого варианта этого автоматизирования. Результатом является разработка аппарата на её начальных этапах.

Ключевые слова: информационные технологии, вычислительная техника, разработка аппаратных средств, медицинская аппаратура, автоматизация процессов, инфузор.

Annotation: a significant part of innovative research and development in the field of medical equipment automatically implies that it will perform its duties within the program complex. The article suggests a mechanism for simplifying the development of devices, by analyzing the already connected prototypes and changes in the most high-cost elements of the entire software and hardware complex. The process is a sequential study and analysis of machine segments that need automation and suggestions for a simplified version of this automation. The result is the development of the apparatus at its initial stages.

Key words: information technology, computer technology, hardware development, medical equipment, process automation, infusion.

При разработке проекта была поставлена задача совершенствования инфузионных систем подачи лекарственных препаратов (капельниц), способом выведения их на минимальный уровень электронного контроля с помощью применения датчиков и модулей, контролируемых микропроцессором [1]. Исходя из более точной постановки задачи элементы системы должны обеспечивать: ввод пользователем объёма и скорости подачи препарата; одновременное отображение вводимых данных на дисплей; контроль датчика и передача данных с него на процессор, непосредственно во время процедуры; автоматическая остановка подачи препарата по окончании процедуры и передача сигнала об окончании пользователю. Кроме этого, возникла необходимость взять во внимание экономическую составляющую проекта. Это обусловлено текущим состоянием финансирования, поддержки и кадрового обеспечения сферы разработки. При этом стояла цель разработки аппарата без потерь в качестве и устойчивости.

В настоящий момент существует лишь одна отечественная разработка подобной системы, она пока не вышла на уровень производства и находится в стадии медицинских испытаний. При анализе данной разработки можно отметить что в сравнении с зарубежными прототипами – этот аппарат является более финансово выгодным для производства и реализации, но всё же остаётся функционально, аппаратно и программно более сложным, чем текущая разработка. Это обусловлено наличием множества датчиков, в частности определяющих номенклатуру препарата и его температуру. Целью же текущей работы является максимально упростить конструкцию и программную составляющую аппарата.

Кроме этого, ставится цель экономического и социального эффекта, за счёт достижения минимальной себестоимости конечного аппарата и возможностью применения в учреждениях региона.

При этом в приборе планируется предусмотреть функцию передачи соответствующих сигналов на пост медицинского сотрудника, по окончании процесса инфузии препарата.

Далее о структуре и принципе работы аппарата.

Структура системы приведена на рисунке 1.

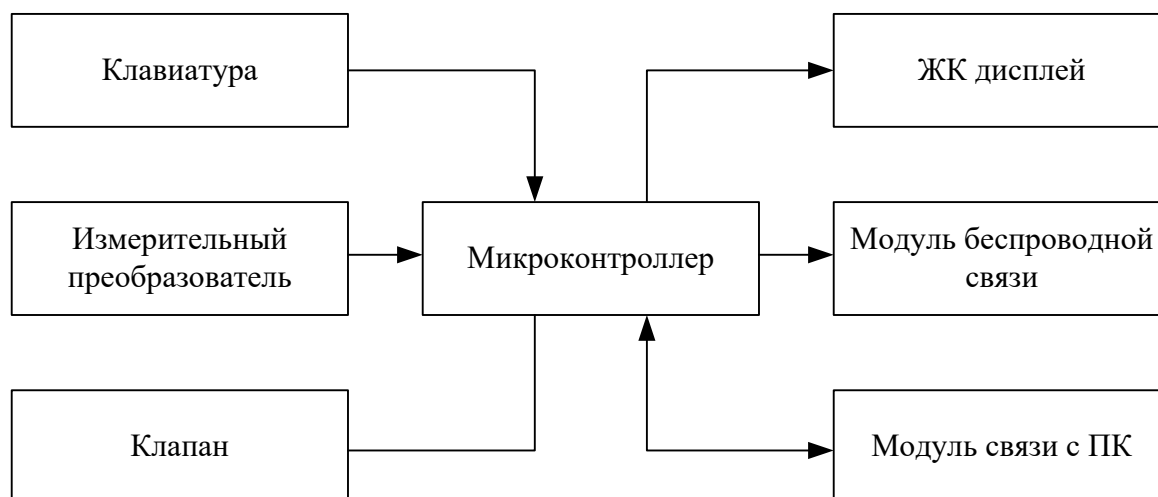


Рис.1. Структура системы контроля

В качестве микроконтроллера используется ATmega328, на базе которого построена аппаратная платформа Arduino Nano, являющаяся очень удобным средством макетирования системы на этапе проверки схемных решений и отладки программного обеспечения. Помимо основной роли управления устройством, считывания данных с датчиков и выводом текущей информации на дисплей, платформа так же выполняет функцию программатора для микроконтроллера. Выбор этой платформы для разработки обоснован многофункциональностью, но простотой среды программирования, открытостью исходных кодов и спецификаций. Кроме этого, платформа выполняет поставленную задачу – экономичности и минимизации себестоимости. В дальнейшем при оптимизации разработки будет применяться специализированная печатная плата [2].

Основной задачей системы является подсчет капель препарата, прошедших через капельницу. С этой целью в системе используется оптический измерительный преобразователь, помещенный в корпус из светонепроницаемого материала [3]. Корпус датчика размещается на капельнице.

После анализа и обработки данных от датчика, они выводятся на жидкокристаллический дисплей символьного типа. Который соединяется с контроллером путём подключения линий к цифровым выводам платформы. На нём так же отображаются данные, вводимые с клавиатуры. Оптимальным вариантом устройства ввода является матричная клавиатура, размером 4x4. Подключение клавиатуры к платформе осуществляется с помощью соединительных проводов-перемычек типа Male to Male [4] Выбор проводов в качестве соединителей обуславливается невысокой концентрацией соединений на плате. Запрограммировать работу клавиатуры можно с помощью специальной библиотеки. Схема подключения дисплея и клавиатуры представлена на рисунке 2.

Следующим шагом в работе устройства является информирование пользователя об окончании процедуры. Этот этап реализуется с помощью модуля беспроводной связи Bluetooth HC-06. Существует большое количество реализаций модулей Bluetooth. Каждая имеет свои особенности, но, в общем и целом, они все очень похожи. Выбор именно этого модуля обоснован его пониженным употреблением энергии, повышенным уровнем защиты данных и простым подключением к устройству. Подключение модуля к платформе осуществляется по принципу схожему с подключением клавиатуры. При этом изменяется тип проводов-перемычек на тип male-female[4]. На рисунке 3 представлена схема подключения модуля Bluetooth.

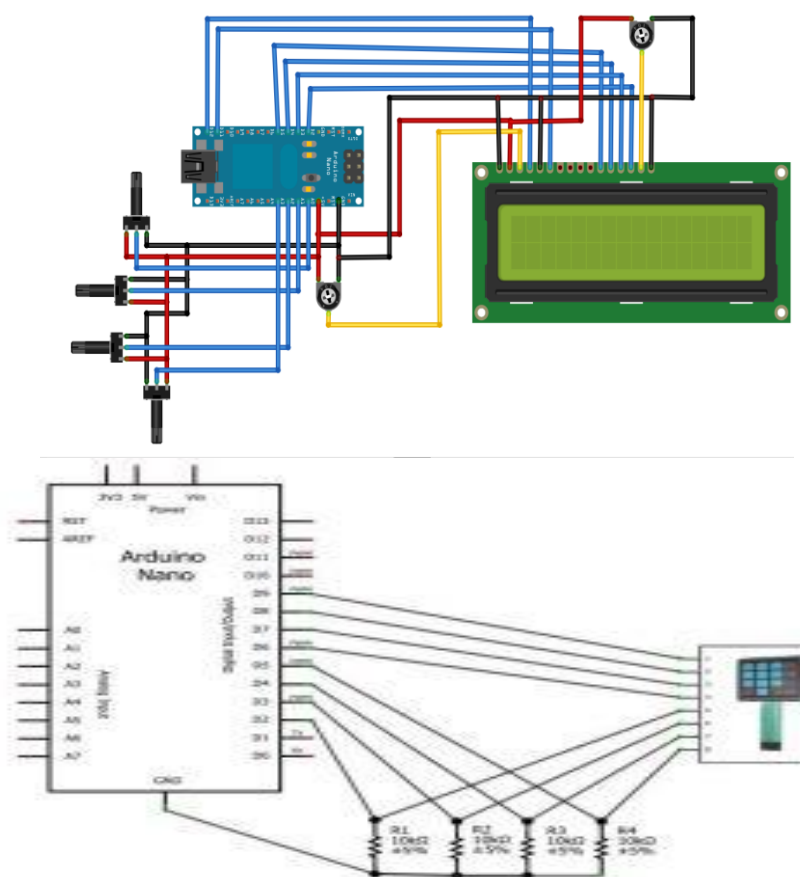


Рис.2. Подключение ЖК-дисплея и матричной клавиатуры к платформе Arduino

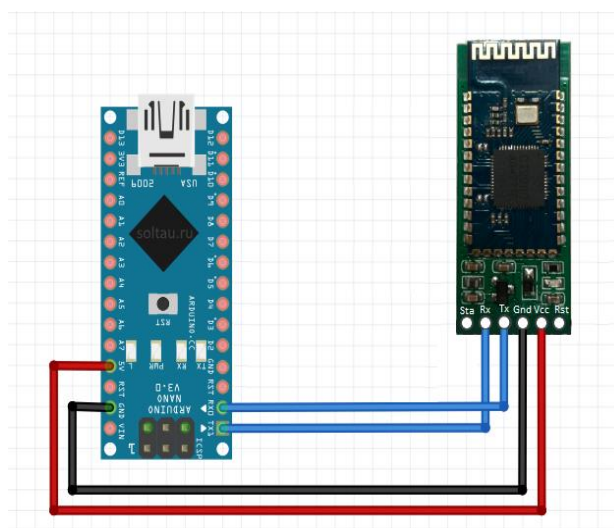


Рис.3. Подключение модуля беспроводной связи к платформе Arduino

В системе предусмотрена возможность использования клапана с электромеханическим управлением [4], который позволяет прекращать подачу медицинского препарата по достижении заданного объема с выдачей сигнала на пост дежурного.

Таким образом мы рассмотрели один из способов модернизации капельницы до уровня автоматизированного устройства с применением уменьшенного количества электронных элементов и минимизацией затрат. Ставя перед собой задачу минимизации затрат на разработку данного устройства, мы, кроме цели финансовой и трудовой экономии, предприняли попытку сделать разработки в такой сфере как автоматизация медицинских

процессов проще и доступнее. Делая попытку прогноза развития рассмотренного вопроса, можно сказать, что одной из задач было сделать автоматизацию простейших медицинских процессов доступной для Севастопольского региона, что может позитивно отразиться на общем состоянии такой важной для населения сферы как медицинское обслуживание.

Библиографический список

1. Гуменюк Н.И. Инфузионная терапия / Гуменюк Н.И., Киркилевский С.И – К.: Книга плюс, 2004. – 208с.
2. Евстифеев. А.В. Микроконтроллеры AVR семейства Mega.–М.: Додэка-XXI. 2010.–285с.
3. Удд Э. Волоконно-оптические датчики. – М.: Техносфера. 2008. – 520 с
4. Петин В.А. Arduino и Raspberry Pi в проектах Internet of Things. – СПб.: БХВ, 2016. – 320с.
5. Самойлов Е.А. Детали машин и основы конструирования/ Рощин Г.И., Самойлов Е.А. - М.: Дрофа, 2006. — 415 с.

УДК 615.47:616-71

А.С. Шамков, студент 4-го курса**В.С. Чернега, к.т.н., доцент***Севастопольский государственный университет***ОСОБЕННОСТИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ СИСТЕМЫ КОНТРОЛЯ МЕДИЦИНСКИХ КАПЕЛЬНИЦ**

Аннотация: В статье приведено описание разработки системы дистанционного контроля функционирования медицинских капельниц. Система состоит из оптического датчика капель, микроконтроллера, модуля приема-передачи данных по беспроводному каналу, центрального монитора и смартфона. В состав программной реализации входят функции передачи данных по беспроводному каналу, программирование таймеров микроконтроллера, функции обработки прерываний, вычисление частоты вводимой жидкости, ее остатка, вычисление количества времени до завершения процедуры. Рассмотрены различные варианты реализации системы, проблемы, возникшие при разработке и пути их решения.

Ключевые слова: Микроконтроллер, капельница, контроль, веб-интерфейс, веб-сервер, дистанционный доступ, WiFi-модуль, передача пакета.

Annotation: There is article to bring description development of the system remote control functioning medical dropper. System consists optical sensor of drops, microcontroller, wireless data link module, central monitor and smartphone. Software implementation consists of function data link, programming timers of microcontroller, function of attach interrupt, calculation of the frequency of the injected fluid and rest there, calculation of time to finish procedure. Here we consider different ways implementation of system, problems and decision.

Key words: Microcontroller, dropper, control, web interface, web server, remote access, WiFi-module, packet transmission.

В медицинских учреждениях при лечении больных широко используются капельницы, посредством которых лекарство по каплям с определенной частотой подается в организм пациента. При этом дежурной медсестре необходимо постоянно контролировать частоту капель подаваемого раствора. В отделении обычно одновременно задействовано несколько капельниц и медсестре требуется периодически проверять состояние капельницы. Это отвлекает ее от выполнения основных функциональных обязанностей.

С целью повышения эффективности работы медсестры спроектирована система удаленного контроля функционирования медицинских капельниц. Система непрерывно опрашивает датчик капель, измеряет их частоту, вычисляет общий объем введенной жидкости и ее остатка, время до окончания процедуры. Эти параметры выводятся на экран монитора медицинского поста. Предусмотрена возможность удаленного доступа к данной информации со смартфона или планшета по сети WiFi. При выходе частоты капель за допустимые пределы на медицинском посту осуществляется тревожная сигнализация. Кроме реальной частоты капель на экран монитора также выводится информация о фамилии больного и месте локализации капельницы. Эта же информация с центрального монитора передается по беспроводному каналу на смартфон или планшет постовой медсестры и дежурного врача. Допустимые пределы частоты капель, объем вливаемого раствора, место локализации капельницы вводятся медсестрой с консоли монитора перед установкой капельницы.

Спроектированная система состоит из датчика контроля капель поступающей жидкости, микроконтроллера, WiFi-модуля приема-передачи данных по беспроводному каналу, точки доступа, осуществляющей ретрансляцию данных, центрального монитора поста медсестры и, непосредственно, смартфона или планшета, который постоянно находится у медсестры.

Датчик контроля капель поступающей жидкости построен на основе инфракрасного излучающего диода и фототранзистора, работающие на длине волны 940 нм. Применение оптических устройств инфракрасного диапазона позволяет использовать датчик в условиях дневного и искусственного освещения. Датчик подключен к входу вывода внешнего прерывания микроконтроллера. В момент пролета капли на выходе этого датчика формируется импульс, который вызывает прерывание у микроконтроллера. Это позволяет контроллеру выполнять фоновую программу без постоянной проверки состояния датчика, а в случае возникновения прерывания прекратить выполнение фоновой программы, и переключиться на обработку этого действия.

Для управления системой контроля медицинских капельниц разработана программа-монитор, осуществляющая инициализацию системы, опрос датчика капель, установление соединения с точкой доступа и центральным монитором, передачу измеряемых данных на медицинский пост и смартфон, графический интерфейс центрального монитора. Для программирования использовался язык C.

Процесс инициализации системы состоит в задании связи IP-адреса устройства, указанного на корпусе датчика, с номером помещения (палаты), в котором устанавливается капельница, фамилии больного, вид раствора и его объем, фамилия дежурной сестры и врача.

Основной проблемой при опросе состояния датчика капель явилась проблема различения импульса капли от импульса помехи. При решении этой проблемы была использована временная селекция, основанная на том, что длительность импульса от капли составляет $(2...5) \pm 1$ мс, а длительность импульсная помехи не превышает 1 мс. Поэтому в функции обработки прерывания от импульса датчика капель предусмотрено измерение длительности импульса. Если длительность находится в заданных пределах, то осуществляется увеличение счетчика капли на единицу и измерение времени, прошедшего с момента пролета последней капли для вычисления частоты капель. В случае если длительность импульса меньше 1 мс, то он игнорируется. Это позволило повысить помехоустойчивость системы.

В качестве микроконтроллера использовалась модуль Arduino Uno, на базе процессора ATmega328P. Для программирования были выбраны среда Arduino и язык программирования C. Вызов обработчика прерывания реализован с помощью встроенной в язык Arduino функции `attachInterrupt (interrupt, function, mode)`, запускаемой в момент изменения состояния цепи прерывания. Аргумент `interrupt` – в данном случае это номер прерывания. В связи с тем, что использовалась плата Arduino Uno, то номер аргумента указан 0, это соответствуют цифровому порту 2, который также является входом внешних прерываний микроконтроллера. Следующий аргумент `function` – это имя функции, являющейся обработчиком прерывания. Последний параметр `mode` – это режим обработки прерывания, в нашем случае этот аргумент принимает значение «CHANGE». Это означает, что прерывание будет вызываться при смене состояния входа прерывания. Внутри функции обработчика осуществляется проверка этого состояния. Если на электроде имеется потенциал логического нуля, это означает начало срабатывания датчика, поэтому запускается таймер отсчета, посредством которого осуществляется вычисление длительности импульса. При повторном срабатывании этой функции осуществляется считывание времени таймера и проверка длительности импульса относительно заданных пределов. Если длительность выходит за установленные границы, то импульс игнорируется. В противном случае осуществляется увеличение количества капель на единицу. При первом срабатывании данной функции осуществляется запоминание текущего времени на основном таймере контроллера при помощи функции `millis()`. При последующих срабатываниях, путем вычитания текущего времени на таймере и предыдущего, вычисляется интервал между импульсами, на основании которого рассчитывается частота падения капель.

Микроконтроллер использует еще один таймер, который отсчитывает 10 с и вызывает прерывание для передачи данных. Для программирования этого таймера использовалась библиотеки «`timer-api.h`» и «`timer_setup.h`» [1]. В разделе установки микроконтроллера (функция `void setup()`) указывается функция обработчик прерывания по таймеру. Для обращения к нужному таймеру применялась глобальная переменная `TIMER_DEFAULT`, которая принимает значение свободного таймера, в нашем случае второго. Частота срабатывания функции прерывания от таймера устанавливается 10 МГц, что соответствует интервалу 10 с. Связь микроконтроллера с радиомодулем осуществляется при помощи универсального асинхронного приёмопередатчика (UART). К порту RX модуля подключается порт TX контроллера, и через каждые 10 с осуществляется передача данных. В случае критической ситуации (прекращение поступления жидкости, уменьшение частоты вливания ниже заданной границы) осуществляется мгновенная передача данного пакета модулю. В программной реализации этого процесса на стороне микроконтроллера реализован стандартный потоковый вывод при помощи функции `Serial.println()`. На приемной

стороне (модуль приемо-передачи) осуществляется постоянный опрос порта при помощи функции `Serial.available()`. Если данная функция возвращает значение, отличное от нуля, то это означает наличие байта в регистре, которое считывается при помощи функции `Serial.read()`. Использование UART позволяет автоматически контролировать целостность данных методом контроля битовой четности. Это гарантирует безошибочную передачу данных, что является немаловажным аспектом в работе системы.

В качестве устройства передачи по беспроводному каналу выбран модуль ESP8266-12e, который имеет встроенную WiFi-антенну. Этот модуль решает следующие задачи: подключение к известной точке доступа, прием пакета данных от микроконтроллера, передачу данного пакета точка доступа. Настройка подключения устройства к беспроводной сети осуществляется при первом вводе в эксплуатацию. Модуль хранит в своей постоянной памяти данные о последнем успешном подключении к точке доступа (логин и пароль). В случае если в течение 10 с модуль не смог подключиться к известной ему точке доступа он переключается в режим «точки доступа», для осуществления настройки. Программирование модуля осуществляется также на языке C, в среде Arduino. Поэтому использовались библиотека «ESP8266WiFi.h» – для организации передачи данных по беспроводной сети и библиотека, «ESP8266WebServer.h» – для организации собственного сервера на базе данного модуля [2]. Это необходимо для реализации интерфейса с пользователем, когда модуль находится в режиме «точки доступа». Библиотека «WifiManager.h» облегчает подключение к другим точкам доступа или переход в режим «точки доступа», а «EEPROM.h» использована для работы с постоянной памятью модуля.

Первоначально планировалось, что данный модуль будет осуществлять функции текущего микроконтроллера, но возникли некоторые трудности. В частности, этот модуль не содержит портов вывода внешних прерываний. Поэтому необходимо постоянно опрашивать порт на наличие соответствующего потенциала, что занимает большую часть работы модуля, и делает невозможным выполнение других функций, например передачи пакета. Другая проблема заключалась в том, что в момент передачи данных модуль прерывает работу таймера, что сказывается на точности измерений частоты. Вследствие этого было принято решение о разграничении выполняемых функций между микроконтроллером и модулем приемо-передачи данных.

Также планировалось, что на текущем модуле приемо-передачи будет содержаться собственный веб-сервер, в котором будет отображаться текущая информация о частоте введения жидкости, объеме введенной жидкости и ее остатке, а также о времени до окончания процедуры. Таким образом, медсестра подключалась бы к основному серверу, реализованному на посту, который предоставлял бы доступ к веб-серверу модуля, для мониторинга текущего состояния. Это сокращает количество передач данных между модулем и основным сервером. Однако у этого способа реализации есть и минус: в случае потери связи между сервером и модулем не будет никакой индикации, что недопустимо в данном случае.

Вторая реализация, которая и была принята, заключается в том, что модуль постоянно отправляет данные на основной сервер. У этого способа есть недостаток, заключающейся в том, что возможно возникновении коллизии, когда несколько модулей одновременно будут передавать данные. Для разрешения данной проблемы используется метод со случайным доступом. Модуль отсчитывает случайное количество времени и отправляет данные на сервер. Причем отправка осуществляется три раза. Сервер, получая три пакета данных от модуля и производит их сравнение. В случае несовпадения данных одного из пакетов с двумя другими, осуществляется замена несовпадающих компонент. Таким образом, в случае возникновения ошибок при передаче, происходит их автоматическое исправление.

Библиографический список

1. Arduino-timer-api [Electronic resource]. – URL: <https://github.com/sadr0b0t/arduino-timer-api>.
2. Arduino documentation [Electronic resource]. – URL: <https://www.arduino.cc/reference/en>.

СЕКЦИЯ 6
Компьютерные сети



УДК 004.05

В.В. Евсеева, магистрант 2-го курса**Е.Н. Машенко, канд. техн. наук, доцент***Севастопольский государственный университет***ИССЛЕДОВАНИЕ КРИТЕРИЕВ ЭФФЕКТИВНОСТИ ФУНКЦИОНИРОВАНИЯ БЕСПРОВОДНЫХ СЕНСОРНЫХ СЕТЕЙ**

Аннотация: В данной статье обсуждаются основные параметры, характеризующие беспроводные сенсорные сети. Анализируются основные критерии эффективности БСС. Предложена методика комплексной оценки работы сенсорной сети.

Ключевые слова: Беспроводные сенсорные сети, критерии эффективности сенсорных сетей, комплексная оценка эффективности, БСС, QoS.

Annotation: This article discusses the basic parameters that characterize wireless sensor networks. The main efficiency strategies of wireless sensor networks are analyzed. The technique of complex evaluation of the sensor network operation is proposed.

Key words: wireless sensor networks, criteria of effectiveness of WSN, a comprehensive assessment of effectiveness, WSN, QoS.

Потребность в оперативном получении информации о состоянии объектов и развитие микроэлектроники и технологий беспроводной связи привели к возникновению особого класса систем – беспроводных сенсорных сетей. Беспроводная сенсорная сеть (БСС) – это распределенная, самоорганизующаяся сеть множества сенсоров и исполнительных устройств, объединённых между собой посредством радиоканала [1].

Сенсор (англ., sensor) – устройство, которое воспринимает контролируемое воздействие (свет, давление, температуру и т. п.), измеряет его количественные и качественные характеристики и преобразует данные измерения в сигнал [2]. Сенсоры включают в себя датчики электрического и магнитного поля; датчики частоты радиоволн, инфракрасные датчики, лазеры, датчики местоположения (навигации), сейсмические датчики, экологические, биохимические датчики и т.д.

Датчик (англ., transducer) – устройство, которое используется для преобразования одного вида энергии в другой [2]. Датчики в БСС характеризуются различными целями, функциями и возможностями, благодаря чему, под давлением последних технологических достижений, БСС нашли применение во множестве различных сфер деятельности, таких как радарные сети, используемые в контроле воздушного сообщения, метеорологические станции, национальные электроэнергетические сети, военная сфера и т.д.

Важную роль в функционировании БСС играет архитектура сети. Архитектура БСС представлена двумя классами: с одним главным узлом (single-hop) и с несколькими главными узлами (multi-hop) [3], рисунок 1.



Рис. 1. Классификация методов построения БСС

На сегодняшний день невозможно оценивать эффективность беспроводной сети с точки зрения лишь одного критерия. Эффективность – это комплексное свойство системы выполнять поставленную цель в заданных условиях и с определенным качеством. При этом степень приспособленности системы к выполнению поставленных задач характеризуется множеством критериев эффективности, т.е. количественными характеристиками сети, определяемыми процессом функционирования сети.

Для оценки эффективности функционирования БСС необходимо определить назначение сети для выбора основных параметров, характеризующих сенсорную сеть (таблица 1), а на основе модели сети и ее параметров можно произвести комплексную многокритериальную оценку (рисунок 2).

Таблица 1. Параметры, характеризующие БСС

№ п/п	Критерий	Определение	Обозначение
1	Общее число измеряемых параметров	Параметры, измеряемые датчиками (температура, влажность, давление и т.д.)	ЧИП
2	Количество узлов в сети	-	КУ
3	Топология сети	Архитектура сети (количество главных узлов, положение узлов и т.д.)	ТС
4	Плотность размещения узлов	Максимальное, минимальное и среднее число «соседей» узла сети	ПРУ
5	Диаметр сети	Минимальное количество ретрансляций в маршруте передачи данных между двумя самыми удаленными узлами сети	ДС
6	Объем памяти сенсорного узла	-	ОПС
7	Скорость передачи данных по каналу	-	СПДК
8	Площадь мониторинга	-	ПМ
9	Интенсивность опроса узлов	Количество опросов узлов сети в единицу времени	ИО
10	Стоимость	Суммарная стоимость оборудования сети	С
11	Латентность	Задержка от момента возникновения события до доставки информации о нем до базовой станции	Л
12	Время жизни сети	Время работы БСС без замены или подзарядки аккумуляторов	ВЖ
13	Пропускная способность	Объем информации, проходящий через сеть в единицу времени	ПС

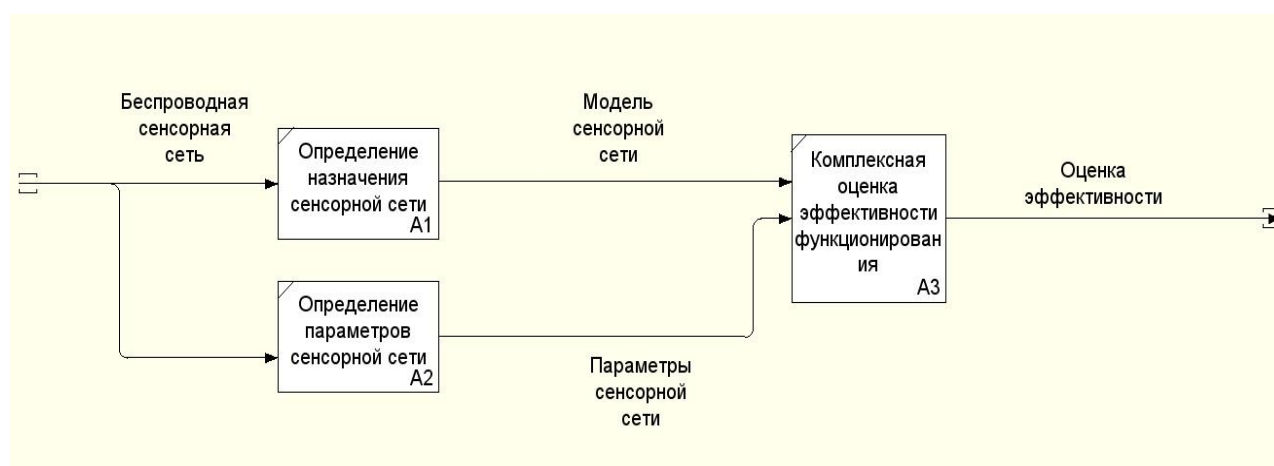


Рис. 2. Методика оценивания эффективности БСС

Эффективное функционирование БСС в условиях внешнего воздействия, помех, возможной мобильности и отказов узлов не может осуществляться без эффективной системы управления сетью. Процесс функционирования системы управления БСС в отличие от традиционных сетей включает в себя функционирование трех подсистем [1]:

1. Подсистема мониторинга. Основная задача данной подсистемы – обеспечить покрытие зоны и ее мониторинг по заданному множеству параметров. Подсистема должна обеспечить заданное качество покрытия и мониторинга.

2. Подсистема телекоммуникаций. Обеспечивает передачу данных от узлов, осуществляющих мониторинг, к базовой станции с заданным качеством.

3. Подсистема энергосбережения. Т.к. узлы сети имеют автономные источники питания, то одной из основных целей управления сетью при обеспечении заданного качества процессов покрытия, мониторинга и передачи данных является увеличение времени жизни сети на основе реализации различных методов управления, нацеленных на минимизацию расхода энергоресурса каждого узла и сети в целом.

На этапах проектирования, развертывания и оперативного управления БСС возникает задача оценки параметров ее функционирования. В силу того, что различные приложения выдвигают свои специфические требования, особенности построения и функционирования БСС определяют дополнительные критерии оценки эффективности.

Критерии энергоэффективности. Так как сенсорные узлы работают от автономных источников питания, как правило, от аккумулятора с невозполняемой энергией, то количество доступной энергии чрезвычайно ограничено и, следовательно, наступает момент, когда сеть более не может решать возложенные на нее задачи. Время от начала работы сети до данного момента называется временем жизни – $T_{ж}$.

Различают время жизни узла БСС и время жизни сети в целом.

Факторами, влияющими на время жизни узла, являются:

- Характеристики аппаратных средств;
- Частота сбора и передачи данных, зависящая от приложения;
- Топология сети, определяющая объем информации, проходящий через каждый элемент [3].

Время жизни узла определяется как отношение начальной энергии батареи E_{i0} i -го узла к средней потребляемой им мощности P_i :

$$T_{жи} = \frac{E_{i0}}{P_i} \quad (1)$$

Формула средней потребляемой узлом мощности имеет вид:

$$P = (P_{мон}t_{мон} + P_{пер}t_{пер} + P_{прм}t_{прм} + P_{ож}t_{ож} + P_{сон}t_{сон})/t_{ц}, \quad (2)$$

где $t_{ц} = t_{мон} + t_{пер} + t_{прм} + t_{ож} + t_{сон}$ – продолжительность одного цикла работы узла (считывание данных мониторинга, передача и прием данных, ожидание, сон);

$P_{пер}t_{пер}$ – средняя мощность и время, затрачиваемых в процессе передачи данных;

$P_{прм}t_{прм}$ – средняя мощность и время приема данных;

$P_{мон}t_{мон}$ – суммарное время, затрачиваемое на считывание показаний с датчиков, их обработку и подготовку к передаче и потребляемая мощность в режиме мониторинга;

$P_{сон}t_{сон}$ – потребляемая мощность в режиме сна и время сна;

$P_{ож}t_{ож}$ – время и затрачиваемая энергия в режиме ожидания.

Критерии, основанные на количестве работающих узлов. Время жизни сети $T_{ж}$ может определяться минимальным временем жизни n критических узлов сети $T_{ж}(n)$, которые все время должны быть в работоспособном состоянии:

$$T_{жи} = \min_{i \in N} t_{жи} \quad (3)$$

Критерии, основанные на времени выполнения задач на различных этапах функционирования сети. Временные интервалы на реализацию данных этапов должны

быть не больше заданных, т.е. время планирования $t_{пл} \leq t_{плзад}$, время развертывания $t_{раз} \leq t_{раззад}$, время восстановления (наращивания) $t_{вос} \leq t_{восзад}$ сети должны быть не более заданных (директивных). Однако, существует противоречие: чем меньше собираем служебной информации о состоянии сети, тем менее качественно принимаемое решение.

Критерии, основанные на требованиях приложений. Качество обслуживания (Quality of service – QoS) вводится в сетях для определения способности сети соответствовать заданным требованиям и характеристикам. Качество обслуживания определяет, может ли предоставить необходимый сервис по передаче данных при заданных условиях. В отличие от классических сетей БСС предоставляют сервис не только по передаче данных, но также по сбору и обработке. Соответственно качество обслуживания в беспроводных сенсорных сетях и его показатели также будут отличаться от классического представления.

Показателями качества функционирования подсистемы мониторинга $K(MON)$ могут быть: время развертывания, степень покрытия заданной зоны, время ее мониторинга, время жизни, вероятность обслуживания событий, точность наблюдения (ошибки измерения), необходимое количество параметров мониторинга, интенсивность и тип трафика мониторинга.

Для подсистемы телекоммуникаций основными параметрами являются $K(TKM)$: пропускная способность, задержка передачи, вероятность ошибок и т.п. Т.е. сеть должна обеспечить пропускную способность S_c не ниже заданной $S_c > S_{зад}$ в пределах заданного промежутка времени. В свою очередь пропускная способность сети S_c определяется как сумма пропускных способностей направлений.

Критерий минимума максимальной задержки передачи:

$$\max_{t \in T} \{t_{zi-BS}\} \rightarrow \min, \quad (4)$$

где T – время работы системы;

t_{zi-BS} – время задержки в доставке данных от узла i до базовой станции BS в момент времени t .

Данный критерий применяется для систем реального времени, которые требуют необходимого уровня качества связи для каждого узла.

Критерий суммарной средней задержки:

$$\sum_{i=1}^N M\{t_{zi-BS}\} \rightarrow \min, \quad (5)$$

где N – количество узлов сети;

$M\{t_{zi-BS}\}$ – математическое ожидание времени доставки данных от узла i до базовой станции BS .

Данный критерий применяется для систем, где единичная задержка не критична, однако в целом, необходима определенная производительность.

Для оценки эффективности функционирования беспроводной сенсорной сети будем использовать вектор локальных критериев $K = (k_1, ..., k_i, ..., k_n)$, где k_i – значение i -го локального критерия. Значение комплексного критерия $f(K)$, характеризующее эффективность исследуемой системы, определяется в соответствии с принципами, обозначенными в таблице 2, где a_i определяет степень относительной важности i -го критерия.

С учетом требований к надежности сети выбор критерия комплексной оценки эффективности функционирования ее структурных компонент обусловлен необходимостью получения гарантированных оценок. Поэтому в качестве комплексного критерия будем использовать критерий $f_1(K)$, который представляет собой пошаговую процедуру агрегирования оценок и позволяет получить объективное соотношение «значение критерия» – «вес критерия» без нормализации величин, входящих в состав критерия.

Оценка эффективности сети осуществляется на основе парных сравнений по значениям показателей (таблица 1).

Таблица 2. Способы построения комплексных критериев

№ п/п	Название принципа агрегирования	Аналитическое выражение
1	Аддитивная свертка	$f_1(K) = \sum k_i a_i$
2	Мультипликативная свертка	$f_2(K) = \prod (k_i)^{a_i}$
3	Равномерная оценка локальных критериев	$f_3(K) = \min((k_i)^{a_i})$
4	Принцип пессимизма-оптимизма	$f_4(K) = a_i \min(k_i) - (1 + a_i) \max(k_i)$
5	Оценка по наиболее важному критерию	$f_5(K) = \max((k_i)^{a_i})$

$$A^k = \{a_{ij}\} = \begin{pmatrix} 1 & \cdots & \alpha_{1m} \\ \vdots & \ddots & \vdots \\ \alpha_{m1} & \cdots & 1 \end{pmatrix}, \quad (6)$$

где A^k – матрица сравнений альтернатив по k -му критерию; $a_{ij} = a_i/a_j$ – оценки преимущества i -й альтернативы над j -й; a_i, a_j – значения k -го критерия для i -й и j -й альтернатив.

Вектор приоритетов альтернатив $p^k(p_1, p_2, \dots, p_m)$ вычисляется в соответствии с принципом среднего геометрического взвешивания:

$$p_i^k = \frac{\sqrt[m]{a_{i1} * \dots * a_{im}}}{\sum_{i=1}^m \sqrt[m]{a_{i1} * \dots * a_{im}}}, \quad (7)$$

где p_i^k – оценка значимости i -й альтернативы по k -му критерию; a_{ij} – элемент матрицы парных сравнений (МПС).

Итоговые оценки эффективности вычисляются с использованием аддитивного принципа свертки S :

$$S_i = \sum_{k=1}^n p_i^k * a_k, \quad (8)$$

где S_i – оценка эффективности i -го альтернативного варианта реализации сети.

После подсчета количественной характеристики эффективности беспроводной сенсорной сети необходимо произвести оценку полученных результатов. Также для дальнейшей работы необходимо составить базу правил, на основе которой, можно получить качественную характеристику сети.

Таким образом, в данной статье были рассмотрены вопрос анализа критериев эффективности беспроводных сенсорных сетей. Определены параметры сенсорных сетей, влияющих на производительность сетей. Определены множества критериев эффективности для каждой из функциональных подсистем БСС. Установлено, что задача определения эффективности функционирования является многокритериальной задачей, в которой важно предварительное ранжирование показателей. Для дальнейших исследований будет произведена комплексная оценка, а именно рассчитаны показатели эффективности функционирования, для созданных трех типов моделей сенсорных сетей с тремя различными подтипами параметров.

Библиографический список

1. Жук О.В. Система управління тактичними сенсорними мережами / О.В. Жук, В.А. Романюк, О.Я. Сова // Збірник наукових праць ВІТІ НТУУ „КПІ”. – 2008. – № 2. – С. 88 – 96.
2. Интернет вещей: учебное пособие / А.В. Росляков, С.В. Ваняшин, А.Ю. Гребешков. – Самара: ПГУТИ, 2015. – С. 46–60.
3. Мозговая В.В., Мащенко Е.Н. Анализ методов максимизации длительности жизненного цикла беспроводных сенсорных сетей // Материалы Всероссийской молодежной научной школы. Воронеж: ФГБОУ ВО «Воронежский государственный технический университет», 2017. Ч.1. С. 58-62.

УДК 004:65

Е.Ю. Соколова, студентка 4-го курса**М.В. Потанина, кандидат технических наук, доцент***Севастопольский государственный университет***РОЛЬ КОМПЬЮТЕРНЫХ СЕТЕЙ И ИНТЕРНЕТА В РАЗВИТИИ БИЗНЕСА***Аннотация: в данной статье рассматривается роль компьютерных сетей и Интернета в развитии современных компаний.**Ключевые слова: компьютерные сети, Интернет, бизнес.**Annotation: this article discusses the role of computer networks and the Internet in the development of modern companies.**Key words: computer networks, Internet, business.*

Сегодня в мире происходит активное развитие новых информационных технологий и постоянная модернизация уже существующих. Современное общество пережило множество изменений, которые связаны с общением и получением информации.

Еще совсем недавно настоящий фурор по всему миру производили факсы, системы голосовой почты, пейджеры, видеодиски – все эти носители и средства передачи информации сейчас практически забыты. Им на смену пришли спутники, кабельное телевидение, мобильные телефоны и, самое главное, Всемирная сеть Интернет. Без этих атрибутов жизнь современного человека кажется практически невозможной.

Любой процесс управления невозможно представить без оперативного обмена информацией различного рода. Для этого необходимо владеть телекоммуникационной сетью. В мире существует пять видов телекоммуникационных сетей: телеграфная, телефонная, радиосвязь, спутниковая связь и компьютерная сеть.

Самой первой и самой старой связью является телеграфная, она появилась еще в 19 веке. Ею можно пользоваться и сегодня, но данную связь вытесняют более современные виды связи. Так, изобретение в 1876 году телефона, положило начало развитию телефонных сетей. Радиосвязь практически не применяется в деятельности различных компаний. Она является составной частью вычислительных сетей, в частности сети Интернет. Спутниковые системы связи получили широкое распространение по всему миру. Они необходимы для передачи данных в других системах связи, для организации телевизионного вещания. Самым актуальным и современным средством телекоммуникации выступает компьютерная вычислительная сеть [1].

Многие компании имеют большое количество филиалов и подразделений не только по всей России, но и в мире, которые находятся на большом расстоянии друг от друга. Для полноценного функционирования им необходимо ежедневно обмениваться большим количеством данных, в этой ситуации помогает наличие компьютерной сети с доступом в Интернет.

Объединение всех компьютеров предприятия в единую сеть позволяет осуществлять общий доступ всех сотрудников к конкретным данным или оборудованию. Также создание общей сети позволяет экономить память диска компьютера, ведь всю необходимую информацию можно занести на одно устройство, а остальным предоставить доступ. Подобным образом предприятие может сэкономить на оборудовании для сотрудников, приобретая один мощный дорогостоящий компьютер и организовав на его основании сеть для остальных сотрудников предприятия. Создание компьютерных сетей создает более гибкую рабочую среду, предоставляя сотрудникам предприятия возможность работать удаленно и в любое время [2].

Интернет – это глобальная сеть, которая предоставляет пользователям всего мира доступ к хранилищам данных, содержащих информацию, звук и изображения. Данная сеть охватывает все большее количество стран, регионов, городов, людей по всему миру, поэтому является важнейшей составляющей эффективного управления бизнесом.

Сеть Интернет с каждым годом становится все мощнее и больше, к ней подключаются новые индивидуальные пользователи, а также целые компании. Согласно данным представленным в таблице 1, можно утверждать, что количество пользователей сетью Интернет становится все больше и данная паутина охватывает разные сферы жизни людей.

Таблица 1 – Статистика пользователей интернета в мире по годам

Год	Число пользователей,	Население земли, %
1995	16 млн.	0,39
1997	76 млн.	1,81
2000	368,54 млн.	6,07
2001	513,41 млн.	8,46
2007	1,15 млрд.	17,2
2012	2,27 млрд.	31,9
2015	3,2 млрд.	43,4
2016	3,7 млрд.	50,2
2017	4,02 млрд.	54,5

Ученые прогнозируют, что к 2020 году количество пользователей возрастет до 4,1 млрд. человек, а это составит приблизительно 60% населения планеты. [3]

Исходя из вышеприведенной статистики можно предположить, что уже к концу 2018 года количество пользователей сети Интернет побьет показатели запланированные учеными к 2020 году.

Если обратиться к статистике пользователей сети Интернет в России, также можно увидеть значительный рост данного показателя. Российский филиал исследовательского концерна GfK (Gesellschaft für Konsumforschung) Group, 16.01.2018, опубликовал отчет "Проникновение Интернета в России: итоги 2017 года", который представлен на рисунке 1.

ПРОНИКНОВЕНИЕ ИНТЕРНЕТА В РОССИИ



Рис. 1. Проникновение Интернета в России

Данный показатель проникновения Интернета среди молодого населения достиг высоких значений еще пару лет назад. Сейчас рост происходит за счет увеличения доли пользователей в возрастной категории 55+ [4].

Исходя из вышеприведенных доводов, можно утверждать, что в современном мире Интернет является главным средством общения и поиска информации среди большей части населения и этот факт необходимо учитывать при управлении компанией.

Важнейшей составляющей успешной деятельности современной компании является наличие сайта предприятия в сети Интернет. Большая часть людей, прежде, чем пойти, например, в магазин, поищут информацию о нем в интернете, прочитают отзывы или свяжутся с магазином, не выходя из дома благодаря системе обратной связи, которая очень часто присутствует на сайтах предприятий. Поэтому крупные и мелкие, коммерческие и некоммерческие организации – все не обходят вниманием данную телекоммуникационную сеть.

В связи с этим возросла потребность в качественных конструкторах сайтов. В зависимости от масштабов предприятия, можно воспользоваться услугами команды профессионалов, которые создадут сайт «под ключ» в зависимости от требований компании.

Также, в связи с колоссальным развитием сети Интернет, для традиционного бизнеса появилась возможность расширить пределы функционирования и выйти на рынок электронного бизнеса. При чем размеры компаний особого значения не имеют – это могут быть и крупные корпорации, и небольшие фирмы. Большим преимуществом электронного бизнеса являются небольшие затраты на его ведение.

При переходе предприятия в сферу электронного бизнеса следует учитывать ряд факторов. Данный сектор все-таки более подходит для крупных предприятий, работающих в секторе «B2B». Однако, не стоит бояться экспериментов. Существует множество примеров компаний, которые однажды рискнули, попробовав расширить свою деятельность в данный сектор, и не прогадали. К примеру, компания Laird Superfood, перейдя на оптовые поставки через интернет-канал, увеличила продажи на 550%

Объем электронной коммерции растет, но не так масштабно, как ожидалось. Оказалось, что большинство людей предпочитают традиционную торговлю электронной. Например, в США доля электронной торговли составляет всего 9,1% (по данным на конец 2017 года), а в России эта доля составляет примерно 3-4%.

Основным трендом сегодняшнего дня в электронной коммерции выступает автоматизация всех процессов. Только крупные компании могут позволить себе экспериментировать с нововведениями и добиваться наилучших результатов в автоматизации бизнес-процессов, при чем, свои достижения они сохраняют в строжайшем секрете. Для компаний среднего уровня невозможно найти единый программный продукт, который позволит автоматизировать подразделения внутри самой компании и автоматизировать взаимодействие с контрагентами. [5]

В заключение можно сказать, что в последнее десятилетие наблюдается значительный рост и развитие телекоммуникационной компьютерной сети, а также сети Интернет, что открывает большие возможности для управления и развития бизнеса. Темпы роста пользователей сети Интернет превысили все прогнозы.

Сегодня Интернет как средство коммуникации, источник справочной информации, средство рекламы и маркетинга для ведения бизнеса (хозяйственной деятельности) вне электронных сетей и Интернет как инструмент ведения электронного бизнеса, основанного на принципах сетевой экономики.

Интернет играет огромную роль в бизнесе как средство коммуникаций (E-mail, телеконференции, IP-телефония так далее). Интернет не просто средство коммуникации, а интерактивный канал взаимодействия предпринимателей, позволяющий им общаться в режиме онлайн. Огромную роль играет Интернет как источник справочной информации для компаний. Вся информация о товарах и услугах, предоставляемых компаниями в Интернете

представлена в виде сайтов. Интернет является средством массовой информации, на серверах сети хранится огромное количество информационных ресурсов, поиск которых обеспечивают информационно-поисковые системы.

В свою очередь компьютерные сети позволяют упростить работу предприятий и в современном мире являются не такой уж и экзотикой, а скорее всего необходимостью. Компьютерные сети обеспечивают всех работников легким и быстрым доступом к информации, а также ограничивают доступ к различным интернет-ресурсам, которые отвлекают от работы.

Библиографический список

1. Виды телекоммуникаций. Лекция по дисциплине «Информационные и коммуникационные технологии в государственном и муниципальном управлении» // Елизавета Стройновская / 2017 [Электронный ресурс]: Режим доступа: <http://docplayer.ru/29050192-Vidy-telekommunikaciy.html>
2. Автоматизированные информационные технологии в экономике: Учебник/ Под ред. проф. Г.А. Титоренко. - М.: ЮНИТИ, 2003. стр. 22-32
3. Статистика интернета // 28.08.2017 [Электронный ресурс]: Режим доступа: <http://vawilon.ru/statistika-interneta/?user=5>
4. Количество пользователей интернета в России // 19.03.2018 [Электронный ресурс]: Режим доступа: http://www.bizhit.ru/index/users_count/0-151
5. Какой станет электронная коммерция в 2018-2019 годах? // Игорь Бахарев / 06.03.2018 [Электронный ресурс]: Режим доступа: <https://e-pepper.ru/news/kakoy-stanet-elektronnaya-kommertsiya-v-2018-2019-godak.html>

СЕКЦИЯ 7

Web – разработка



УДК 004.75

Н.А. Дерезглазов, студент 4-го курса**М.В. Заморёнов, канд. техн. наук, доцент***Севастопольский государственный университет***ПРОГРАММНАЯ СИСТЕМА, РЕАЛИЗОВАННАЯ В ВИДЕ
ВЕБ-ПРИЛОЖЕНИЯ НА МИКРОСЕРВИСНОЙ АРХИТЕКТУРЕ**

Аннотация: данная статья рассматривает сервис-ориентированный архитектурный подход при реализации распределенной программной системы. Описываются преимущества и недостатки данного подхода, а также способы проектирования интерфейса для передачи сериализованных данных между микросервисами. Рассмотрена программная реализация, аргументирована эффективность архитектурного подхода для организации распределённой вычислительной системы с децентрализованным управлением и хранением данных.

Ключевые слова: распределенные системы, микросервисная архитектура, балансировка нагрузки, gRPC, Protocol Buffers, protobuf.

Annotation: this article is about the service-oriented architecture for distributed software systems. The advantages and disadvantages of chosen architecture approach and interface design methods for serialized data transfer between micro-services are analyzed. The software realization of chosen architecture and its efficiency for building distributed software system with decentralized management and data storing is discussed.

Keywords: distributed computing systems, micro service architecture, load balancing, gRPC, Protocol Buffers, protobuf.

Важнейшим этапом проектирования программной системы является выбор архитектурного решения. Архитектура системы непосредственно влияет на её эффективность, гибкость, масштабируемость, устойчивость к нагрузкам и скорость развития программной системы в целом.

Классическим подходом является развитие системы с монолитной структурой, состоящей из нескольких базовых слоев (рисунок 1). Данный архитектурный подход является наиболее удачным для приложений, обладающих небольшой структурой и не подразумевающие дальнейшее добавление новых сервисов и громоздкой программной логики.

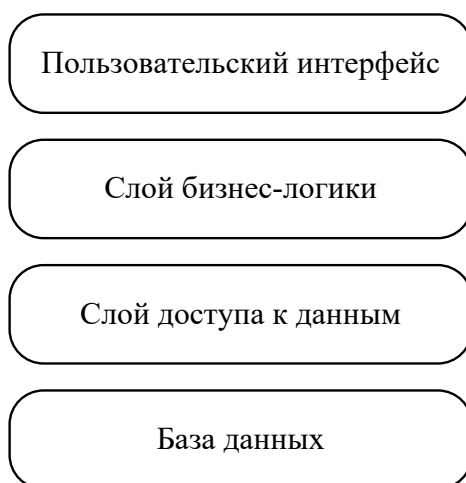


Рис. 1. Программная система с монолитной структурой

По мере по мере роста и развития программного продукта, становятся очевидными следующие проблемы монолитных архитектур:

- непрерывный рост сложности программной системы;
- проблематично выполнение поддержки и масштабирования программной системы;
- сложность тестирования монолитной структуры;
- дорогостоящее внесение изменений в систему;
- устаревание используемых технологий.

Микросервисная архитектура позволяет решить вышеперечисленные проблемы и открывает ряд возможностей, которые недоступны при использовании монолитной структуры. Микросервисная архитектура является подмножеством сервис-ориентированной, и характеризуется разделением системы на компоненты (сервисы) с группировкой компонент по бизнес-задачам, которые они решают. Данный архитектурный подход позволяет децентрализовать управление системой и данными, автоматизировать развертывание и мониторинг сервисов, а также выбрать и использовать наиболее простой способ при коммуникации между ними [1].

С точки зрения сервис-ориентированных архитектур и, в частности, микросервисов, децентрализованное хранение – один из ключевых моментов данного подхода. Под децентрализованным хранением подразумевается использование отдельной базы данных для каждого микросервиса, при этом базы данных между собой никак не взаимодействуют (рисунок 2).

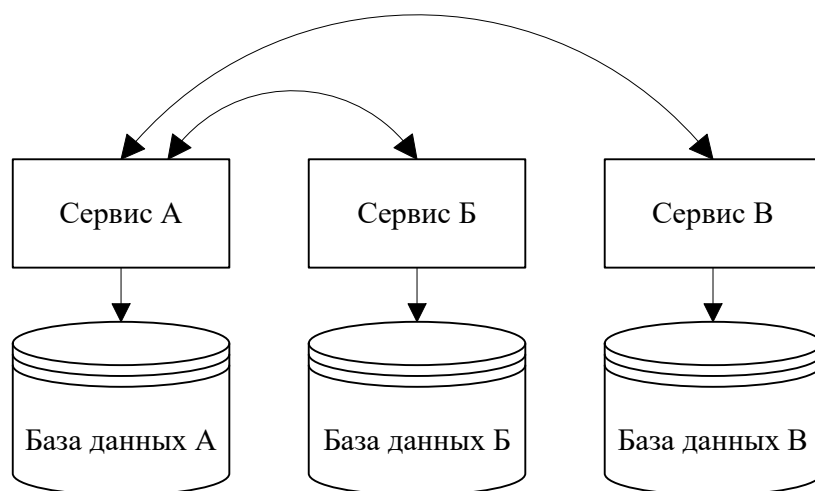


Рис. 2. Программная система с микросервисной структурой

При использовании микросервисного архитектурного подхода раскрывается ряд преимуществ:

- модульная структура, которая достигается благодаря модульному делению программной системы;
- изолированность: при разворачивании микросервисов в рамках одного домена приложения, будут получены изолированные модули, взаимодействующие между собой через оперативную память, минуя накладные сетевые расходы;
- доступность: отсутствие обязательного требования к функционированию всех сервисов системы в любой момент времени. Идеализированной структурой является система, в которой каждый сервис не влияет на работоспособность всех остальных;
- гибкость: широкие возможности выбора инструментов для разработки и использования наиболее подходящих программных средств в рамках определенного микросервиса;
- независимость при развертывании: благодаря слабой связанности сервисов достигается простота разворачивания системы и минимизируется вероятность её отказа, достигается отказоустойчивость и надежность системы в целом.

Однако следует учитывать ограничения, которые накладывает данный подход:

- сложность разработки необходимого набора микросервисов и программной системы в целом;
- сложность операционной поддержки: востребованность высококвалифицированных инженеров и администраторов, осуществляющих непрерывное развертывание и

реализующих автоматический мониторинг сервисов, без которых использование микросервисов теряет смысл.

- Немаловажным моментом является организация связи между микросервисами. Существует три общепринятых подхода для реализации их коммуникации [2]:
- Service Discovery (Remote Procedure Call style) – сервисы взаимодействуют друг с другом напрямую посредством удаленного вызова процедур;
- Message Bus (Event-driven architecture) – сервисы используют шаблон «издатель-подписчик» (рисунок 3). При использовании данного подхода «подписчику» неизвестны сервисы, которые на него подписаны, а «издателю» неизвестно какой из сервисов отправляет данные. Таким образом, главным для микросервиса является только содержимое определенного типа;
- Hybrid – смешанный подход, при котором используется как Remote Procedure Call, так и Message Bus, в зависимости от конкретной ситуации.

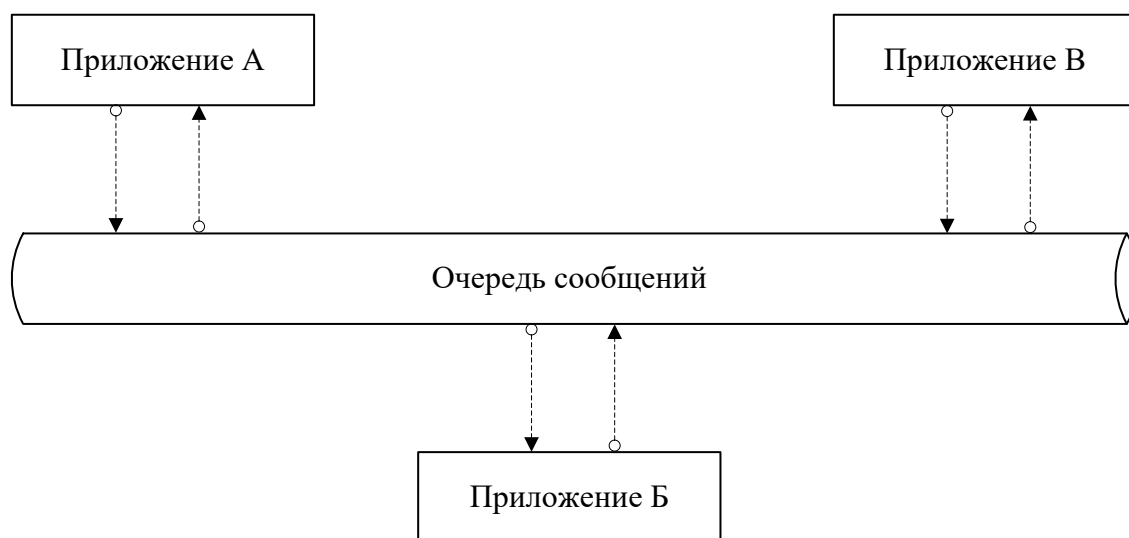


Рис. 3. Коммуникация между сервисами, используя шаблон «издатель-подписчик»

Самым распространенным способом взаимодействия компонентов распределенного приложения в сети является Representational State Transfer (REST). Это такой архитектурный стиль, который описывает набор ограничений, необходимых при проектировании распределенной системы и является альтернативой удаленному вызову процедур (RPC). Для взаимодействия между сервисами используется удаленный вызов процедур посредством HTTP-запросов, так называемых REST-запросов. В таких запросах данные передаются в качестве параметров либо в теле запроса, используя JSON-сериализацию.

Данный подход является приемлемым для несложных программных систем, однако по мере увеличения сложности проекта недостатки REST API становятся очевидными. Стоит отметить, что REST – это лишь архитектурный стиль, а не строго задокументированная спецификация. Из этого вытекает отсутствие единого соглашения, согласно которому будет использоваться тот или иной HTTP метод, отсутствуют четко определенные критерии для выбора данных, подлежащих передаче либо в URI, либо в теле запроса.

REST подход обладает немалым количеством сущностей, таких как совокупность HTTP-методов, наличие заголовков запроса и возможность их произвольной интерпретации, что влечет за собой ряд проблем, возникающих при проектировании API, нацеленного на использование различными сервисами.

В результате возникает проблема сложности декларативного описания интерфейса API и описания используемых типов данных. Ещё одним недостатком является

дорогостоящая процедура синхронизации клиентов и серверов при использовании интерфейса REST API, подверженного постоянному развитию и модификации.

Таким образом, можно сделать вывод, что при проектировании сложной системы нужен совершенно другой подход, при котором станет возможной быстрая передача данных с минимальным размером сообщения, предоставляющий возможность балансировки нагрузки. Для организации такой системы наиболее подходящим является использование высокопроизводительного каркаса gRPC, предназначенного для удаленного вызова процедур (RPC) и использующего сетевой протокол HTTP/2 [3] (рисунок 4).

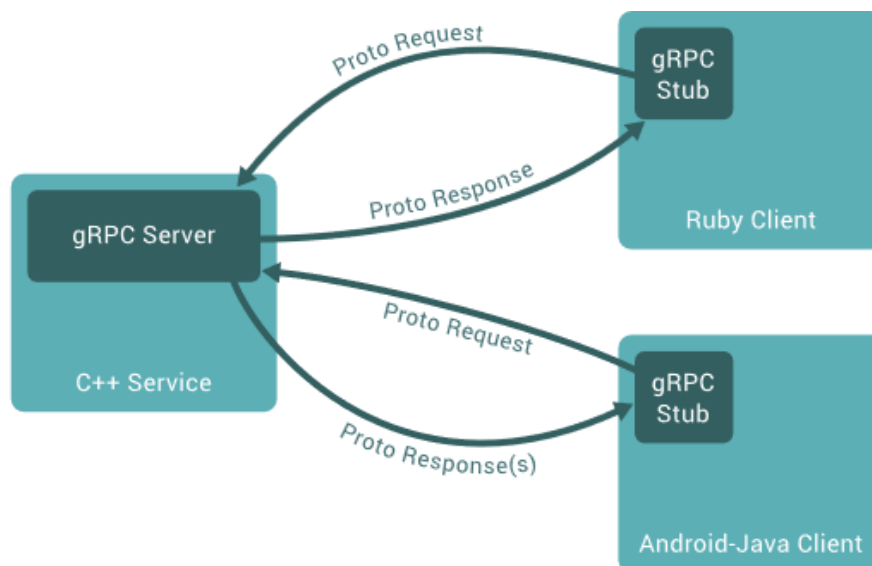


Рис. 4. Использование gRPC для коммуникации с микросервисом

Программная система, использующая gRPC имеет возможность прямого вызова методов, для которых строго заданы параметры и явно указаны типы данных. Задача сериализации, десериализации и передачи структурированных данных решается при помощи протокола Protocol Buffers (protobuf). Благодаря строгой типизации полей и бинарному формату данных минимизируется потребление ресурсов, а производительность системы значительно возрастает.

В общем виде формат передаваемых данных представляет собой закодированную последовательность полей «ключ-значение». В данном случае в качестве ключа выступает номер поля сообщения в proto-файле. Перед каждым полем указывается номер в формате varint, а также тип поля. При записи строки, вложенного сообщения, набора байт или повторяющегося сообщения, то вместе с ним передается размер записываемых данных в формате varint. Благодаря восьмиразрядному типу данных varint достигается уменьшение размера пакета при передаче небольших чисел, однако объем данных значительно возрастает если требуется передать числовые значения, близкие к максимально допустимым.

При использовании varint старший бит байта числа, выставленный в 1, является флагом окончания последовательности байт, составляющих число. В результате преобразования в стандартное представление, старший бит каждого байта отбрасывается, а все оставшиеся биты соединяются между собой. Для кодирования отрицательных чисел используется алгоритм ZigZag, суть которого заключается в переносе бита знака из старшего разряда в младший [4].

Для описания API микросервиса создаются protobuf-файлы, использующие язык спецификаций для описания интерфейсов (Interface Definition Language). Созданный protobuf-файл с описанием структур данных, процедур и методов компилируется в соответствии с языком программирования микросервиса, в результате чего будут получены

простейшие классы доступа к данным и методы для сериализации и десериализации описанных структур из байтов. Данный подход позволяет модифицировать и масштабировать созданные структуры данных без потери обратной совместимости.

Основные преимущества, которые предоставляет использование каркаса gRPC при реализации микросервисной архитектуры программной системы [3]:

- синхронная и асинхронная связь между узлами. Возможность использовать HTTP-синхронный и асинхронный запрос/отклик или двунаправленную потоковую передачу, аналогичную протоколу полнодуплексной связи WebSocket;
- поддержка SSL/TLS аутентификации и шифрования передаваемых данных;
- token-аутентификация через сервис Google и поддержка технологии OAuth2;
- поддержка балансировки нагрузки программной системы и контроля работоспособности микросервисов (рис. 5).

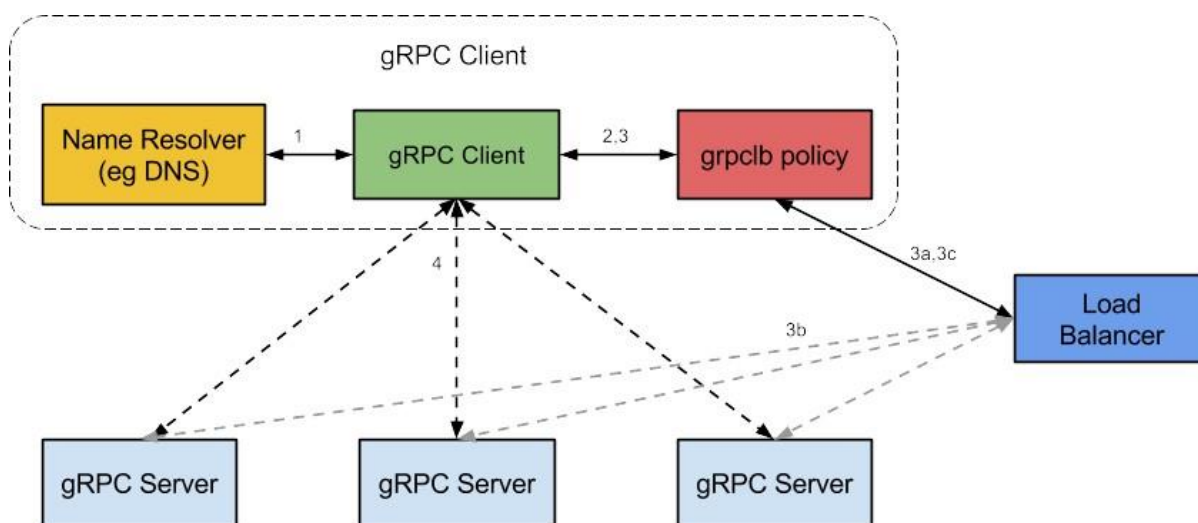


Рис. 4. Балансировка нагрузки программной системы

Таким образом, получена программная система, в которой каждый микросервис решает четко поставленную задачу, используя наиболее подходящий язык программирования, необходимый для решения задачи набор библиотек и наиболее удачную конфигурацию оборудования для разворачивания. В результате чего достигается горизонтально масштабируемый и отказоустойчивый программный код, обладающий конвейерным параллелизмом. Рассмотрены различные подходы для организации структуры программной системы, выбраны инструменты для реализации микросервисной архитектуры, их преимущества и недостатки.

Библиографический список

1. Chris Richardson Introduction to Microservices. [Электронный ресурс]. URL: <https://www.nginx.com/blog/introduction-to-microservices/> (дата обращения 22.03.2018)
2. Chris Richardson Microservice Architecture. [Электронный ресурс]. URL: <http://microservices.io/> (дата обращения 23.03.2018)
3. What is gRPC. [Электронный ресурс]. URL: <https://grpc.io/docs/guides/> (дата обращения 24.03.2018)
4. Protocol Buffers. [Электронный ресурс]. URL: <https://developers.google.com/protocol-buffers/docs/tutorials> (дата обращения 24.03.2018)

УДК 681.3

Д.А. Захаров, студент 4-го курса

Е.Н. Заикина, старший преподаватель

Севастопольский государственный университет

**РЕАЛИЗАЦИЯ ПЛАТФОРМЫ ДЛЯ ПРОВЕДЕНИЯ ВЫБОРОВ
ПРИ ИСПОЛЬЗОВАНИИ ТЕХНОЛОГИИ БЛОКЧЕЙН**

Аннотация: В данной статье ставится задача рассмотреть технологию блокчейн и возможность использования технологии блокчейн в современных системах, на примере платформы для выборов. Для управления голосованием используется технология «умных контрактов».

Ключевые слова: Блокчейн, технологии хранения данных, платформа для проведения выборов, концепция прозрачных выборов, умные контракты.

Annotation: In this article is considered the issue of examining the blockchain technology and the possibility to apply it in the modern systems using the example of the platform for elections. For controlling the elections the technology "Smart contract" is used.

Key words: Blockchain, storage technologies, a platform for elections, the development of transparent elections, smart contracts.

Введение

В настоящее время, в период информатизации современного общества и развития информационных технологий, информация играет важную роль. С развитием технологий, также развиваются требования к информации. Помимо основных требований к информации, важными требованиями являются надёжность и достоверность.

В повседневной жизни каждый человек ежедневно делает выбор. Выбор человека – это его желание одного события или нежелание другого. В более глобальном масштабе выбор человека может выражаться в процессе голосования. Голосование – это процесс, в котором выбор человека может изменить не одного человека, а целое сообщество. Как и к информации, так и к процессу выборов ставятся требования надёжности и достоверности.

Выборы – это процесс обработки информации, и важным является то, что этот процесс не должен изменять или подменять данные, или голоса, которые отдают избиратели и результат выборов должен полностью соответствовать совокупности всех исходных данных.

Необходимо фиксировать каждый голос и не давать возможности дублирования одного голоса либо изменения уже существующего.

Исходя из выше перечисленных требований определим задачу в реализации платформы для выборов. Платформа должна предоставлять коммерческим и некоммерческим предприятиям возможность проведения выборов в открытой форме, где каждый может увидеть процесс голосования, но соблюдая анонимность.

Необходимо реализовать требования надёжности и достоверности по отношению как к самой платформе для выборов, так и к результатам выборов и получить доверие пользователей системы.

Исходя из выше сказанного, для хранения информации о выборах будем использовать технологию блокчейн и технологию «умных контрактов» [1].

1. Технология блокчейн.

Блокчейн— выстроенная непрерывная последовательная цепочка блоков, содержащих информацию. Впервые этот термин появился в 2008 году, как название полностью реплицированной распределённой базы данных для криптовалюты «Биткойн». Но, следует заметить, что технология может и должна использоваться в современных информационных системах. Так как технология блокчейн является распределённой, все данные хранятся реплицировано и удаленно. Это гарантирует надёжность и достоверность данных и защиту от несанкционированного доступа к одному из удалённых узлов.

На рисунке 1 изображена структурная схема блоков транзакций.

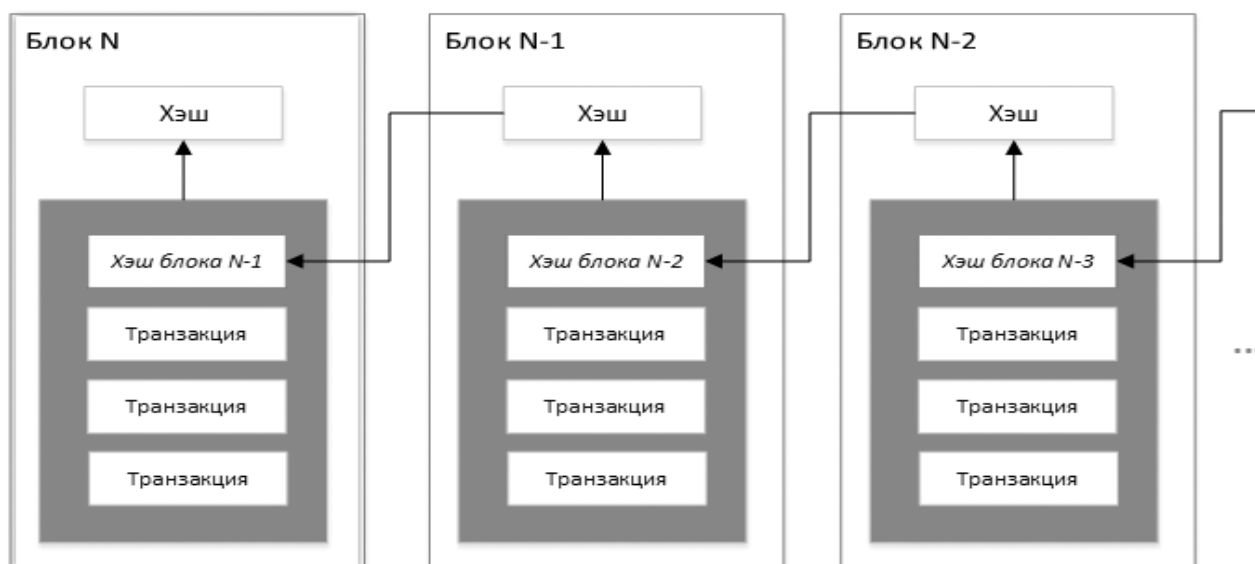


Рис. 1. Структурная схема блоков транзакций.

Блок транзакций — это специальная структура для записи группы транзакций. Транзакция считается завершённой и подтверждённой, только и только тогда, когда проверены её формат и подписи, и когда сама транзакция объединена в группу с несколькими другими и записана в специальную структуру — блок. Блок транзакций состоит из заголовка и списка транзакций. Заголовок блока транзакций включает в себя свой хеш, хеш предыдущего блока, хеши транзакций и служебную информацию.

Определим ряд преимуществ технологии, исходя из вышесказанного:

1) данные хранятся удаленно и распределенно;

При хранении данных используется реплицирование данных, что позволяет повысить безопасность системы. Каждый раз при обращении к необходимым данным происходит синхронизация со всеми узлами и проверкой данных.

2) прозрачность данных;

Все блоки доступны для публичного просмотра.

2. Умные контракты

Умный контракт — компьютерный алгоритм, предназначенный для заключения и поддержания коммерческих контрактов в технологии блокчейн.

Первые идеи смарт-контрактов были предложены в 1994 году Ником Сабо. Он описал смарт-контракт как компьютерный протокол, который на основе математических алгоритмов самостоятельно проводит сделки с полным контролем над их выполнением.

Умные контракты позволяют обмениваться денежными единицами, не прибегая к услугам посредников. Умные контракты содержат информацию об обязательствах сторон, штрафы за их нарушение, и сами автоматически обеспечивают выполнение всех условий договора. Документ договора (контракт) храниться в децентрализованном реестре и дублируется, что обеспечивает его надёжность и не позволяет ни одной из сторон менять условия соглашения.

Умные контракты записываются в виде кода, существующего в распределенном реестре — блокчейне, который поддерживается и управляется сетью компьютеров. И запрещают редактирование смарт-контракта одной из сторон.

Обычно смарт-контракт записывается в блокчейн, где вся его логика помещается в программный контейнер — блок. Последний объединяет все сообщения, относящиеся к конкретному смарт-контракту.

Исходя из вышесказанного, определим следующие преимущества умных контрактов:

1. Автономность: отсутствуют посредники при оформлении контракта.
2. Надёжность и безопасность: контракт хранится в зашифрованном виде, в распределённом реестре.
3. Надёжность хранения: все документы продублированы в блокчейне.
4. Экономия и скорость: ускоряются посредники, за счёт чего повышается скорость выполнения.

У умных контрактов есть особенность, связана она с тем, что умный контракт нужно записывать без ошибок и учитывать все условия и детали сделки, так как изменить контракт не будет возможности.

3. Использование технологий блокчейн и умных контрактов в процессе проведения выборов

Исходя из выше сказанного, заметим, что при использовании технологий блокчейн и умных контрактов будут достигнуты необходимые требования как к самой информации, так и к площадке.

При реализации голосования каждый голос должен проходить через алгоритм умных контрактов, в процессе чего происходит подписание контракта. После подписания контракта все данные о голосе хранятся удаленно и распределенно в блоках. Сам контракт хранится в реестре контрактов в блокчейне, и редактировать контракт не предоставляет возможности.

Так как блокчейн хранит данные публично, то площадка будет отображать в режиме онлайн результаты выборов и сами голоса избирателей, что отображает процесс прозрачных выборов. При публикации голосов будет соблюдена анонимность избирателей.

Подводя итоги, можно сказать, что площадка будет отвечать всем требованиям, которые возлагались на неё, при использовании технологий блокчейн и умных контрактов.

Библиографический список

1. Различия, достоинства, недостатки: публичные и приватные блокчейны [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/company/bitfury/blog/324458/> – (Дата обращения: 21.01.2018).

УДК 004.7

Д.А.Мачавариани, студент 4-го курса

В.А.Строганов, старший преподаватель

Севастопольский государственный университет

WEB-СИСТЕМА “ПАНЕЛЬ УПРАВЛЕНИЯ QIWI КОШЕЛЬКАМИ”

Аннотация: В данной статье описывается веб-приложение для управления Qiwi кошельками. Автор описывает актуальность разрабатываемого приложения, подчеркивает его преимущества по сравнению с официальным сайтом. Рассмотрены основные функции, которые можно будет выполнить в приложении.

Ключевые слова: киви, кошелек, автоматизация.

Annotation: This article describes the web application for the management of Qiwi wallets. The author describes the relevance of the developed application, emphasizes its advantages over the official website. The basic functions that can be performed in the application are considered.

Keywords: Qiwi, wallet, automation.

На данный момент существует всего лишь один способ взаимодействия с Qiwi кошельком - это официальный сайт, на котором при помощи логина и пароля можно попасть в личный кабинет и получить доступ к своему кошельку. С прошлого года у Qiwi появился свой API [1], который позволяет автоматизировать управление кошельком.

Узнав об этой возможности, было принято решение создать приложение, с помощью которого можно было бы управлять сразу несколькими кошельками. Актуальность темы заключается в том, что в современной интернет-коммерции зачастую есть необходимость использования нескольких кошельков для автоматического перераспределения средств на кошельках. Разработанное приложение позволит сэкономить большое количество времени на рутинную работу перераспределения средств. Для того чтобы управлять кошельком за пределами официального сайта, необходимо получить специальный токен для каждого кошелька который вы хотите добавить в систему. Токен работает всего лишь несколько месяцев, после чего его необходимо перевыпускать. После того как токен получен, возможно приступить к работе с приложением. Для этого после авторизации необходимо заполнить поле токена и номера телефонов, нажать кнопку «добавить». Пользователь может управлять кошельком (см. рисунок 1).

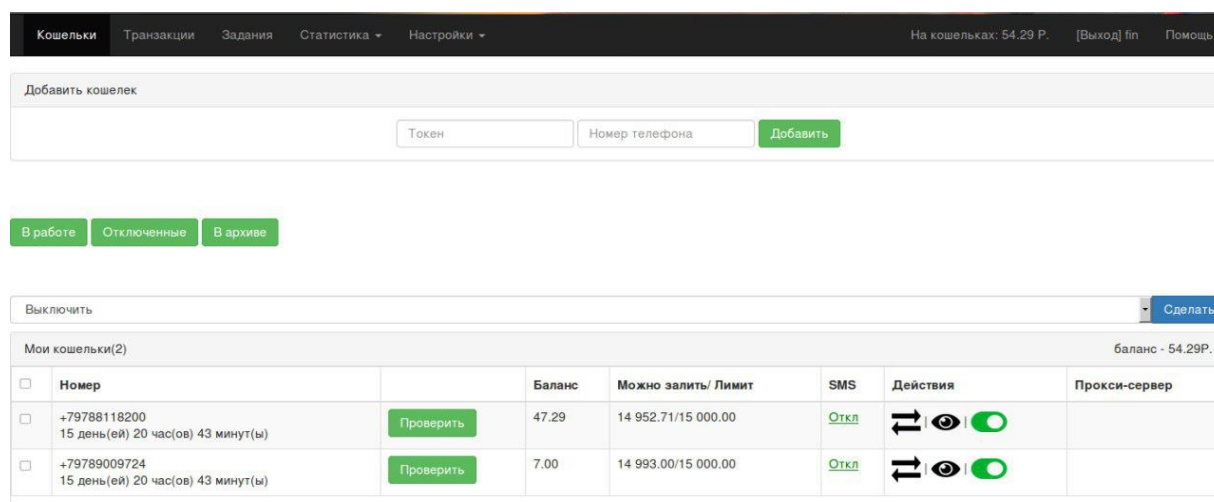


Рис. 1 – Режим просмотра персонального кабинета

После добавления нужного количества кошельков можно совершить с ними следующие действия:

1. Проверить баланс
2. Перевести средства
3. Создать задание на перевод
4. Увидеть историю транзакций
5. Автоматизировать работу с переводами.

6. Работа с API для выдачи кошельков.

7. Проверка платежей.

Вид страницы конкретного кошелька приведен на рисунке 2.

The screenshot shows the Qiwi wallet interface. On the left is a form titled 'Вывод средств' (Withdrawal). It includes fields for 'Платежная система:' (Payment system) with a dropdown menu showing 'test1y', 'Назначение:' (Destination) with a dropdown menu showing 'Выберите метку' (Select tag), 'Сумма:' (Amount) with a text input field, 'Номер кошелька:' (Wallet number) with a text input field, 'Комментарий:(необязательно)' (Comment:(optional)) with a text input field, and 'Платежный пароль:' (Payment password) with a text input field. At the bottom of the form is a green button labeled 'Подтвердить платеж' (Confirm payment). On the right is the wallet balance section. It shows the wallet number '+79788118200' and a green button labeled 'Проверить' (Check). Below this, it shows the balance 'Баланс: 0 Р.' and the amount that can be topped up 'Можно залить: 180000 Р. из 180000 Р.' (Can top up: 180000 R. from 180000 R.). The owner is listed as 'Владелец: firefox'. Below the balance section is a section titled 'Запрос истории' (Request history) with links for 'За сутки' (For 24 hours), 'За неделю' (For a week), 'За месяц' (For a month), and 'За период' (For a period).

Рис. 2. Режим просмотра страницы кошелька

Приложение было создано для автоматизации некоторых процессов, которые на официальном сайте Qiwi выполнены не очень удобно, или же совершить их было невозможно, а именно:

1. Работа с несколькими кошельками.
2. Автоматизация перевода больших сумм.
3. Автоприём и подтверждение платежей для сторонних проектов.

Все эти проблемы приложение решает, так как имеется возможность добавлять неограниченное количество кошельков, следовательно, можно работать сразу с большим числом кошельков. В случае с переводом больших сумм Qiwi ограничивает максимальный размер транзакции, а с помощью приложения возможно поставить задание на перевод суммы значительно превышающей максимальную. Система сама будет переводить средства с кошелька на кошелек в необходимом размере, и в конечном итоге переведёт всю сумму, указанную в задании. Также возможен автоприём средств и подтверждение платежей.

Разработан публичный API для взаимодействия с приложением, с его помощью можно получить номер кошелька для перевода, в который ещё можно перевести средства на ту сумму, которую запрашивает автор, при обращении к API на предмет получения средств также можно определить оплатил ли клиент полную сумму, или же нет.

Приложение не имеет встроенной регистрации, поэтому все аккаунты должны создаваться программистом путём ручного внесения данных в базу данных.

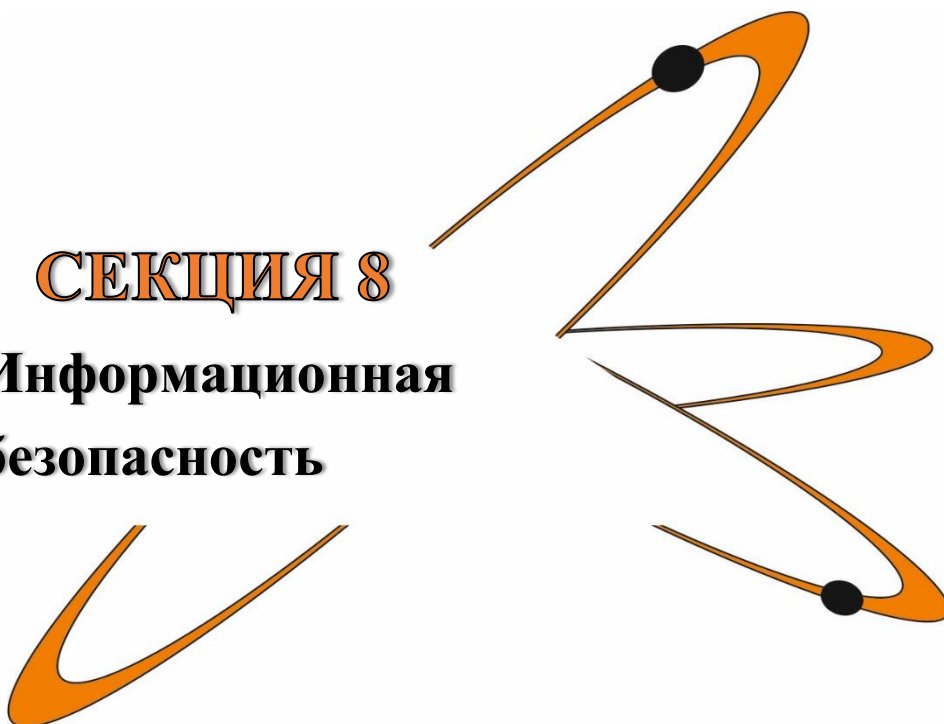
Приложение будет полезно во многих сферах интернет-коммерции, будь то интернет магазины, игровые сайты, оплата услуг.

Разрабатываемое приложение имеет интуитивно-понятный интерфейс, все формы имеют описание и подсказки в случае неправильного заполнения. При написании приложения использовался язык программирования php, php-фреймворк Yii2, в качестве СУБД использовалась MariaDB.

Библиографический список

1. Официальный сайт с API Qiwi. Режим доступа: <https://developer.qiwi.com/ru/qiwi-wallet-personal/> (дата обращения: 25.03.2018)

СЕКЦИЯ 8
**Информационная
безопасность**



УДК 004.056

А. А. Акимов, студент 4-го курса**А. А. Белоног, студент 4-го курса****М. А. Лебедева, старший преподаватель***Севастопольский государственный университет***ВИРТУАЛЬНАЯ ЧАСТНАЯ СЕТЬ НА БАЗЕ БРАНДМАУЭРА**

Аннотация: статья посвящена перспективному направлению защиты информации при передаче по сети, состоящему в построении виртуальных частных сетей. Данная технология обеспечивает конфиденциальность, целостность и доступность передаваемой информации за счет механизмов шифрования, аутентификации, авторизации и туннелирования.

Ключевые слова: виртуальная частная сеть, шифрование, идентификация, аутентификация, брандмауэр, туннелирование.

Annotation: the article is devoted to the perspective direction of information security during transmission over the network, consisting in the construction of virtual private networks. This technology ensures confidentiality, integrity and availability of transmitted information through encryption, authentication, authorization and tunneling mechanisms.

Key words: virtual private network, encryption, identification, authentication, firewall, tunneling.

Сеть Internet широко используется в качестве коммутационного канала, позволяющего удаленным пользователям обмениваться информацией. Протоколы передачи информации по глобальной сети не обеспечивают защищенности данных. Одним из эффективных средств защиты передаваемой информации, гарантии ее конфиденциальности и целостности является технология виртуальных частных сетей (VPN), которая заключается в создании защищенного канала поверх внешней сети, например, Internet. Протоколы внешней сети должны осуществлять пакетную передачу данных. VPN могут быть реализованы на канальном, сетевом и сеансовом уровнях OSI. Защита информации в виртуальных частных сетях обеспечивается применением криптографических алгоритмов и методов идентификации и аутентификации в сочетании с механизмом туннелирования, заключающегося в шифровании пакета вместе со служебной информацией [1].

Виртуальные частные сети строятся на базе брандмауэров, маршрутизаторов, аппаратного и программного обеспечения.

При построении VPN на базе брандмауэра межсетевой экран является VPN-агентом, который обеспечивает фильтрацию входящего и исходящего сетевого трафика и выполняет функцию туннелирования (рис.1). Достоинством этого метода является экономичность, к недостаткам относится зависимость производительности от аппаратного обеспечения.

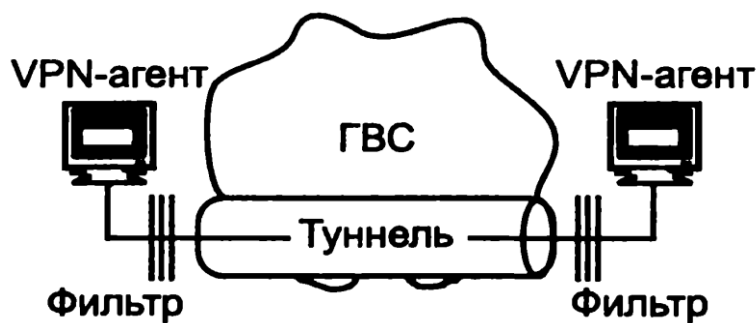


Рис 1. Механизмы VPN

VPN на базе брандмауэра строятся на сетевом уровне OSI. Средства защиты являются прозрачными для приложений и прикладных протоколов. Фильтрация сетевого трафика выполняется на основе правил, основанных на анализе заголовка сетевого пакета. Брандмауэр пропускает или отбрасывает пакеты на основе анализа таких данных из заголовка пакета, как тип протокола, IP-адрес отправителя, номер порта. При этом

целесообразно использовать строгую политику брандмауэра: запретить доступ для пакета, если он явно не разрешен. В этом случае соединение с удаленным пользователем будет установлено только в том случае, если IP-адрес в заголовке полученного пакета будет совпадать с адресом из разрешенного списка. В противном случае пакет будет отброшен. Недостатком аутентификации на основе адреса отправителя является то, что она не защищает от подмены адресов злоумышленником.

Создание безопасного туннеля включает:

- аутентификацию пользователя;
- шифрование данных ip-пакета;
- вычисление хэш-функции и объединение с зашифрованными данными;
- инкапсуляцию полученного пакета в другой пакет, заголовок которого вместо исходной информации об адресате и отправителе содержит соответственно информацию о VPN-агенте адресата и VPN-агенте отправителя.

Каждый VPN-агент хранит ключи шифрования для всех возможных соединений.

Инициатор VPN соединения высылает пакет, содержащий аутентификационные данные. Если результат фильтрации положительный, производится взаимная аутентификация участников соединения, которая может проводиться на основе симметричной или асимметричной криптографии.

Симметричная криптография лежит в основе протокола Берроуза [2], который используется для взаимной идентификации и процесса распределения ключей в системе. Протокол взаимной аутентификации называется также протоколом рукопожатия. Аутентифицирующей информацией является секретный ключ, который разделяют VPN-агенты. При иницировании соединения первый VPN-агент (V_1) посылает второму VPN-агенту (V_2) случайное число r_1 , V_2 зашифровывает с помощью алгоритма симметричного шифрования E_K полученное случайное число и случайное число r_2 , сгенерированное им. V_1 расшифровывает полученное сообщение с помощью алгоритма дешифрования D_K , если оно корректно, то посылает V_2 для проверки зашифрованную последовательность r_2, r_1 .

Алгоритмы E_K и D_K представляет собой блочный симметричный шифр. Блочные шифры являются итерационными, обрабатывают информацию блоками определённой длины (обычно 64, 128 бит). На каждой итерации (раунде шифрования) выполняются операции перемешивания и подстановки с участием секретного ключа. Результатом повторения раундов является лавинный эффект — нарастающая потеря соответствия битов между блоками открытых и зашифрованных данных [2].

Повышения производительности VPN можно добиться, используя легковесную криптографию, основная цель которой — минимизация ресурсов для реализации алгоритмов шифрования с сохранением их криптостойкости и быстродействия. Хорошо изученным и проверенным на практике является низкоресурсный шифр Tea (рис.2). Это блочный шифр, основанный на сети Фейстеля, с размером блока 64 бит и длиной ключа 128 бит.

Шифруемый блок делится на две равные части — ветви X и Z . На каждом раунде над ветвями X и Z выполняются простые преобразования: сложение чисел по модулю 2^{32} , побитовое исключающее «ИЛИ», побитовый сдвиг влево и вправо. В преобразованиях участвует константа δ , выведенная из золотого сечения (1):

$$\delta = (\sqrt{5} - 1) \cdot 2^{31} \quad (1)$$

В каждом раунде константа δ умножается на номер цикла i . Это было сделано для предотвращения простых атак, основанных на симметрии раундов. Алгоритм устойчив к линейному и дифференциальному криптоанализу, но требует тщательного выбора ключей.

VPN агент выполняет шифрование пакета вместе с заголовком, обеспечивая тем самым защиту данных от несанкционированного доступа.

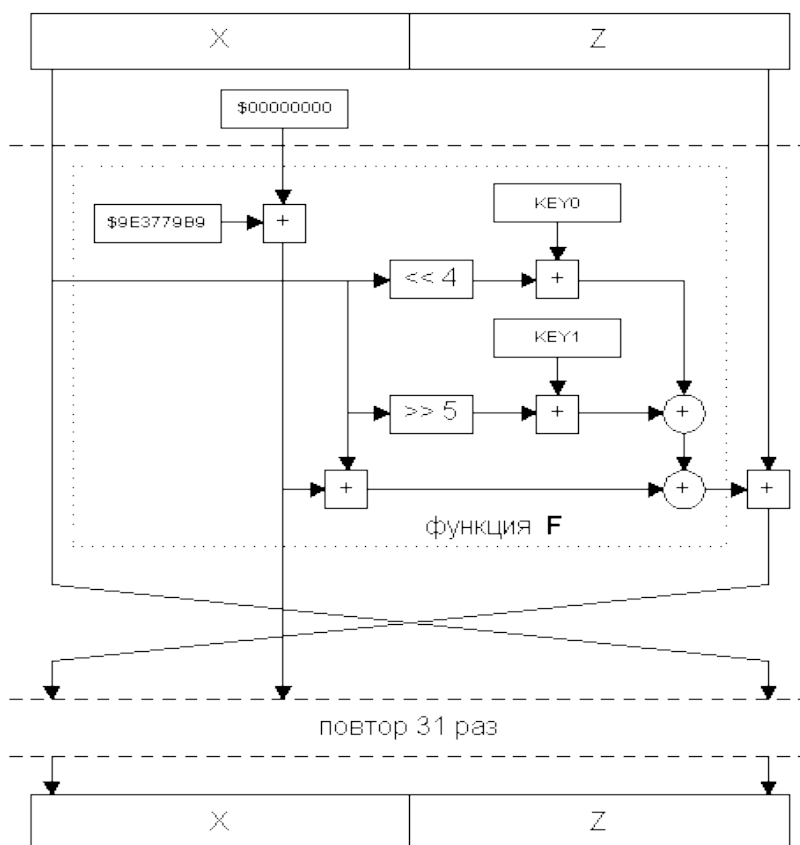


Рис.2 Схема блочного шифра Tea

Для обеспечения целостности информации используют хэш-функции или MAC-коды, которые требуют меньше временных и вычислительных ресурсов. В основе MAC-кодов [2] лежат блочные симметричные алгоритмы. MAC-код H является разновидностью криптографической функции хеширования на основе одношаговой функции сжатия по итеративной схеме Меркеля – Дамгард (рис. 3).

Сообщения M разбивается на m -битовых блоки M_i , $|M_i| = m$, при необходимости предварительно дополняется, так чтобы длина увеличенного сообщения была кратна числу m , где m – длина строки, которую может обработать функция сжатия.

Последовательная процедура вычисления MAC-кода H :

$$H_0 = IV \quad H_i = f(M_i, H_{i-1}) \quad H(M) = H_n \quad (2)$$

где H_i – свертки, получаемые в ходе итераций, n – число блоков. IV – начальный вектор.

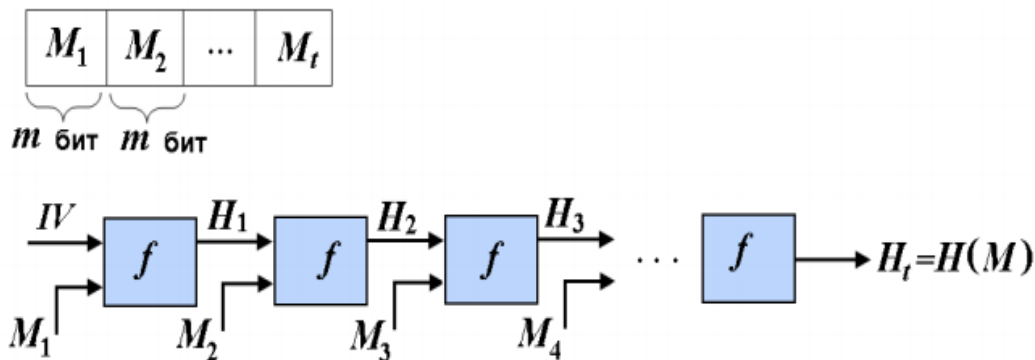


Рис.3 – Схема получения MAC-кода

В качестве функции f используется алгоритм блочного шифрования в режиме шифрования со сцеплением блоков (CBC), результатом является последний блок:

$$H_0 = IV \quad H_i = E_K(M_i \text{ xor } H_{i-1}) \quad H(M) = H_n \quad (3)$$

где K – криптографический ключ блочного алгоритма.

Безопасность MAC-кода базируется на безопасности используемого шифра

К зашифрованному пакету VPN агент добавляет MAC-код.

Защищенный пакет с помощью механизма инкапсуляции помещается в готовый для передачи IP-пакет, заголовок которого вместо исходной информации об адресате и отправителе содержит соответственно информацию о VPN-агенте получателе и VPN-агенте отправителя и передается по сети.

VPN-агент получателя, приняв пакета, анализирует его заголовок с целью выделения информация о его отправителе. В случае, если отправитель не входит в число разрешенных (согласно политике безопасности) или неизвестен (например, при приеме пакета с намеренно или случайно поврежденным заголовком), пакет не обрабатывается и отбрасывается;

VPN-агент может находиться непосредственно на защищаемом компьютере. В этом случае с его помощью защищается информационный обмен только того компьютера, на котором он установлен, однако описанные выше принципы действия остаются неизменными. Защищающий локальную вычислительную сеть (ЛВС) VPN-агент может быть совмещен с маршрутизатором IP-пакетов, который также должен находиться на выходе из ЛВС. Основное правило построения VPN: связь между защищенной ЛВС и Интернетом должна осуществляться только через VPN- агенты, категорически запрещаются какие-либо способы связи, минуя защитный барьер в виде VPN-агента; иными словами, должен быть определен защищаемый периметр, связь с которым возможна только через соответствующее средство защиты.

Виртуальные частные сети позволяют решить задачу защиты информации от несанкционированного доступа внутри сети или отдельного узла и в процессе ее передачи по открытой сети. VPN широко используется для создания защищенных соединений в корпоративных сетях, доступа к WEB-сервисам, при подключении удаленных пользователей.

Библиографический список

1. Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. / В.Ф.Шаньгин – М.: «ДМК», 2012 – 592 с.: ил.
2. Шнайер Б. Прикладная криптография/ Б. Шнайер.—СПб.:Питер, 2003. — 368 с.

УДК 004.056.55

П.А. Киричук, магистрант,

Н.Л.Корепанова, канд. техн. наук, доцент

Севастопольский государственный университет

ПРАВИЛА ПАРОЛЬНОЙ ЗАЩИТЫ В СОВРЕМЕННЫХ СЕТЕВЫХ СТРУКТУРАХ*Аннотация: Рассматриваются результаты парольной защиты в зависимости от длины пароля и уровней взлома.**Ключевые слова: методы шифрования, парольная защита, уровни взлома**Annotation: Discusses the results of password protection depending on password length and levels of hacking attacks**Key words: encryption, password protection, levels of hacking attacks*

Главной уязвимостью сетевых структур является то, что зачастую в них используют простые пароли, либо обходятся без них. Обычно пароль устанавливает администратор или пользователь, обладающий правами администратора, который может поставить пароль высокой сложности. При этом пользователь может изменить пароль на более простой, что в свою очередь может нарушить безопасность системы, а затем безопасность всей сети в целом. Большинство пользователей использует лишь один пароль, как для входа в систему, так и для различных приложений типа почты и программ требующий авторизацию. Эти приложения хранят пароли в зашифрованном виде, но при этом существует риск взлома. Для проверки подлинности пользователя при использовании его логина запрашивается его пароль, он шифруется и сравнивается с уже имеющимся зашифрованным паролем. При совпадении пароля пользователь продолжает свою регистрацию в сети.

Существует ряд методов шифрования паролей [1-3]. При методе шифрования brute force (md5) пароль можно узнать только при помощи полного перебора. Метод выборки символов состоит в запросе системой определенных символов пароля, выбираемых случайным образом. Метод паролей однократного использования предполагает собой список из N паролей, которые хранятся в системе. Обращаясь к системе, пользователь вводит пароль, который после окончания сессии зачеркивается из списка, хранящегося в системе.

Идея метода генерации одноразовых паролей заключается в том, что имеется односторонняя функция F, которая известна пользователю и серверу авторизации. Есть ключ K, который известен пользователям. На начальном этапе авторизации пользователя функция F применяется к ключу несколько раз и результат хранится на сервере. Процедура проверки пользователя при этом имеет следующий вид:

- на систему пользователя сервер посылает число (n-1);
- пользователь применяет F к секретному ключу (n-1) раз и отправляет результат по сети на сервер авторизации;
- сервер применяет функцию F к полученному от пользователя значению и сравнивает результат с ранее сохраненной величиной. Если величина совпала, то пользователь авторизован, и сервер запоминает новое значение присланные пользователем.

Еще одним подходом надежной авторизации является генерация нового пароля, через заданный интервал времени для чего используют программы или интеллектуальные карты. Сутью метода группы паролей является то, что система для пользователя может потребовать пароль из двух групп. В первой группе включены пароли, которые являются ответами на общие вопросы, например, имя, фамилия, номер комнаты и т.п. Второй группой паролей будут ответы на эти вопросы, которые устанавливает администратор при регистрации нового пользователя. Эти вопросы составляют лично для каждого пользователя. Как только пользователь обращается к системе, она случайно выбирает по несколько вопросов из каждой группы. Минусом данной системы является то, что для реализации требуется большой объем памяти для хранения вопросов и ответов.

Метод функционального преобразования предполагает, что пользователю при регистрации для работы в системе сообщается некоторое преобразование, которое он может выполнить в уме. Паролем в этом случае является результат такого преобразования.

Паролирование является одним из наиболее незащищенных средств авторизации, поэтому для предотвращения несанкционированного доступа необходимо использовать проверку сложности паролей. При проверке сложности пароля можно использовался специальный контроллер, который проверяет пароль на уязвимости. Для этого он пытается взломать пароль. В качестве пароля используется имя пользователя, а также разные комбинации его имени и инициалов, а также использование словарей для проверки разных слов в качестве паролей. В результате исследований, данный контроллер определил 10% паролей, состоящих из пяти символов, 40% паролей из шести символов, 25% из семи и 20% при восьми символьном пароле. С учетом особенности метода полного перебора, стойкость пароля напрямую зависит не только от его длины, но и от используемых наборов символов при его генерации. Скорость перебора разных комбинаций зависит от таких факторов как используемое программное и аппаратное обеспечение хакеров. Проверка устойчивости паролей, на основе четырех уровней взлома, которые часто используются на практике, приведена в таблице 1 [4]. Четвертый уровень взлома базируется на линейке старых поколений (например, Pentium), с приблизительной частотой 800-900 МГц., в идеале получая за секунду 12 миллионов комбинаций подборов пароля. Третий уровень берет за основу Intel Core i5 на архитектуре «Sandy Bridge» и обрабатывает в секунду почти 28 миллионов комбинаций. С учетом ресурсов видеокарты среднего сегмента GeForce GTX 660 на втором уровне скорость подбора пароля возрастет до 516 миллионов комбинаций в секунду. На первом уровне суперкомпьютеры взломщики Deer Crack и COPACOBANA обладают скоростью 92 миллиарда комбинаций в секунду.

Таблица 1. Время для взлома восьмизначных паролей при различном уровне хакерской угрозы

Уровень угрозы	Арабские цифры	Латинский алфавит без регистра	Арабские цифры латинский алфавит без регистра	Латинский алфавит с регистром	Арабские цифры + латинский алфавит с регистром	Арабские цифры + латинский алфавит с регистром + спец. символы
IV	8.3 сек.	4.8 час.	2.7 сут.	51.6 сут.	210.6 сут.	17.5 год
III	3.6 сек.	2.1 час.	1.2 сут.	22.1 сут.	90.3 сут.	7.5 год
II	0.2 сек.	6.7 мин.	1.5 час.	1.2 сут.	4.9 сут.	148.8 сут.
I	0.0 сек.	2.3 сек.	30.7 сек.	9.7 мин.	39.6 мин.	20.0 час.

Проведем анализ случайно сгенерированных восьмизначных паролей, которые состоят только из арабских цифр. Как видно для взлома такого пароля вполне может справиться и оборудование с четвертым уровнем угрозы, что составляет по времени 8.3 секунды, а первый уровень справляется практически мгновенно. Случайно сгенерированные восьмизначные пароли, составленные только из арабских цифр, даже при четвертом уровне хакерской угрозы не могут продержаться более 8,3 секунды при атаке методом полного перебора, а при первом уровне угрозы они взламываются немедленно.

Пароль составленный из арабских цифр и латинского алфавита с регистром будет относительно устойчив к четвертым и третьим уровням угрозы. При четвертом уровне угрозы его следует менять раз в 3 месяца, а при третьем уровне угрозы раз 1-2 месяца. Это связано с тем, что подбор ключа к паролю занимает почти половину времени, которое требуется для перебора.

Восьмизначный пароль, составленный из арабских цифр, латинского алфавита с регистром и специальных символов, можно считать надежным при втором, третьем и четвертом уровнях хакерских угроз. При втором уровне угрозы в этом случае пароль следует менять раз в 2-2,25 месяца.

В заключение определим для каждого набора символов длину пароля, устойчивую к наиболее высокому первому уровню хакерской угрозы. Отметим сразу, что при таком уровне для взлома десятизначного пароля, состоящего только из арабских цифр, хакеру потребуется лишь 0,11 секунд. В то время как пароль, состоящий из букв латинского алфавита (без регистра), уже при длине в 13 и 14 символах потребует для взлома, соответственно, 312,1 дней и 22,2 года. В случае первого уровня угрозы пароль, состоящий из арабских цифр и букв латинского алфавита (без регистра), при длине в 11 символов потребует для взлома 1,6 года, а при 12 символах – 58,8 лет. Пароль, составленный из латинских букв с регистром, при длине 11 и 12 символов потребует для взлома, соответственно, 2,6 и 134,6 года. Пароль из арабских цифр и латинского алфавита с регистром при длине в 10 и 11 символах потребует для взлома, соответственно, 105,6 дней и 17,9 лет. И, наконец, пароль, состоящий из арабских цифр, латинского алфавита с регистром и специальных символов, уже при длине в 9 символов потребует для взлома 79,8 дней, а при длине в 10 символов – 20,6 года.

На базе вышерепредставленных теоретических исследований было проведено тестирование сервера Apache в рабочей системе, подверженной разным атакам. Атаки производились с целью получения информации к закрытым директориям, при таком типе атаки использовались специально написанные скрипты или перебор (brute-force). Большое количество ботов сканировали веб-сайт на разные типы уязвимости, а также искали e-mail адреса для отправки рекламы-спама. Для отслеживания попыток подбора был использован fail2ban-инструмент, который при помощи log-файла проверял обращения к сервисам SSH, FTP, HTTP. В случае обнаружения повторяющихся неудачных попыток авторизации, fail2ban определял это как попытку взлома и блокировал следующие попытки вторжения с этого адреса.

Библиографический список

1. Монахов Д.Н., Монахов Н.В., Прончев Г.Б., Кузьменков Д.А. Облачные технологии. Теория и практика.- М.: МАКС Пресс, 2013 - 128 с.
2. М.Я. Беккер, Ю.А. Гатчин, Н.С. Кармановский, А.О. Терентьев, Д.Ю. Федоров Информационная безопасность при облачных вычислениях: проблемы и перспективы. Научно-технический вестник Санкт-Петербургского государственного университета выпуск №1(71) 2011 г.
3. Корепанова Н.Л., Павленко Л.О. Оценка алгоритмов криптографии для облачных вычислений. В книге: Ломоносовские чтения. Сборник материалов научной конференции. 2017 г. С. 14.
4. Бирков В.П. Энтропия на службе секретности. [Электронный ресурс]. Режим доступа: <http://samag.ru/vart/more/53>. Дата доступа: 20.03.2018

УДК 681.3

С.В. Петров, студент 4-го курса

Т.Б. Пичугова, студентка 4-го курса

М.В. Заморенов, канд. техн. наук, доцент

Севастопольский государственный университет

ПРОГРАММНЫЙ КОМПЛЕКС ПРЕДОСТАВЛЕНИЯ УСЛУГ АВТОРИЗАЦИИ И АУТЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ

Аннотация: Краткое описание и предложение способов для проектирования и реализации унифицированной системы управления доступом к информационным ресурсам, а также рассмотрение возможных проблем при их внедрении.

Ключевые слова: авторизация, аутентификация, криптография, протокол, идентификация.

Annotation: authorization, authentication, cryptography, protocol, identification

Key words: Brief description and suggestion of ways to design and implement a unified system of access control to information resources, as well as consideration of possible problems in their implementation.

Для обеспечения требуемого уровня информационной безопасности при проектировании любой информационной системы (ИС) разработчикам приходится принимать решения о добавлении подсистемы для авторизации и аутентификации пользователей. Чаще всего, принятые решения и их реализации совпадают у разных разработчиков. Исходя из этого, эффективность разработки падает из-за дублирования программного кода и решения одинаковых проблем проектирования процедур авторизации и аутентификации.

Для решения проблемы низкой эффективности этой части разработки, связанной с управлением доступом пользователей, можно воспользоваться готовыми протоколами, отестированными и стандартизованными различными комитетами - решение повышающее эффективность при проектировании ИС. Для решения проблемы реализации свода протоколов авторами предлагается прототип программной системы, которую любой разработчик может интегрировать в разрабатываемую им ИС.

Рассмотрим понятия авторизации и аутентификации.

Авторизация — предоставление определённого лицу или группе лиц прав на выполнение определённых действий; а также процесс проверки (подтверждения) данных прав при попытке выполнения этих действий [1].

В качестве примера можно рассмотреть возможность добавления новой записи в личном блоге. Данную операцию, зачастую, может выполнить только владелец блога, как субъект, авторизованный для выполнения данной операции.

Аутентификация [2] – процедура проверки подлинности, например:

- проверка подлинности пользователя путем сравнения введённого им пароля с паролем, сохранённым в базе данных пользователей;
- подтверждение подлинности электронного письма путем проверки цифровой подписи письма по открытому ключу отправителя;
- проверка контрольной суммы файла на соответствие сумме, заявленной автором этого файла.

Для аутентификации в Российской федерации разработан стандарт ГОСТ Р ИСО/МЭК 9594-8-98, который был принят в 1999 году. В то время еще не было протоколов HTTPS и TLS, которые сейчас являются примитивами для защиты данных в интернет пространстве. Следовательно, для настоящего времени этот стандарт можно считать устаревшим и мало применимым в текущих реалиях разработки программного обеспечения.

В рамках текущего прогресса протоколов передачи данных и защиты передаваемой информации через данные протоколы (HTTPS в частности), существуют открытые протоколы для авторизации - OAuth 2.0 в виде RFC 6749 [3] и для аутентификации OpenID Connect [4].

Протокол OAuth 2.0 описывает свод процедур для получения т.н. JWT-токена, который защищен механизмом который гарантирует что токен был получен от доверенного сервиса и исключает его подделку. Протокол OpenID Connect определяет, как субъект может проходить аутентификацию. Оба протокола совместимы и могут использоваться друг с другом, точнее, OpenID Connect основан на базе OAuth 2.0.

OAuth 2.0 работает по схеме, изображенной на рисунке 1.

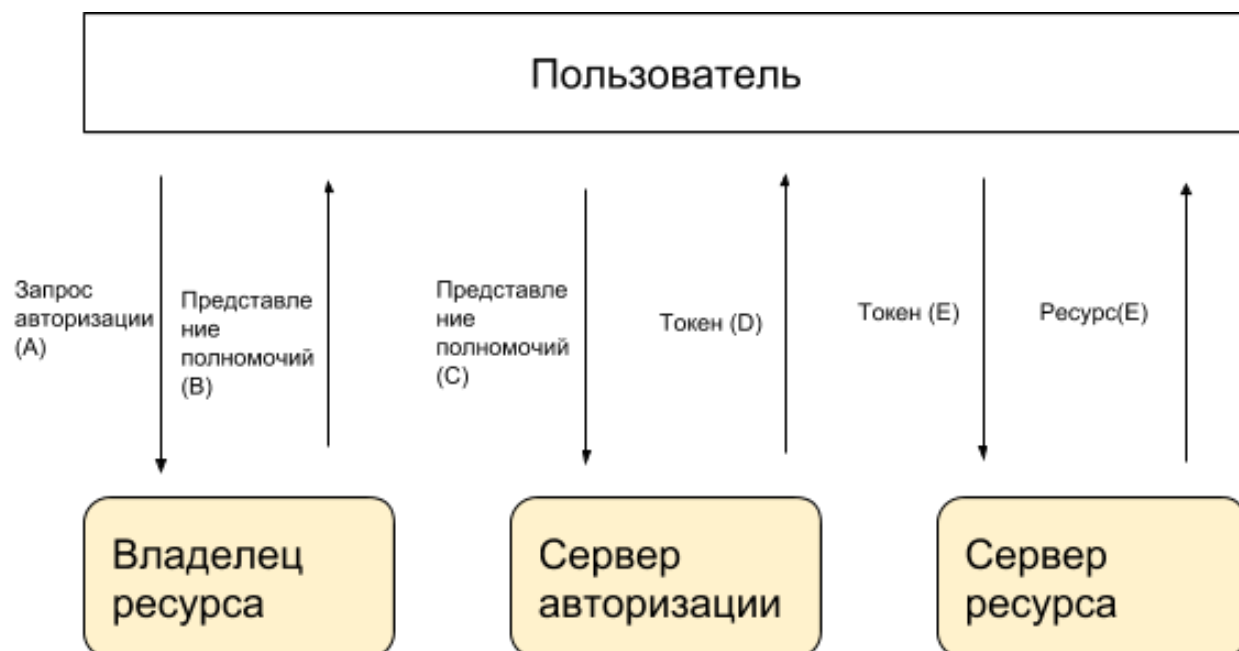


Рис. 1. Упрощенная схема получения доступа к ресурсу

Для получения доступа к ресурсу пользователь должен выполнить последовательно шаги А, В, С, D. Затем пользователь получает токен от сервера авторизации, и, после предъявления токена серверу ресурса в определенный промежуток времени, сервер ресурса может быть полностью удостоверен в том, что пользователь имеет требуемые привилегии, для выдачи запрашиваемого ресурса.

Первое достоинство данной техники в том, что токен остается действующим определенное время, т.е. его можно получить один раз, и использовать без повторного запроса привилегий. Второе достоинство, в свою очередь, дает возможность серверу ресурса, зная ключ для расшифровки подписи токена, заверить подпись и пользоваться открытой частью токена (например, имя пользователя или его идентификационный номер), и не проводить трудоемкие процедуры по получению информации о пользователе из слоя хранения данных приложения.

Описанный выше способ подписи называется HMAC [5]. Смысл его работы достаточно прост: используя секретный ключ, зашифровать сообщение и получившуюся шифрограмму добавить в само сообщение, с возможностью отделения зашифрованной части. На другой стороне, расшифровав и сверив открытое сообщение и расшифрованное сообщение, можно убедиться, что сообщение не изменяли, в случае если обе части будут совпадать. Такой элегантный механизм дает возможность быть уверенными всем представителям системы в том, что сообщение создано в рамках доверенного круга (теми, у кого был ключ). На рисунке 2 это область элементов под меткой E.

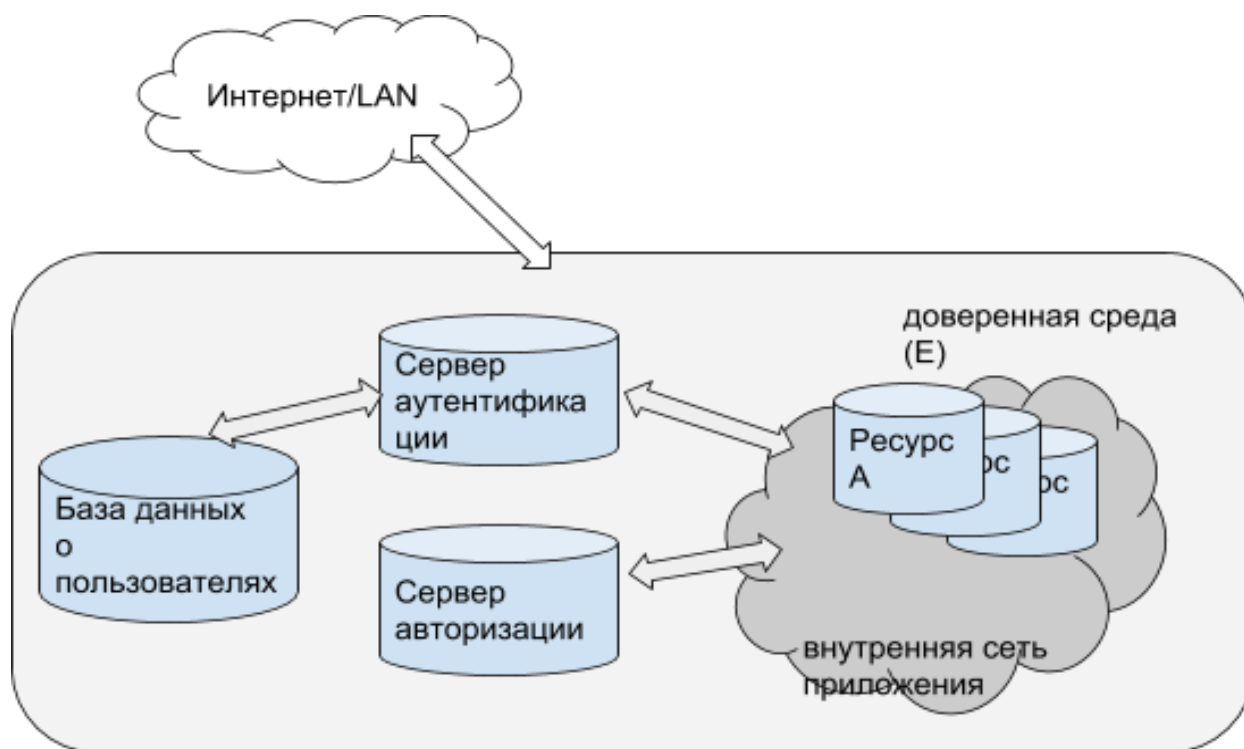


Рис. 2. Структурная схема элементов предлагаемой системы

Для решения проблемы дублирования программного кода и, соответственно, чтобы повысить скорость разработки, можно реализовать унифицированную систему, готовую к повторному использованию и интеграции в различные проекты. Общая структурная схема такой системы приведена на рисунке 2.

Библиографический список

1. Авторизация // Википедия. [2018—2018]. Дата обновления: 22.03.2018. URL: <http://ru.wikipedia.org/?oldid=91652824> (дата обращения: 22.03.2018).
2. Аутентификация // Википедия. [2018—2018]. Дата обновления: 16.01.2018. URL: <http://ru.wikipedia.org/?oldid=90337610> (дата обращения: 16.01.2018).
3. The OAuth 2.0 Authorization Framework // Internet Engineering Task Force. [2012—2012]. Дата обновления: 1.10.2012. URL: <https://tools.ietf.org/html/rfc6749>.
4. OpenID Connect Core 1.0 incorporating errata set 1 // OpenID Foundation. [2014—2014]. Дата обновления: 8.10.2014. URL: http://openid.net/specs/openid-connect-core-1_0.html.
5. HMAC // Википедия. [2018—2018]. Дата обновления: 31.01.2018. URL: <http://ru.wikipedia.org/?oldid=90638310> (дата обращения: 31.01.2018).

УДК 004.056.5

Н.А. Фатеева, магистрант**М.В. Потанина, канд. техн. наук, доцент***Севастопольский государственный университет***ОСНОВНЫЕ АСПЕКТЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ
В МЕДИЦИНСКИХ УЧРЕЖДЕНИЯХ РОССИИ***Аннотация: в данной статье рассматриваются вопросы информационной безопасности медицинских учреждений в России, применяемые методы для защиты сведений в системе здравоохранения.**Ключевые слова: медицина, информационная безопасность, клиника, пациент, доступ, система информации.**Abstract: this article discusses the issues of information security of medical institutions in Russia, the methods used to protect information in the health care system.**Keywords: medicine, information security, clinic, patient, access, information system.*

В настоящее время в России проводится активная информатизация российских учреждений здравоохранения, планомерный переход к комплексной автоматизации медицинской деятельности, слиянию разнообразных медицинских информационных систем в единое информационное пространство, внедрению технологий электронной записи к врачу и единых электронных медицинских карт пациента. В различных учреждениях в медицинских системах могут обрабатываться не только персональные данные пациентов, включая их истории болезни, но и другие данные – например, касающиеся коечного фонда, или результаты лабораторных исследований либо статистические данные, на основе которых проводится анализ работы учреждения.

Медицинские центры, клиники, больницы, а так же прочие учреждения здравоохранения сталкиваются с большим количеством персональных данных, как сотрудников, так и клиентов, обращающихся в данные медицинские организации. Многие документы попадают в категорию врачебной тайны, и в результате этого информационная безопасность в медицине переходит на новый уровень.

Лечебные организации в настоящее время стремятся перейти на электронный документооборот, ведение электронного учета или медицинских карт пациентов. В связи с растущими темпами развития информационных технологий ускорился процесс перехода медучреждений на новый уровень обработки и хранения персональных данных.

Без информационной трансформации повысить эффективность оказания медицинских услуг невозможно. Как и любая отрасль, здравоохранение имеет свои особенности. В сфере информационного обеспечения эти особенности проявляются в следующем[1]:

1. Существуют четкие требования к информационной безопасности баз данных. Это продиктовано тем, что вся обрабатываемая информация принадлежит к первому классу информационных систем, поскольку сведения о состоянии здоровья – одна из самых личных категорий информации. К этому классу относятся данные, разглашение которых может привести к негативным результатам для гражданина и безопасность которых не была обеспечена должным образом.

2. Низкий уровень автоматизации информационного обеспечения государственных медучреждений, которые, в свою очередь, располагают несовременным оборудованием.

3. Обеспечение необходимого доступа к системам информации. Данные меры позволят проконтролировать эксплуатацию систем диагностики, клинического оснащения медицинской организации, закупку и расход медикаментов, соблюдение протоколов лечения.

Базовая информатизация медицинских учреждений совершается в соответствии с Концепцией создания единой государственной информационной системы в сфере здравоохранения (ЕГИСЗ). В пределах данной системы ведутся работы по оснащению медицинских учреждений следующими видами оборудования:

- компьютеры;
- телекоммуникационные системы;

- средства информационной безопасности,
- региональные программы модернизации здравоохранения;
- стандарты информационного обмена в пределах системы;
- федеральных центров обработки данных с общесистемными и прикладными компонентами.

ЕГИСЗ предусматривает создание региональных проектов модернизации электронных баз данных, нормы электронного документооборота между медучреждениями, соответствие стандартам медицинских информационных систем, наличие всероссийского центра обработки информации, обеспечение компьютерной техникой [2].

В системе здравоохранения применяются такие методы для защиты сведений:

- юридические - устанавливают ответственность за нарушение правил использования данных, которые могут воздействовать на возможных нарушителей;
- организационно-управленческие методы устанавливают границы, в которых описаны условия работы ресурсов, функции сотрудников, а также система взаимоотношений между абонентами и администратором;
- технические методы (программное обеспечение) - ограничивают свободный доступ недобросовестных пользователей к информации, устанавливают абонентов, ограничения на доступ и редактирование сведений, обеспечивают криптографическую охрану баз данных.

Защитные механизмы, которые используются для защиты персональных данных, обеспечены необходимыми техническими средствами. Способы автоматизации и обеспечения информационной безопасности в медицинских организациях зависят от размера организации, объема обрабатываемых сведений:

- крупные учреждения имеют собственные центры обработки персональной информации, ответственны за обмен сведениями, синхронизацию систем электронной корреспонденции.
- небольшие организации имеют достаточно возможностей региональных центров сбора, обработки сведений, требования к технической оснащенности и компьютерному обеспечению у таких медучреждений значительно ниже.

Информационное сопровождение данного процесса применяется для реализации управленческой и технологической деятельности. В процессе автоматизации, обеспечения информационной защиты используется разработка и ввод стандартных комплексных программ по защите конфиденциальных сведений. В концепции создания ЕГИСЗ предусмотрено введение юридически значимого электронного документооборота в здравоохранении, которые не требуют необходимости дублирования документов на бумажных носителях, использования электронной цифровой подписи в здравоохранении, обеспечения информационной безопасности при использовании электронных медицинских документов, что потребует соответствующей технической реализации.

Действия, которые могут нанести ущерб информационной системе безопасности, можно разделить на несколько категорий:

- целенаправленная кража или уничтожение данных на рабочей станции или сервере;
- повреждение данных пользователем в результате неосторожных действий.
- несанкционированное проникновение в компьютерные сети;
- DOS-атаки.

Целью несанкционированного проникновения извне в сеть предприятия может быть нанесение вреда (уничтожения данных), кража конфиденциальной информации и использование ее в незаконных целях, использование сетевой инфраструктуры для организации атак на узлы третьих фирм, кража средств со счетов и т. п.

Все данные, которые обрабатываются в медицинских организациях, это сведения, относящиеся к врачебной тайне. Их защита обеспечена на законодательном уровне. Поэтому защита МИС (медицинские информационные системы) представляет собой комплекс мероприятий, в которые входят:

- предоставление комплекса сведений;
- защиту от несанкционированного доступа к системе или данным.

Основным информационным ресурсом, нуждающимся в защите, является конфиденциальная информация о пациенте. В России взаимоотношения врача и пациента регулируются Основами законодательства Российской Федерации «Об охране здоровья граждан» и понятие врачебной тайны там так же присутствует. В частности, в статье 61 указано, что «информация о факте обращения за медицинской помощью, состоянии здоровья гражданина, диагнозе его заболевания и иные сведения, полученные при его обследовании и лечении, составляют врачебную тайну» [4].

Под персональными данными подразумевается следующая информация: фамилия, имя, отчество, год и место рождения, адрес места жительства, сведения о семейном, социальном, имущественном, служебном положении, образовании, профессии, доходах и др.

Согласно закону РФ, оператор имеет право получать от субъекта персональных данных только те сведения, которые необходимы для достижения цели их обработки и только при согласии самого субъекта персональных данных. При этом оператор должен обеспечить конфиденциальность полученной информации. В соответствии с данным документом не допускается обработка специальных категорий персональных данных, которые касаются:

- национальной и расовой принадлежности;
- политических и религиозных взглядов;
- состояния здоровья;
- интимной жизни.

В случае, если указанные данные не будут защищены на необходимом уровне, к ответственности за распространение закрытых сведений привлекут руководство медучреждения.

Техническое обеспечение реализации концепции информационной защиты включает в себя:

- юридически обеспеченное электронное документоведение;
- отсутствие дубликата зафиксированных данных на бумаге;
- заверка электронных документов цифровой подписью;
- обеспечение безопасности информационных баз данных [3].

На рынке информационных систем уже существуют комплекты оборудования, которые разработаны в соответствии с требованиями Министерства здравоохранения и представляющие собой набор сервера-терминала с включенными модулями защищенного ввода и определения абонента. Включенный аппаратный клиент обеспечивает безопасный доступ к образу всей базы данных. Аутентификация абонента производится с помощью ввода личного ключа и секретного кода. Загрузка системы производится исключительно после проверки соответствия введенных паролей доступа.

Терминал производит разграничение доступа отдельных абонентов, обеспечивая аппаратную защиту сведений в медицинской сфере. Администратор обеспечивает опознавание персональных кодов доступа. Это позволяет терминальному ресурсу установить связь с сервером и «опознать» абонента.

Начать работу с удаленным сервером через «тонкий» терминал абонент может после ввода персонального ключа и кода доступа. Таким образом, система идентифицирует клиента и гарантирует защиту информационной базы [5].

Исходя из вышесказанного, можно сделать следующий вывод: информационной безопасности в медицинских учреждениях в настоящее время уделяется все больше внимания. Внедрение медицинской информационной системы позволит более качественно и надежно защищать информацию о клиентах. В тоже время на первый план выходят вопросы, связанные с хранением врачебной тайны. Поэтому каждый работник в отдельности должен соблюдать определенные правила в процессе своей работы с персональными данными.

Для решения проблемы обеспечения информационной безопасности в медицинских учреждениях необходимо применение современных законодательных, организационных и программно-технических мер. Пренебрежение хотя бы одним из аспектов этой проблемы может привести к утрате или утечке медицинской информации, стоимость и роль которой в жизни современного общества приобретает все более важное значение.

Библиографический список

1. Информационная безопасность: Учебное пособие.pdf [Электронный ресурс] – Режим доступа.-URL/[window_catalog/pdf2txt?p_id=18264](#) Дата обращения 20.03.2018
2. Концепция информационной безопасности [Электронный ресурс] – Режим доступа.-URL: <http://www.kaspersky.ru/> / Дата обращения 20.03.2018
3. Особенности решения проблем информационной безопасности в медицинских информационных системах [Электронный ресурс] – Режим доступа.-URL: <http://skif.pereslavl.ru/psi-info/interin/interin-publications/pib2.pdf/> Дата обращения 20.03.2018
4. Федеральный закон «Об обеспечении информационной безопасности» [Электронный ресурс] – Режим доступа.-URL: <http://www.russianlaw.net/law/acts/z38.htm#ch4> Дата обращения 21.03.2018
5. Информационная безопасность медицинских учреждений [Электронный ресурс] – Режим доступа.-URL: <http://www.kkb1.krasu.ru/inform/protect.htm> Дата обращения 21.03.2018

УДК 004.056

Ш.А. Хасанов, магистрант,**Т.Х. Агишев, канд. техн. наук, доц.***Уфимский государственный нефтяной технический университет***БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ СИСТЕМ ПЕРСОНАЛЬНЫХ ДАННЫХ**

Аннотация: В статье рассматриваются вопросы обеспечения безопасности персональных данных, объясняется актуальность данного вопроса, приводится описание типов угроз актуальных для информационных систем персональных данных, порядок определения уровней защищенности информационных систем персональных данных, классы средств защиты информации, используемых для обеспечения необходимого уровня защищенности.

Ключевые слова: информационная безопасность, защита информации, персональные данные, информационные системы персональных данных, уровни защищенности, средства защиты информации.

Annotation: The article deals with the security of personal data, explains the relevance of this issue, describes the types of threats relevant to information systems of personal data, the procedure for determining the security levels of information systems of personal data, classes of information security tools used to provide the necessary level of security.

Key words: information security, personal data, information systems of personal data, levels of security, information security tools.

Персональные данные (ПДн) - любая информация, относящаяся к прямо или косвенно определенному или определяемому физическому лицу (субъекту персональных данных). Обработка персональных данных может проводиться с использованием средств автоматизации и без. Автоматизированная обработка (обработка с помощью средств автоматизации) осуществляется в информационных системах персональных данных (ИСПДн). ИСПДн представляет собой совокупность персональных данных, содержащихся в базах данных, а также информационных технологий и технических средств, позволяющих осуществлять обработку таких персональных данных с использованием средств автоматизации. Обработка персональных данных определяется Федеральным законом от 27.07.2006 № 152-ФЗ «О персональных данных» [1]. К операторам ИСПДн предъявляются требования, изложенные в Приказе ФСТЭК России от 18 февраля 2013 г. № 21 «Об утверждении состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных».

Актуальность рассмотрения вопросов защиты персональных данных объясняется большим количеством утечек. Статистика утечек информации по типам данных за первое полугодие 2017г., представленная компанией Infowatch, показана на рисунке 1.



Рис. 1. Распределение зарегистрированных утечек информации по типам данных за первое полугодие 2017г.

В результате утечек скомпрометировано 7,78 млрд. записей (персональных и платежных данных) — номера социального страхования, реквизиты пластиковых карт, иная критически важная информация. Доля утечек персональных данных составила 65,8% от общего количества утечек. При этом было зафиксировано 20 «мега-утечек». В результате каждой «утекло» более 10 млн. персональных данных. На «мега-утечки» пришлось 98% всех скомпрометированных записей [2]. Эти данные показывают, что мало внимания уделяется вопросам, связанным с информационной безопасностью персональных данных, а если и уделяются, то принимаемые меры являются недостаточно эффективными.

Состав мер по обеспечению безопасности персональных данных зависит от уровня защищенности информационной системы персональных данных. Для информационных систем персональных данных устанавливаются 4 уровня защищенности. Уровень защищенности ИСПДн определяется в соответствии с таблицей 1, при этом необходимо установить категории обрабатываемых персональных данных, вид обработки по форме отношений между субъектами и организацией, количество субъектов, а также тип угроз актуальных для информационной системы. Законодательство устанавливает 4 категории персональных данных: специальные категории ПДн, биометрические категории ПДн, общедоступные категории ПДн, иные категории ПДн. К специальным категориям ПДн относятся данные о национальной и расовой принадлежности, о религиозных, философских, политических убеждениях, информация о состоянии здоровья и интимной жизни субъекта, к биометрическим — данные, характеризующие биологические или физиологические особенности субъекта, к общедоступным — сведения о субъекте, полный и неограниченный доступ к которым предоставлен самим субъектом, к иным — данные, не представленные в 3 предыдущих категориях. По форме отношений между организацией и субъектами ПДн обрабатываться могут ПДн сотрудников оператора и ПДн субъектов, не являющихся сотрудниками оператора. Для информационных систем персональных данных устанавливается 3 типа актуальных угроз. Угрозы 1 типа — угрозы, связанные с наличием недеklarированных возможностей в системном программном обеспечении (ПО), используемом в ИСПДн. Угрозы 2 типа — угрозы, связанные с наличием недеklarированных возможностей в прикладном ПО, используемом в ИСПДн. Угрозы 3 типа — угрозы, не связанные с наличием недеklarированных возможностей в системном и прикладном ПО, используемом в ИСПДн [3].

Что касается технических мер и средств защиты информации (СЗИ), то для обеспечения первого уровня защищенности должны использоваться средства защиты информации не ниже 4 класса, прошедшие проверку не ниже чем по 4 уровню контроля отсутствия недеklarированных возможностей (НДВ), для обеспечения второго уровня защищенности — СЗИ не ниже 5 класса, прошедшие проверку не ниже чем по 4 уровню контроля отсутствия НДВ, для обеспечения третьего и четвертого уровней защищенности — СЗИ не ниже 6 класса.

В случае если для информационных систем третьего уровня защищенности актуальны угрозы 2 типа, должны использоваться СЗИ, прошедшие проверку не ниже чем по 4 уровню контроля отсутствия НДВ. Средства вычислительной техники не ниже 6 класса могут использоваться в информационных системах 4 уровня защищенности, в остальных случаях должны использоваться средства вычислительной техники не ниже 5 класса [4].

При обработке персональных данных в государственных информационных системах (ГИС) к операторам ИСПДн дополнительно предъявляются требования Приказа ФСТЭК России от 11 февраля 2013 г. № 17 «Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах». При этом должно обеспечиваться соответствие уровня защищенности ИСПДн

классу защищенности государственной информационной системы в соответствии с таблицей 2. Класс защищенности ГИС должен быть повышен, если класс защищенности ГИС не обеспечивает необходимый уровень защищенности ИСПДн [5].

Таблица 1 – Уровни защищенности информационных систем персональных данных

Категория обрабатываемых ПДн		Угрозы 1-го типа	Угрозы 2-го типа	Угрозы 3-го типа
	Обрабатываемые субъекты ПДн и количество субъектов;			
Специальные	> 100000 субъектов ПДн	1 УЗ	1 УЗ	2 УЗ
	< 100000 субъектов ПДн	1 УЗ	2 УЗ	3 УЗ
	сотрудники Оператора	1 УЗ	2 УЗ	3 УЗ
Биометрические		1 УЗ	2 УЗ	3 УЗ
Иные	> 100000 субъектов ПДн	1 УЗ	2 УЗ	3 УЗ
	< 100000 субъектов ПДн	1 УЗ	3 УЗ	4 УЗ
	сотрудники Оператора	1 УЗ	3 УЗ	4 УЗ
Общедоступные	> 100000 субъектов ПДн	2 УЗ	2 УЗ	4 УЗ
	< 100000 субъектов ПДн	2 УЗ	3 УЗ	4 УЗ

Таблица 2 – Соответствие уровней защищенности ИСПДн классам защищенности ГИС

Класс защищенности ГИС	Уровень защищенности ИСПДн
К1	1 УЗ
К2	2 УЗ
К3	3 УЗ, 4 УЗ

Необходимость применения тех или иных средств и методов защиты информации определяется моделью угроз безопасности персональных данных. Частная модель угроз составляется на основе Базовой модели угроз и служит для конкретизации действующих угроз. Частная модель угроз определяет перечень актуальных угроз и позволяет существенно снизить расходы на систему защиты информации информационной системы персональных данных. Рекомендации по разработке частной модели угроз содержатся в «Методике определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных». Согласно этому документу, для разработки частной модели угроз необходимо определить исходные данные для построения частной модели угроз, выявить угрозы и их источники, составить перечень актуальных угроз. Актуальность угрозы безопасности ПДн определяется на основе возможности реализации угрозы и показателя опасности угрозы. При этом для каждой угрозы вычисляется возможность реализации угрозы Y по формуле (1):

$$Y = \frac{Y_1 + Y_2}{20}, \quad (1)$$

где:

Y_1 – числовой коэффициент, определяющий исходную степень защищенности;

Y_2 – числовой коэффициент, определяющий вероятность возникновения угрозы;

20 – нормирующий коэффициент.

Коэффициент Y_1 определяется, исходя из набора технических и эксплуатационных характеристик, общих для всех ИСПДн. Коэффициент Y_2 определяется на основе экспертной оценки вероятности возникновения угрозы [6].

Библиографический список

1. О персональных данных: Федеральный закон от 27.07.2006г. №152 -ФЗ // Собрание Законодательства РФ, 2006. - №21.
2. Аналитический центр InfoWatch. Глобальное исследование утечек конфиденциальной информации в I полугодии 2017 года [Электронный ресурс]. Электрон. дан. – Режим доступа: https://www.infowatch.ru/report2017_half. Загл. с экрана.
3. Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных: Постановление Правительства РФ от 01.11.2012г. №1119 // Собрание Законодательства РФ, 2012. - №45
4. Об утверждении состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных: Приказ ФСТЭК от 18.02.2013г. №21 // Российская газета, 2013. - №107
5. Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах: Приказ ФСТЭК от 11.02.2013г. №17 // Российская газета, 2013. – №136
6. «Методика определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных» [Электронный ресурс]. Электрон. дан. – Режим доступа: <https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty/114-spetsialnye-normativnye-dokumenty/380>. Загл. с экрана.

СЕКЦИЯ 9

Системное и прикладное программное обеспечение



УДК 681.3.068

С.Н. Дзюба, магистрант,

С.Н. Фисун, канд. техн. наук., доц.

Севастопольский государственный университет

ИССЛЕДОВАНИЕ МЕТОДОВ АВТОМАТИЗАЦИИ ПОСТРОЕНИЯ НИСХОДЯЩИХ СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ

Аннотация: статья посвящена исследованию методов автоматизации построения нисходящих синтаксических анализаторов, в частности, рассмотрена зависимость длительности синтаксического разбора входной строки синтаксическим анализатором от длины входной цепочки. Оценено влияние особенностей работы виртуальной машины Java на оценки производительности кода. По результатам изучения влияния работы JVM на длительность исполнения кода внесены правки в методику тестирования и получены новые результаты оценок производительности.

Ключевые слова: синтаксический анализ, нисходящий метод LL(1), JVM, виртуальная машина Java, динамическая компиляция, HotSpot, OSR.

Annotation: the article is devoted to the study of methods for automating the construction of descending parsers, in particular, the dependence of the duration of parsing of the input string by a parser on the length of the input chain is considered. The influence of the features of the Java virtual machine on the evaluation of code performance is evaluated. Based on the results of the study of the influence of the JVM on the duration of the code execution, corrections were made to the testing methodology and new results of the manufacturer's performance estimates were obtained.

Key words: parsing, descending algorithm LL(1), JVM, Java virtual machine, dynamic compilation, HotSpot, OSR.

Компиляторы составляют существенную часть программного обеспечения ЭВМ. Это связано с тем, что языки высокого уровня стали основным средством разработки программ. Только очень незначительная часть программного обеспечения, требующая особой эффективности, программируется с помощью ассемблеров. В настоящее время распространено довольно много языков программирования. При разработке компиляторов для языков программирования грамматика исходного языка часто изменяется. Любая корректировка грамматики влечет за собой серьезные последствия, выражающиеся в необходимости изменения управляющей таблицы или программы. Соответственно, чем больше правил содержит язык, тем сложнее и длительнее этот процесс. Синтаксический анализатор является одной из важных частей компилятора.

Рассмотрим пример исследования зависимости времени работы анализатора от длины входной цепочки для следующего фрагмента грамматики учебного языка программирования ASPLE-3:

1. $\langle S \rangle \rightarrow \text{while } v \text{ do } \langle \text{OpSequence} \rangle \text{ end}$
2. $\langle \text{OpSequence} \rangle \rightarrow \langle \text{Op} \rangle | \langle \text{Op} \rangle ; \langle \text{OpSequence} \rangle$
3. $\langle \text{Op} \rangle \rightarrow o | \langle \text{OpSequence} \rangle$

Фрагмент представляет оператор цикла с предусловием. В данной грамматике:

$\langle S \rangle$ - оператор цикла, $\langle \text{OpSequence} \rangle$ - последовательность операторов, $\langle \text{Op} \rangle$ - оператор, V – логическое выражение

Введем следующие сокращения: «W» - «While», «D» - «Do», «Seq» - «OpSequence», «E» - «End».

Сравнительный анализ зависимости времени работы синтаксических анализаторов от длины входной цепочки грамматики учебного языка ASPLE-3 проводился на ноутбуке со следующими характеристиками: процессор- Intel core i5 4200m, ОЗУ- DDR3 1666 МГц объемом 6 Гб, жесткий диск- Western Digital объемом 1 Тб. Входными данными для тестов является последовательность «while v do [while v do o end] end» («while v do end» повторяется от 1 раза до 10 000 раз, каждый раз количество повторов увеличивается вдвое). В ходе исследования зависимости времени синтаксического разбора строки, соответствующей грамматике класса LL(1) ASPLE-3, от ее длины были получены неоднозначные данные, представленные на рисунке 1. Нестабильное поведение графика в интервале [0;20000] символов подвергает сомнению методику измерения времени синтаксического разбора.

Процесс компиляции Java-кода отличается от принципов компиляции C и C++ подобных языков программирования. Динамический компилятор JVM выполняет небольшое количество оптимизирующих преобразований, а оптимизация, проводимая статическими компиляторами для языков C и C++, выполняется во время исполнения программы.



Рис. 1. График зависимости длительности синтаксического разбора от длины входной цепочки

В настоящее время Java код компилируется при помощи исполняющего процесса HotSpot, осуществляющего динамическую компиляцию. Исполняющий процесс HotSpot объединяет интерпретацию, профилирование и динамическую компиляцию. Вместо преобразования всех байткодов в машинный код HotSpot начинает работу как интерпретатор и компилирует только "горячий" код, то есть код, выполняющийся наиболее часто [1]. Во время исполнения он определяет фрагменты кода, выполняющиеся достаточно часто и заслуживающих компиляции. Компиляция часто исполняемого кода позволяет не тратить время на компиляцию редко выполняющегося кода. Поэтому компилятор может тратить больше времени на оптимизацию "горячего" кода. Откладывая компиляцию, компилятор может использовать данные анализа, которые могут быть полезны для лучшей оптимизации.

HotSpot поставляется с двумя компиляторами: клиентским и серверным. Стандартно используется клиентский компилятор. У пользователя есть возможность выбрать серверный компилятор, указав параметр `-server` при запуске JVM. Серверный компилятор оптимизирован для повышения максимальной скорости работы и предназначен для долгосрочных серверных приложений. Серверный компилятор может выполнять значительное число видов оптимизации [1]. Среди них множество стандартных видов оптимизации, присущих статическим компиляторам: удаление проверки диапазона, выведение кода из тела циклов, удаление "мертвого" кода, удаление общих подвыражений, анализ движения данных, развертка циклов. Он также выполняет большое количество оптимизирующих преобразований не свойственных статическим компиляторам (принудительное встраивание вызовов виртуальных методов и др.).

Первые версии HotSpot выполняли компиляцию по одному методу в каждый момент времени. Метод считался "горячим", если он выполнял итерации цикла, превышающие определенное количество раз (10000 в первой версии HotSpot). Однако после компиляции метода переключение на компилированную версию не производилось до тех пор, пока не происходил выход из метода и повторный вход. Поэтому компилированная версия могла использоваться только для последовательных вызовов. В такой ситуации вычисляющий метод может быть скомпилирован, но объектный код ни разу не использоваться [1].

Более современные версии HotSpot используют технологию, называемую on-stack replacement (OSR). Переключение от интерпретируемого кода к скомпилированному (или замены одной версии скомпилированного кода на другую) производится в середине цикла. Для динамических компиляторов без замещения в стеке измерение производительности метода осуществляется следующим образом: выполнить определенное количество вызовов метода, запустить таймер, затем выполнить метод определенное количество раз. Если число тренировочных вызовов превышает порог срабатывания процедуры компиляции метода, фактически измеренные вызовы относятся к скомпилированному коду, а все издержки компиляции должны были проявиться до начала измерения.

В настоящее время динамический компилятор запускается в трудно предсказуемые моменты времени, JVM по своему желанию переключает код с интерпретируемой версии на скомпилированную, а один и тот же код при выполнении может быть скомпилирован и перекомпилирован более одного раза. Если не определить время наступления этих событий, то измерения могут быть серьезно искажены. Для доказательства влияния «прогрева» виртуальной машины Java на длительность обработки входной синтаксическим анализатором входной строки был разработан следующий тестовый пример. На вход программы подается строка, соответствующая грамматике ASPL-3 длиной 2504 терминальных символов или 8514 текстовых символа. Строка представляет собой множественные операторы цикла с предусловием while и не содержит синтаксических ошибок. Программа, осуществляющая устранение левой рекурсии, устранение левой факторизации в исходной грамматике, построение управляющей таблицы, а затем разбор входной строки, перезапускается 500000 раз. Каждый запуск и окончание работы программы фиксируется при помощи стандартной функции библиотеки System Java System.nanoTime(). Формально, метод возвращает наносекунды, но последовательные вызовы метода дают точность до микросекунд. После фиксирования времен начала и конца работы программы вычисляется разница между ними и записывается в массив. По окончании всех итераций синтаксического анализа происходит обработка полученных значений времени. Для снижения статистического шума, все полученные результаты делятся на группы по 500 значений, а далее вычисляется среднее значение на группу. Таким образом, получено 1000 значений длительности времени синтаксического разбора входной цепочки длиной 2504 терминальных символов. Результаты измерений представлены на рисунке 2.



Рис. 2. График длительности синтаксического разбора входной цепочки

Исходя из полученных результатов, нетрудно заметить убывающий характер графика. На первых итерациях видны значительные длительности обработки, которые с ростом количества итераций начинают убывать. После 150000 итераций время начинает колебаться

в пределах 48000 – 70000 нс. Таким образом, очевидно влияние “прогрева” (тренировки) JVM на производительность кода.

Для более глубокого исследования моментов окончания тренировки виртуальной машины, запуска сборщика мусора, прерываний компиляции можно воспользоваться дополнительными параметрами запуска JVM. Например, `-XX:+PrintCompilation` – с ее помощью можно отслеживать моменты запуска компилятора, `-verbose:gc` – с ее помощью можно определить время, затрачиваемое на сборку мусора. Также, можно использовать серверный компилятор исполняющего процесса HotSpot, который производит более агрессивную оптимизацию. Все это помогает снизить различные возможные искажения замеров производительности программного продукта.

Используя полученные данные о влиянии динамической компиляции на время работы программы, была скорректирована методика замера длительности синтаксического разбора строки, длина которой на каждой итерации увеличивается. Новая методика состоит в следующем: длина строки увеличивается за счет добавления в нее вложенных операторов цикла `while` (“`while v do o end`”). На каждой итерации количество вкладываемых циклов внутрь одного внешнего основного цикла увеличивается вдвое от 2 до 32768 раз. Таким образом, длина входной строки увеличивается от 14 терминальных символов до 81924 терминальных символов. Входная строка подается на вход средства автоматизации построения синтаксических анализаторов 500000 раз, из которых для достоверного исключения влияния динамической компиляции 450000 считаются «тренировочными». Длительность синтаксического разбора фиксируется после завершения каждого разбора. Затем для усиленного снижения статистического шума для последних 50000 результатов подсчитывается среднее значение, которое принимается в качестве среднего времени обработки входной строки длиной соответствующей итерации. В результате, получены следующие данные, отраженные на графике на рисунке 3.

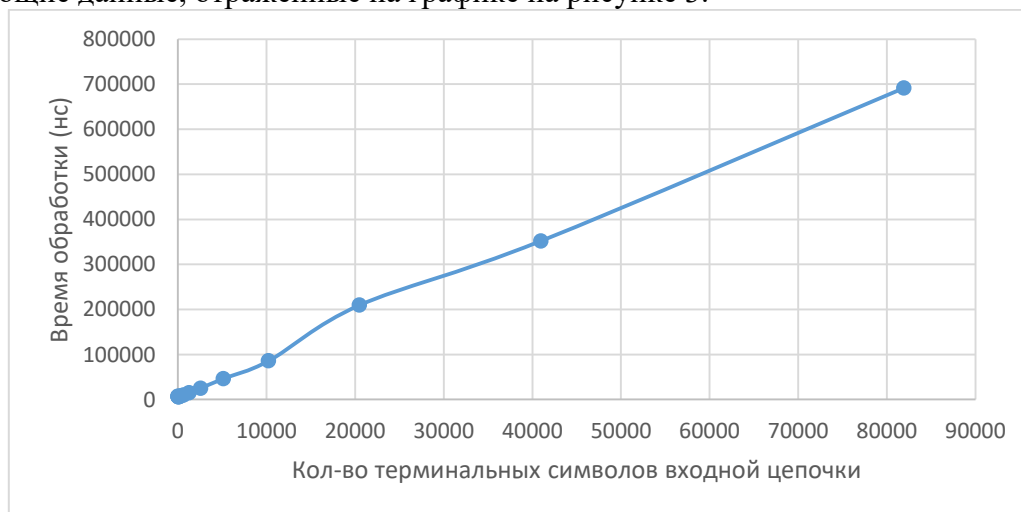


Рисунок 3. Зависимость среднего времени синтаксического разбора от длины входной цепочки

Анализируя полученные результаты и их графическое представление, нетрудно заметить, что теперь на графике нет резких скачков и провалов, график изменяется и возрастает значительно более плавно. Отмечается монотонность возрастания времени синтаксического разбора входной строки в зависимости от ее длины. Наблюдается линейная зависимость между исследуемыми параметрами, характерная для нисходящего синтаксического анализатора LL(1).

Библиографический список

1. В.Goetz Теория и практика Java: Динамическая компиляция и измерение производительности // ibm.ru: статьи по изучению Java. URL: <https://www.ibm.com/developerworks/ru/java/library/j-jtp12214/>

УДК 004.4

Г. Б. Жадан, студент 3-го курса,

Г. А. Левченко, студент 4-го курса,

В. С. Супрунов, студент 3-го курса

И.В. Дымченко, старший преподаватель,

О.А. Сырых, старший преподаватель,

С.А. Кузнецов, старший преподаватель.

*Севастопольский государственный университет***МОБИЛЬНОЕ ПРИЛОЖЕНИЕ «МОБИЛЬНЫЙ СЕВГУ»**

Аннотация: в статье представлены результаты первого этапа проекта «Мобильный СевГУ», в рамках которого разрабатывается многофункциональное мобильное приложение – мобильный информационный инструмент для различных групп пользователей: абитуриентов, студентов и сотрудников Севастопольского государственного университета.

Ключевые слова: мобильное приложение, канбан, паттерн, JSON, класс

Annotation: in the article are presents the results of the first stage of the project "Mobile SevGU". The project develops a multifunctional mobile application - a mobile information tool for various user groups: applicants, students and employees of Sevastopol State University.

Key words: mobile application, kanban, pattern, JSON, class

Каждый ВУЗ, желающий привлечь абитуриентов, стремящийся к повышению известности и своего статуса среди других учебных учреждений, старается идти в ногу со временем и не отставать от трендов. Практически все университеты имеют свои собственные сайты, позволяющие пользователю следить за жизнью ВУЗа, узнавать о событиях и анонсах мероприятий, иметь представления об образовательной и других видах деятельности организации. Часто, наличием web-сайта всё и ограничивается. Однако, увеличивающаяся мобильность пользователей информационного пространства, заставляют образовательные организации реагировать на данный тренд и такие вузы как Московский государственный университет имени М.В.Ломоносова (МГУ), Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики (ИТМО) и Московский Государственный Технический Университет имени Н.Э.Баумана (МГТУ), Сибирский федеральный университет (СФУ) и ряд других уже имеют свои мобильные приложения. В реалиях современной жизни именно мобильные устройства (смартфон, планшет) являются наиболее удобными инструментами для получения информации поскольку практически всегда находятся у человека под рукой за счёт своей компактности. А грамотно реализованные мобильные приложения для таких устройств практически заменяют информационные ресурсы в виде web-сайтов. Таким образом, необходимость разработки такого приложения как «Мобильный СевГУ» достаточно высока.

В настоящий момент завершён этап создания базовой части мобильного приложения, являющейся частью информационного сервиса ФГАОУ ВО «Севастопольский предназначенной для информирования пользователей об Университете, его инфраструктуре, событиях, и образовательной деятельности.

Немаловажным фактором, влияющим на качество разрабатываемого продукта, является организация работы разработчиков. В проекте выбран метод управления разработкой, реализующий принцип «точно в срок» – канбан и принципы бережливого производства [1]. В качестве инструмента для распределения и контроля выполнения задач была выбрана программа для управления проектами – Trello. Несмотря на то, что Trello использует freemium – бизнес-модель, т.е. имеет дополнительный платный функционал, бесплатной части вполне хватит для любой небольшой команды разработчиков.

С помощью Trello производится распределение задач, осуществляется контроль за успеваемостью всех участников процесса разработки. Также данный ресурс хранит загруженные на него документы, необходимые для разработки, что достаточно удобно.

При разработке любого прикладного программного обеспечения, в том числе и мобильных приложений, одним из ключевых аспектов является выбор наиболее подходящей

архитектуры. Обычно, это многокритериальная задача, неверное решение которой скорее всего нанесёт существенный урон как проекту, так и людям, которые над ним работают. Не даром крупные IT-компании выделяют ресурсы для найма отдельных специалистов, занимающихся проектированием системы. Однако при разработке мобильного приложения для операционной системы android задача выбора архитектуры не стоит так остро. За последние годы сообщество android программистов сошлось на мнении, что MVP паттерн проектирования архитектуры является одним из наиболее эффективных инструментов при разработке мобильного приложения. Несмотря на то, что конечному пользователю по сути нет дела до того, что именно находится «под капотом» приложения, MVP паттерн крайне полезен разработчику. Он упрощает понимание логики приложения, и, соответственно, саму разработку. Именно этот паттерн был выбран для приложения «Мобильный СевГУ».

MVP расшифровывается как Model View Presenter. Он предполагает разбиение приложения на три части, что наглядно продемонстрировано на рисунке 1.

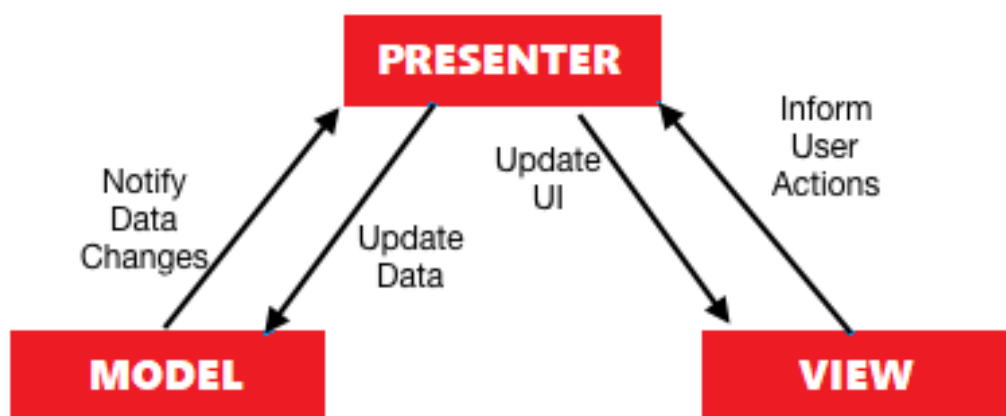


Рис. 1. Принцип работы MVP паттерна

Как видно из рисунка, Presenter устанавливает связь между обработкой данных, получаемых из Model и вызовом методов у View, реализуя тем самым реакцию компонентов пользовательского интерфейса на поступающие изменения. Данный подход является более практичным, нежели его предшественник. До появления паттерна MVP весь код, отвечающий за работу определённого функционала находился в одном файле. Мало того, что это усложняло понимание кода, это ещё и лишало разработчиков возможности использования такого важного инструмента как Unit – тесты. Поскольку логика, которую необходимо протестировать, лежит в обработчике события некоторого графического элемента без его создания тест невозможен. Но в самом тесте его создать нельзя, ведь чаще всего у этого класса нет публичного конструктора. Другими словами, форма создается специальными классами, которых в тестах нет, и которые, в свою очередь, тоже нельзя создать. MVP позволяет прекрасно решить упомянутую проблему. Идея заключается в том, что надо перенести весь код из обработчика в другой класс, который можно будет легко создавать. Этот класс и является Presenter'ом. А в обработчике останется только одна строка, вызывающая метод Presenter'a [2].

В данный момент реализована часть, отвечающая за работу приложения до авторизации пользователя, а также отдельные элементы, предназначенные для абитуриента. Запустив приложение можно увидеть меню (рисунки 2, 3). При активации одного из его пунктов происходит перенаправление на соответствующий раздел приложения.

Для хранения такой сложной сущности как направление подготовки был выбран формат JSON (Java Script Object Notation). Несмотря на происхождение от JavaScript, JSON является независимым от языка и считается эталонным при передаче данных между

клиентским приложением и сервером. Об этом свидетельствует тот факт, что практически для всех языков программирования существуют готовые решения для работы с этим форматом. Не исключением оказалась и Java. На этом языке написано множество библиотек для работы с JSON форматом. Самые известные из них – это Gson, Jackson, JSON.simple и JSONP. Но в выборе одной из них не оказалось необходимости, ведь в Android уже имеются готовые решения по работе с JSON'ом, что лишний раз подчёркивает актуальность его использования [3].

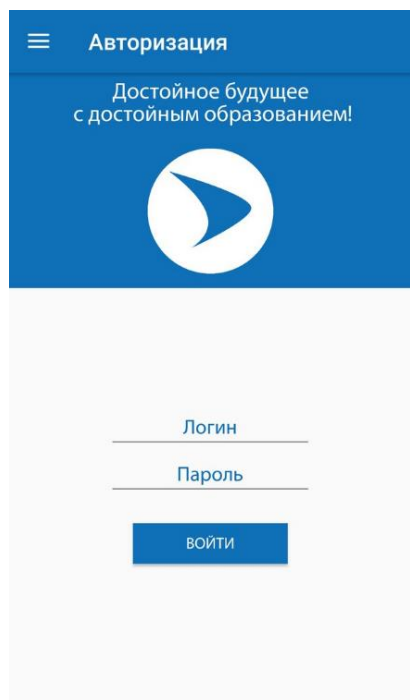


Рисунок 2 – Окно авторизации

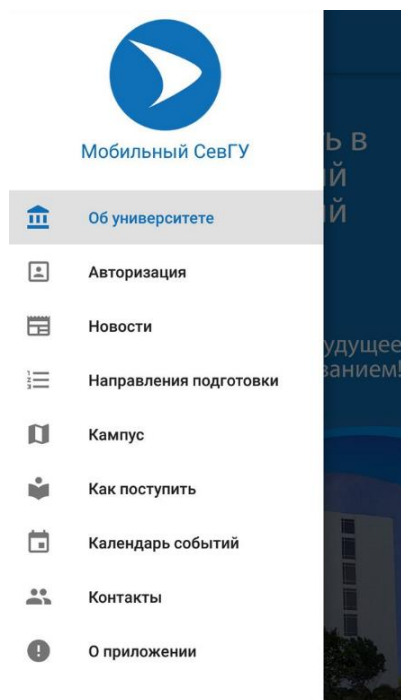


Рисунок 3 – Вид главного меню

Для каждого уровня образования был сформирован отдельный файл расширения JSON, хранящий список всех направлений и информацию о них. Данный файл оформлен таким образом, что даже ничего не смыслящий в программировании человек сможет при необходимости легко модифицировать необходимую информацию. И действительно рано или поздно в этом, без сомнения, возникнет необходимость – изменится название какой-либо специальности, стоимость обучения, да что угодно. И в этом случае не придётся лезть в код и в спешке что-то менять. Достаточно будет изменить одну строчку файла в самом обычном текстовом редакторе. Это, безусловно, является огромным плюсом, ведь подобной реализацией мы делаем наше приложение максимально удобным для дальнейшей поддержки и модификации.

За отображение всей необходимой информации о направлениях подготовки отвечает у показанный на рисунках 4, 5 список. При активации заинтересовавшего пользователя элемента, будет динамически сформирована и выведена вся необходимая ему информация. Особенность реализации данной функции заключается в том, что не все направления имеют одинаковую структуру, которую необходимо вывести на экран. Например, на одной специальности есть заочная форма обучения, на другой её заменяет очно – заочная, а в третьей есть только очная. Абсолютно понятно, что каждый из этих случаев требует свой собственный подход. Для решения этой задачи был разработан класс `ExpandListAdapter`, наследующий класс `BaseExpandableListAdapter` и переопределяющий все необходимые методы своего предка. В одном из переопределённых методов, а именно в `getChildView()`, происходит анализ, определяющий что именно необходимо вывести пользователю. Данный подход позволил динамически формировать необходимую для просмотра информацию,

анализирую данные, находящиеся в JSON файлах. которых в процессе разработки создано пять. Совокупное количество строк около 3000 и тот факт, что вся эта информация приведена к одному виду и обрабатывается по сути с помощью нескольких функций, ещё раз доказывает, что разрабатываемое приложение является долгосрочным проектом, полностью ориентированным на дальнейшую поддержку.



Рисунок 4 – Уровни образования



Рисунок 5 – Список направлений

В приложении также есть раздел новостей (рисунок 6). Календарь событий, изображённый на рисунке 7, позволит упростить организацию деятельности пользователя. Например, абитуриент сможет увидеть, когда в Университете и институтах проводится день открытых дверей и предупрежден об основных событиях приемной компании.



Рисунок 6 – Раздел новостей

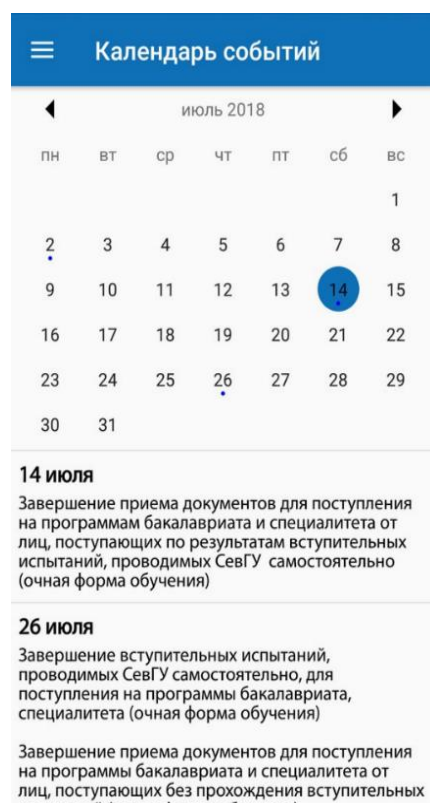


Рисунок 7 – Календарь событий

Раздел «Кампус» позволяет с помощью Яндекс Карт отследить все объекты университета. Для человека, который ни разу не был в Севастополе, данная функция окажется крайне необходимой. Вид данной вкладки отображён на рисунке 8.

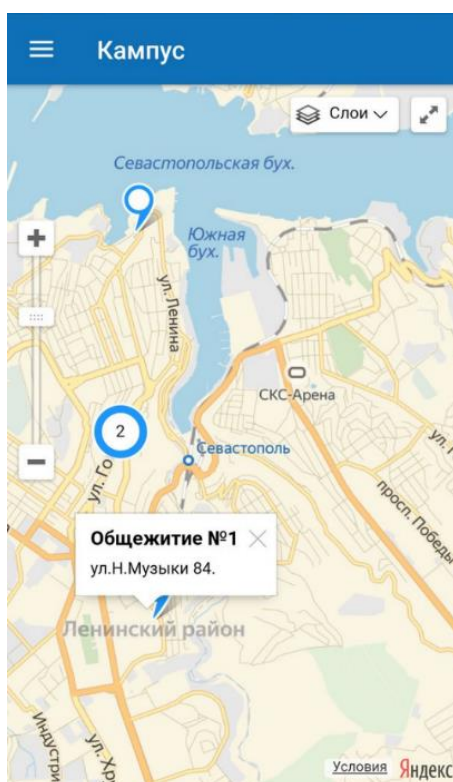


Рисунок 8 – Раздел «Кампус»

В настоящее время мобильное приложение доступно для скачивания с основного сайта СевГУ по адресу <https://www.sevsu.ru/aktivnost/mob>.

Развитие проекта ставит задачи по реализации разделенного доступа для абитуриентов, студентов и сотрудников ВУЗа. Каждая группа пользователей может получить доступ к сервисам, которые необходимы для решения своих задач: абитуриенты получают возможность следить за ходом приемной компании, студенты – доступ к актуальному расписанию, а сотрудники – функционал, зависящий от рода деятельности.

Также планируется доработка нескольких разделов базовой части. Раздел «Новости» пополнится доступом к новостям институтов, данная функция окажется полезна тем, кто хочет следить за событиями, происходящими внутри подразделений, так как каждый из институтов Университета обладает и своей собственной новостной лентой. Раздел «Календарь событий», после доработки основного информационного ресурса СевГУ, усовершенствуется автоматическим обновлением анонсов мероприятий. И, конечно, команда разработчиков, ждет обратной связи от пользователей, с пожеланиями о необходимых функциональных возможностях.

Проект «Мобильный СевГУ» реализуется в рамках проекта Программы развития Севастопольского государственного университета.

Библиографический список

1. Поппендик Мэри. Бережливое производство программного обеспечения: от идеи до прибыли. М.: Вильямс, 2010. – 256 с.
2. Martin Fowler. GUI Architectures. [Электронный ресурс] URL: <https://martinfowler.com/eaaDev/uiArchs.html> (дата обращения 11.03.2018).
3. Официальный сайт Android Developer. Документация. [Электронный ресурс] URL: <https://developer.android.com> (дата обращения 17.03.2018).

УДК 681.3

Д.Ю. Омельченко, студент 3-го курса,

С.Н. Фисун, канд. техн. наук, доцент

Севастопольский государственный университет

ПРОГРАММНАЯ СИСТЕМА ДЛЯ ПРЕОБРАЗОВАНИЯ ГРАММАТИК В НИСХОДЯЩИХ СИНТАКСИЧЕСКИХ АНАЛИЗАТОРАХ

Аннотация: В данной статье рассматривается применение автоматических методов трансляции на основе нисходящих методов. Выявлена основная задача предварительной подготовки входных данных программы в задаче устранения левой рекурсии. Статья подводит некоторые итоги изучения методов устранения леворекурсивных правил их реализации в программных комплексах для дальнейшего применения в построении автономных систем трансляции.

Ключевые слова: трансляция, левая рекурсия, преобразование, грамматика, контекстно свободная, косвенная рекурсия, нисходящий метод.

Annotation: In this article, the application of automatic methods of translation based on top-down methods is considered. The main task of preliminary preparation of the input data of the program in the problem of elimination of the left recursion is revealed. The article sums up some results of studying methods of eliminating left-recursive rules for their implementation in software complexes for further application in the construction of autonomous translation systems.

Key words: translation, left recursion, transformation, grammar, context-free, indirect recursion, top-down method.

Программная реализация методов преобразования грамматик является ключевым этапом построения автоматических систем трансляции естественных языков. Одним из наиболее эффективным для применения в программировании является метод построения трансляторов на базе нисходящего метода. Предиктивные синтаксические анализаторы, т.е. синтаксические анализаторы, работающие методом рекурсивного спуска без возврата могут быть построены для класса грамматик, называющегося LL(1) [1].

Класс LL(1)-грамматик достаточно богат для того, чтобы охватить большинство программных конструкций, хотя при написании грамматики для исходного языка требуется аккуратность. Например, в LL(1)-грамматике не может быть ни левой рекурсии, ни неоднозначности. Особенностью применения данного метода является необходимость предварительного преобразования грамматик языка с целью устранения левой рекурсии и левой факторизации.

Таким образом, для применения нисходящего LL(1)-метода необходимо предварительно выполнить устранение левой рекурсии [1]. Проявление левой рекурсии выражается в повторном обращении нетерминального вызова на себя самого напрямую, либо через некую цепочку выводимых нетерминалов.

Данная задача является алгоритмически разрешимой и делится на два основных типа подзадач: устранение прямой левой рекурсии и устранение косвенной левой рекурсии. Обобщенная схема алгоритма представлена на рисунке 1.

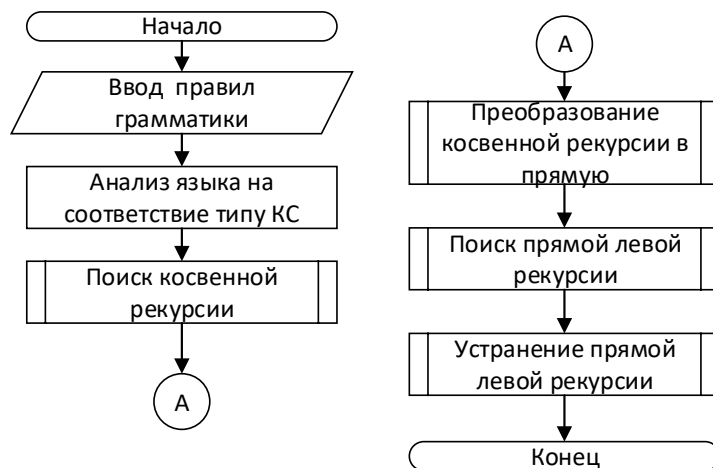


Рис. 1. Обобщенная схема алгоритма леворекурсивных преобразований.

Устранение косвенной левой рекурсии представляет из себя построение списка достижимых нетерминальных символов языка для заданного нетерминала с дальнейшим анализом. Составленный список проверяется на присутствие в нем исходного рассматриваемого нетерминального символа. При обнаружении повторного появления нетерминала в списке допускается существование косвенной рекурсии и строится прямой вывод рекурсивного правила для рассматриваемого нетерминала. Если построение прямого рекурсивного правила является возможным, полученное правило рассматривается как прямая левая рекурсия.

В результате преобразования косвенная рекурсия сводится к задаче устранения прямой левой рекурсии [2]. Прямая левая рекурсия в правиле 1 устраняется с использованием правил 2 и 3. Доказано, что получаемая грамматика - эквивалентна исходной.

$$S \rightarrow Sa_1 | \dots | Sa_n | b_1 | \dots | b_n \quad (1)$$

$$S \rightarrow b_1 S \wedge | b_n S \wedge \quad (2)$$

$$S \wedge \rightarrow a_1 S \wedge | a_n S \wedge | \sim \quad (3)$$

Так как данная задача является основополагающей в формировании автономных систем трансляции на базе нисходящих методов, рассмотрим алгоритм выполнения преобразований по устранению левой рекурсии.

Алгоритм устранения прямой левой рекурсии:

1. Поэтапный просмотр правил на обнаружение леворекурсивного правила. Если правило присутствует перейти на пункт 2, иначе перейти на пункт 8.
2. Для рассматриваемого правила построить два множества:
 - а) Множество А - цепочка выводимых символов содержащая правые части правил непосредственно справа для рассматриваемого нетерминального символа.
 - б) Множество В - цепочка выводимых символов не содержащая рассматриваемый нетерминальный символ в начале правила.
3. Ввести новый нетерминал или к имени рассматриваемого нетерминала добавить «^».
4. Произвести пересечение множества В и исходного нетерминала. Полученные правила в результате пересечения присвоить исходному нетерминалу.
5. Произвести пересечение множества А и нового нетерминального символа.
6. К введенному нетерминалу добавить пустое правило. Полученные правила в результате пересечения и добавления присвоить новому нетерминальному символу.
7. Ввести новые правила в список правил и перейти на пункт 1.
8. Завершить работу алгоритма.

Обобщенная схема алгоритма устранения прямых леворекурсивных правил грамматики представлена на рисунке 2.

Данный алгоритм был реализован на базе средств языка Java и проведен анализ эффективности выполнения преобразований. Исследование осуществлялось на базе Java машины версии 8u161, установленной на ноутбуке со следующими характеристиками: процессор- Intel core i5 7300HQ, ОЗУ- DDR4 2400 МГц объемом 16 Гб, SSD диск- Kingston объемом 256 Гб. Результаты исследования представлены в таблице 1 и на рисунке 3.

Исходя из полученных экспериментальных данных и результатов анализа алгоритма, представленного на рисунке 1, следует, что время выполнения преобразования не зависит от количества нерекурсивных правил. Данное свойство проявляется по причине игнорирования обработки правил не содержащих левую рекурсию.

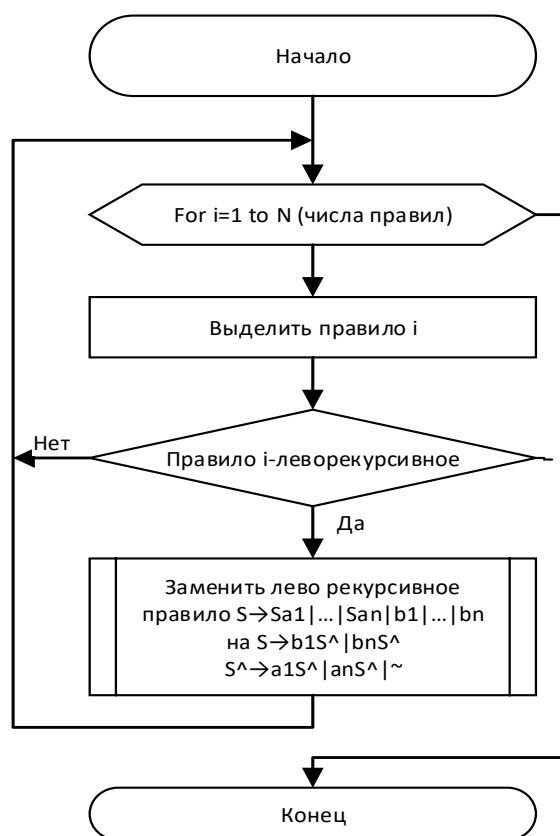


Рис. 2. Обобщенная схема алгоритма устранения прямой левой рекурсии.

Таблица 1. Результаты исследования.

№	Количество правил тестового набора	Количество рекурсивных правил тестового набора	Время выполнения, мкс
1	2	1	52.800
2	4	1	52.801
3	4	2	76.000
4	6	2	76.002
5	6	3	84.500
6	8	3	84.550
7	8	4	88.800
8	10	4	88.900
9	10	5	93.300
10	12	5	94.000

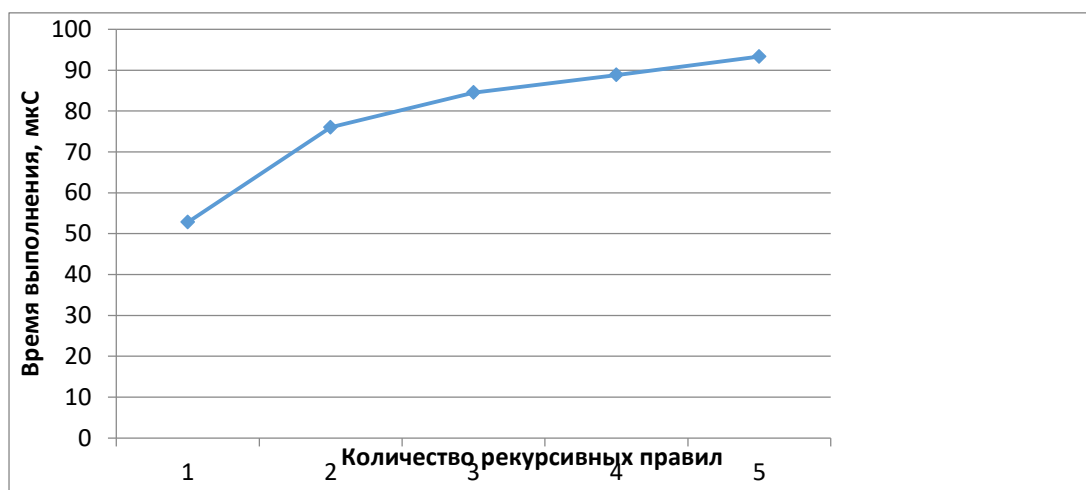


Рисунок 3 – График времени выполнения алгоритма

В настоящее время также разработаны алгоритмы и программы для факторизации КС грамматик и устранения недостижимых нетерминальных символов. Для разработанной программной системы были проанализированы временные затраты на выполнение преобразования КС грамматики в LL(1)-грамматику.

Библиографический список

1. Компиляторы: принципы, технологии и инструменты. / А. Ахо, Р. Сети, Дж.Д. Ульман. - М.: Вильямс, 2001. – 512 с.
2. Системное программное обеспечение: Учебник для вузов. / А.Ю. Молчанов. – СПб.: Питер, 2009. – 396 с.

УДК 004

В.А. Петраков, студент**В.А. Строганов, старший преподаватель***Севастопольский государственный университет***ИСПОЛЬЗОВАНИЕ МЕТОДА ДОПОЛНЕННОЙ РЕАЛЬНОСТИ В КОНТЕКСТЕ ПОВЫШЕНИЯ ПРИВЛЕКАТЕЛЬНОСТИ ГОРОДА БАЛАКЛАВА**

Анотация: Разработано приложение под операционные системы: Android, Windows, Linux, Mac использующее технологию дополненной реальности для отображения планов застройки Балаклавского региона. Разработанное приложение было использовано департаментом экономики г. Севастополь на инвестиционном форуме в г. Сочи. Для разработки были использованы: движок Unity3d, язык программирования C# и плагин Vuforia.

Ключевые слова: дополненная реальность, цифровые технологии, augment reality, Unity3d, C#.

Annotation: The application was developed for operating systems: Android, Windows, Linux, Mac. It's using the augmented reality technology to display the development plans of the Balaklava region. The developed application was used by the Department of Economics of Sevastopol at an investment forum in Sochi. For development, has used the Unity3d engine, the C # programming language, and the Vuforia plugin.

Key words: augmented reality, digital technologies, augment reality, Unity3d, C #

Россия как страна с богатым природным многообразием и культурным наследием, характеризуется уникальностью развития всех без исключения регионов, обладающих мощным историко-культурным потенциалом.

Каждая эпоха оставляет свой след, обнаруживающийся в самобытном наследии исторического развития местности. Своеобразие регионов России связано с сочетанием уникальных объектов материальной и духовной культуры, и представлено мемориальными местами, историческими памятниками, культовыми сооружениями, народными промыслами, национальными традициями и многим другим. То есть, под историко-культурным потенциалом понимается комплексность информации разнообразного характера (природно-географического, исторического, культурного, этно-социального и прочих), подчеркивающей уникальный образ конкретной территории.

Однако, сегодня недостаточно обладать солидным историко-культурным потенциалом, с ним необходимо умело работать (развивать и продвигать), создавая имиджевую привлекательность региона для туристов и инвесторов. В этом процессе незаменимыми становятся новые цифровые технологии, обусловившие трансформацию традиционных способов подачи информации, и породившие новые формы повествования – интерактивные.

Одним из методов интерактивной подачи информации – использование «дополненной реальности» (Augmented reality (AR)), позволяющей интегрировать виртуальные объекты в реальный мир. Именно поэтому в департаменте экономики г. Севастополя решили использовать дополненную реальность в целях повышения привлекательности Балаклавского региона для туристов и привлечения инвестиций.

В связи с этим было предложено разработать приложение, отображающее в дополненной реальности планы по застройке набережной Балаклавы.

В качестве языка программирования был выбран C# из-за его объектной ориентированности и удобства в написании кода и движок разработки интерактивных решений Unity3d. Средой разработки был выбран Visual Studio Community. Для реализации функционала дополненной реальности был выбран плагин Vuforia.

Скачав разработанное приложение, необходимо навести камеру устройства на карту набережной Балаклавы, после чего на ней появятся метки (см. рисунок 1). В качестве карты может выступать как распечатанная на листе картинка, так и обычное фото, открытое на любом устройстве, например, на планшете.

При обнаружении камерой объекта, происходит распознавание найденного и просмотр базы данных для сопоставления найденного маркера (карты) и объектов, которые необходимо разместить поверх карты (метки).



Рис. 1. Главное окно разработанного приложения

Каждая метка – это определенный элемент застройки, при нажатии на метку происходит переход в режим просмотра для более детального ознакомления, как будет выглядеть объект (см. рисунок 2, рисунок 3).

Для отображения объектов застройки изнутри, с помощью программы моделирования 3D max были смоделированы 3D модели. После чего был произведен рендер моделей для получения картинки 360. В дальнейшем весь графический контент был добавлен в проект Unity3d для дальнейшей разработки.



Рис. 2 – Режим просмотра объекта застройки



Рис. 3 – Режим просмотра объекта застройки

Основная идея приложения заключается в распознавании особого маркера – карты Балаклавы. Такую карту достаточно легко распечатать и положить на стенде.

При наведении камеры устройства на распечатанную карту появляются объекты дополненной реальности, которые цепляются к карте и повторяют передвижения карты.

Основная платформа разработанного приложения – смартфоны, работающие на операционной системе Android. Также разработанное приложение можно запустить на Windows, Linux, Mac, но для этого необходима веб-камера.

Разработанное приложение было использовано на российском инвестиционном форуме в качестве интерактивного материала на стенде Севастополя. В дальнейшем предполагается усовершенствование разработки и добавление возможности распознавания объектов реального мира для вывода дополненной реальности поверх макета Балаклавы.

Таким образом, дополненная реальность стала глобальным трендом современности. Использование данного метода в развитии историко-культурного потенциала города становится приоритетным и позволяет осваивать духовный опыт поколений, выступает важным фактором поддержания идентичности, способствует формированию привлекательности региона посредством развития культурного туризма и актуализации историко-культурного наследия.

Библиографический список:

1. Распоряжение Правительства РФ от 17 октября 2015 года № 2073-р. «Об отнесении объектов культурного наследия, расположенных в Крыму, к объектам культурного наследия федерального значения». Режим доступа: <http://government.ru/docs/20140/> (дата обращения: 21.03.2018).
2. Хокинг Джозеф. Unity в действии. Мультиплатформенная разработка на C#, Изд-во Питер, 2016. - 336 с.
3. Торн Алан. Искусство создания сценариев в Unity, Изд-во ДМК Пресс, 2015. - 368 с.

УДК 004.021**Г.А. Тукманов, студент 2-го курса магистратуры***Нижегородский государственный технический университет им. Р.Е. Алексеева,**г. Нижний Новгород.***СЕМАНТИЧЕСКАЯ МОДЕЛЬ ГЕНЕРАЦИИ SQL-ЗАПРОСОВ**

Аннотация: В докладе рассмотрена проблема автоматического построения запросов при работе с базой данных. Представлена гибкая модель построения запроса без привязки к конкретному языку СУБД, с возможностью расширения. Для построения модели применён семантический подход, что делает возможным разработку удобной и расширяемой системы. Обоснована актуальность разрабатываемой системы, приведены преимущества перед другими генераторами SQL запросов.

Ключевые слова: SQL, запросы, автоматизация, семантика, модель, генерация, выборка

Annotation: The report considers the problem of automatic query building when working with a database. A flexible model for constructing a query without binding to a specific DBMS language is presented, with the possibility of extension. To construct the model, a semantic approach is applied, which makes it possible to develop a convenient and extensible system. The urgency of the developed system is substantiated, the advantages over other SQL query generators are given.

Key words: SQL, requests, automation, semantics, model, generation, sampling

В настоящее время ни одна организация, даже маленькая, не обходится без построения базы данных (далее БД), будь то сайт продаж, школа или бизнес-сфера. В каждой отрасли есть информация, которую необходимо структурировать.

Данный доклад сделан с целью обоснования и решения главной проблемы при работе с любой БД. Основные действия любого пользователя с любым скоплением информации – осуществлять поиск и редактировать данные. При работе с базой данных любого размера и любой структуры, человек сталкивается с проблемой построения запросов, которые называют выборками.

Предлагается создать модель построения запросов на основании классификации существующих типов выборок. Её основная цель - максимально упростить поиск нужной информации в любой базе данных. Для решения этой задачи необходимо классифицировать запросы, выявить признаки, различия и сгруппировать их по типу.

Был проведён анализ различных СУБД и выстроенных к ним запросов для выявления наиболее распространённых видов выборок.

Все запросы делятся на две группы:

1) **Запросы-выборки** – это выбор данных из таблиц (одной или нескольких) с заданными параметрами поиска.

2) **Запросы-действия** – это изменение данных в таблицах: удаление, обновление или добавление записей.

При построении модели внимание акцентировано на подробном изучении запросов-выборок. Основываясь на тщательном анализе всевозможных выборок, построена классификация запросов языка SQL в виде графа.

На рисунке 1 изображён связный граф, отображающий проделанную классификацию выборок, построенный с применением семантического подхода.

При построении запроса пользователь поэтапно выбирает области базы данных, из которой хочет получить необходимую информацию, выстраивать запрос для вывода результата, указывает область поиска (таблицы и поля в них), критерии поиска (различные фильтры, конкретные поля). Если необходимо укажет сортировку (будет предоставлен выбор вариантов) и критерии вывода результата (изменить названия полей, таблиц). Всё это – стандартный путь получения информации из любой БД.

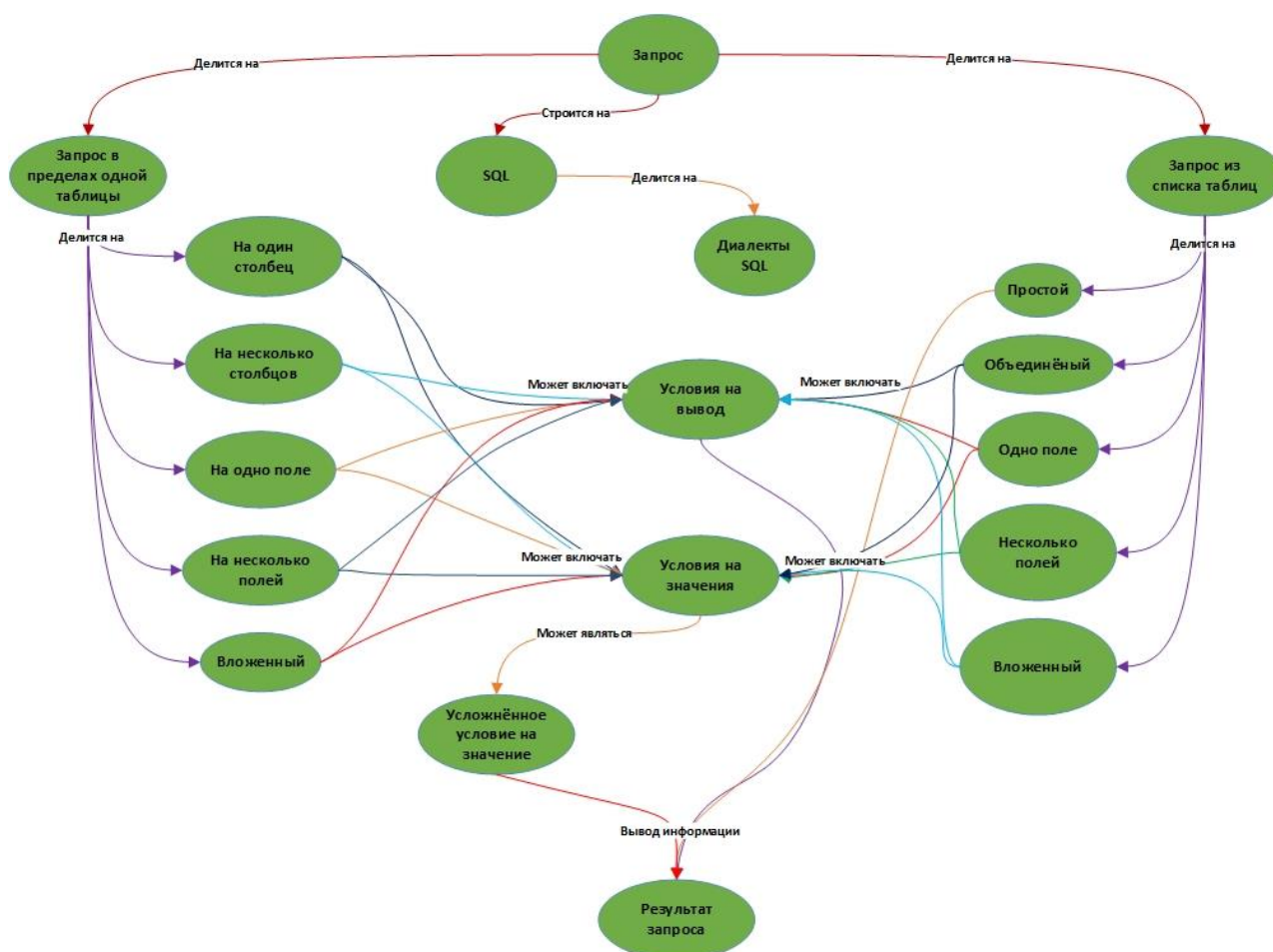


Рис. 1. Граф модели

Гибкость системы

Именно семантика даёт серьёзное преимущество перед другими генераторами запросов, так как появляется чёткая связь между каждым элементом в процессе построения любой выборки. Это даёт возможность безграничного расширения функциональности системы и встраивания новых модулей, независимость от программных платформ и максимальную понятность при работе с БД любого размера.

При таком подходе любой пользователь сможет легко получить необходимую информацию, не задумываясь о сложности своего запроса. Плюс ко всему данная модель делает возможным создавать свои генераторы запросов для любой СУБД. Представленная модель обладает преимуществами перед аналогами и является отличным инструментом генерации SQL кода.

Семантический подход позволяет связать data-логические и инфологические понятия запросов. Процесс создания выборки разбит на шаги, которые представляют собой сущности, связанные друг с другом. После тщательного анализа SQL-запросов был сделан вывод, что можно определить набор шаблонов, состоящий из структур известных запросов. Каждый язык SQL различных СУБД имеет свои особенности, распространённые схемы выборок, которые можно представить в виде различных шаблонов (Рисунок 2). Именно из таких связанных шаблонов и предлагается выстроить систему. Связь блоков в единую систему происходит через интерфейсы.

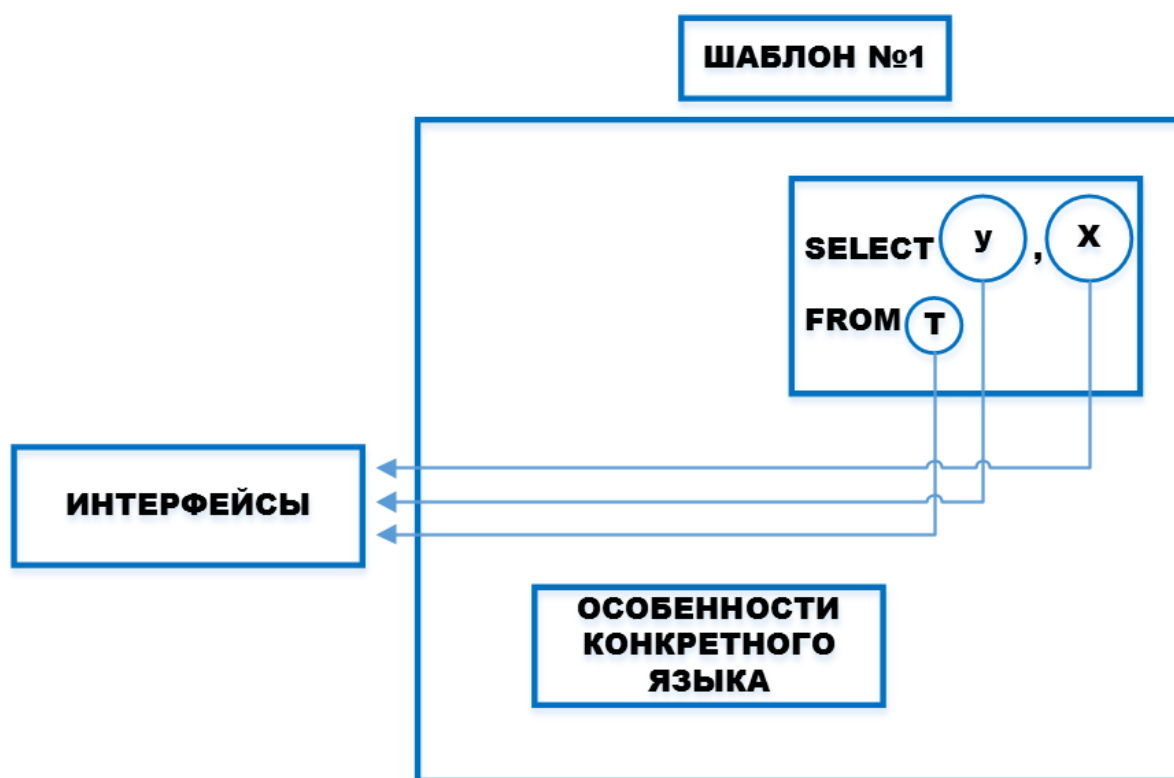


Рис. 2. Модульное представление системы

Построение семантической сети генерации SQL-запроса

Исходя из приведённых выше достоинств семантического подхода, используя проделанную классификацию, была построена семантическая сеть (Рис. 3), которая отображает процесс построения SQL запроса.

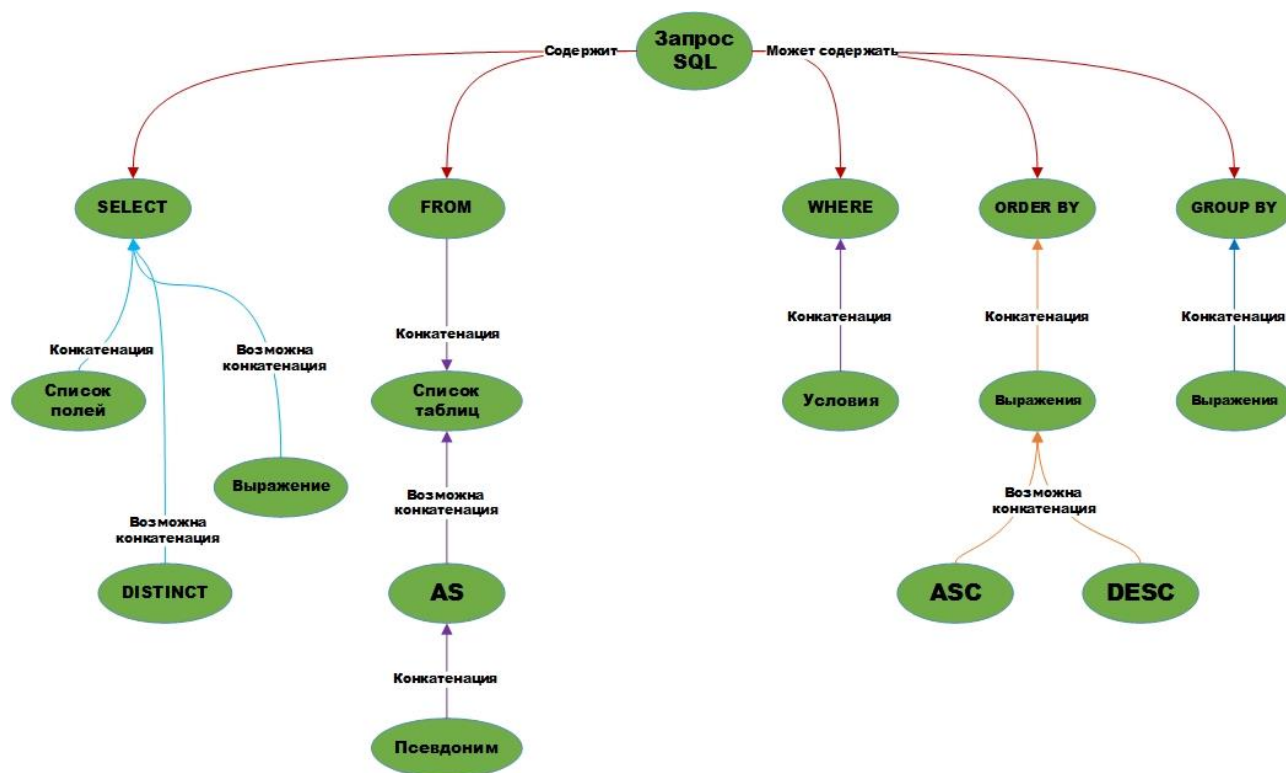


Рис. 3. Семантическая сеть генерации SQL запросов

В модели применено понятие **Конкатенация** — операция склеивания объектов линейной структуры, обычно строк. Например, конкатенация слов «микро» и «мир» даст слово «микромир».

Связь «**Возможная конкатенация**» означает, что пользователь вправе не выбирать связанную с ней сущность.

По семантической сети явно отслеживается, что запрос SQL обязательно должен состоять из команд **SELECT** (соединяется со списком имен полей, из которых будет вестись поиск информации) и **FROM** (соединяется со списком таблиц, из которых будет вестись поиск информации). Команда **SELECT** также может содержать команду **DISTINCT**, исключающую повторы полей. На список полей может быть наложены имена-псевдонимы за счёт команд **AS** + **Псевдоним**.

К необязательным командам относятся:

- **WHERE**. Соединяется с выбранными пользователем условиями.
- **ORDER BY**. Задаёт сортировку + выражение, по которому будет выполнена сортировка. Может соединяться с командами **DESC** и **ASC**, которые задают вид сортировки (по возрастанию и убыванию)
- **GROUP BY**. Задаёт группировку. Соединяется с выражением, задающим группировку

Заключение

Для экономии драгоценного времени человека при запросе информации из любой БД предлагается построить автоматизированную систему генерации SQL запросов. В её основе будет заложена данная семантическая сеть, которая удобно отображает все зависимости элементов системы между собой и основана на шаблонах. Шаблоны построены на основе часто применяемых запросов.

Библиографический список

1. Н. Н. Гринченко, Е. В. Гусев, Н. П. Макаров Проектирование баз данных. СУБД Microsoft Access. Учебное пособие. Москва Горячая линия – Телеком 2004. – 240с.
2. Alan Beaulieu Learning SQL. ИМБО Санкт-Петербург – Москва 2007. – 310с.
3. Грофф Дж. Р., Вайнберг П.Н., Оппель Э. Дж. SQL. Полное руководство. 3-е изд. : Пер. с англ. - М.: ООО "И.Д. Вильямс", 2015. - 960 с.

УДК 004.912

Д.Д. Шевченко, магистрант

В.А. Дубенецкий, канд. техн. наук, доцент

*Санкт-Петербургский государственный электротехнический университет СПбГЭТУ «ЛЭТИ»***ПРОГРАММНАЯ СИСТЕМА ДЛЯ ПОИСКА НЕЧЕТКИХ ДУБЛИКАТОВ В ТЕКСТОВЫХ ДОКУМЕНТАХ**

Аннотация: В статье рассмотрены базовые процессы анализа текстовой информации на предмет наличия недобросовестных заимствований. Предложена структура программной системы для поиска нечетких дубликатов в текстовых документах. Рассмотрен каркас программной системы, реализованный на платформе 1С «Предприятие» 8.3 и использующий алгоритмы: шинглов, I-Match, опорных слов.

Ключевые слова: нечеткий дубликат текста, шингл, мегашингл, супершингл, хеш-функция, I-Match, метод опорных слов.

Annotation: This article considers the basic processes of textual information analyzing aimed to presence unfair borrowing. The structure of the software system for searching the fuzzy duplicates in text documents is proposed. The framework of the software system implemented on the platform 1C "Enterprise" 8.3 is considered and uses next algorithms: Shingles, I-Match, keywords method.

Keywords: fuzzy duplicate text, shingles, megashingles, super-shingle, hash function, I-Match, keywords method.

Введение

В связи с все возрастающими объемами структурированной и неструктурированной информации, развитием электронных журналов, проведением дистанционных курсов и онлайн научно-практических конференций большое значение приобрели проблемы защиты авторских прав в связи с монетарными и репутационными рисками, а также ростом избыточности информации, проявляющемся в распространении плагиата авторских текстов. На сегодняшний день, когда информационные технологии развиваются особенно интенсивно, интеллектуальная собственность приобретает большую значимость. В связи с существенным увеличением объемов данного вида собственности возникает необходимость в мощных автоматизированных инновационных инструментах для защиты авторских прав, инспектирования и нахождения плагиата в текстовой, графической и других видах информации.

Данная работа является продолжением исследований, проводимых в работе [1], где приведен сравнительный анализ методов обнаружения плагиата в текстовых документах.

В соответствии с [2], плагиат в текстовой информации можно условно разделить на следующие виды: дословное копирование – полный перенос авторской информации без указания источника копирования; мозаичное копирование (компиляция текстовой информации) – произвольное объединение текстовых фрагментов из разных источников без авторской ссылки; «рерайтинг» текстов – перестановка слов в тексте, замена слов на синонимы; некорректное цитирование – включение в текстовые материалы оформленной цитаты (в кавычках), но без ссылки на первоисточник.

Поисковые алгоритмы, как правило, эффективно выявляют первые два вида плагиата. При исследовании остальных видов плагиата применим термин «нечеткий дубликат». Согласно [3] «нечеткий дубликат – это документ, в котором основной смысл и стилистика первоисточника сохранена и при этом нет полного дублирования. Нечеткие дубликаты позволяют предположить, являются ли два объекта частично одинаковыми или нет» [4].

Приоритетной подзадачей в рамках анализа текстовой информации на предмет наличия заимствований является поиск. Методы поиска могут быть классифицированы различными способами [5]. В большинстве случаев под поиском понимают задачу выборки, в которой требуется найти элемент, удовлетворяющий некоторым условиям [6]. Однако в рамках задачи анализа нечетких дубликатов возникают ситуации, когда поиск представляет собой нахождение объектов, похожих, в определенном смысле, на заданный объект. В таких случаях принято говорить об использовании алгоритмов нечеткого поиска. Во многом

возникновение задач нечеткого поиска обусловлено проблемами управления качеством данных, в том числе связанными с наличием как в запросах, так и непосредственно в базах данных, ошибок ввода информации [7].

1. Структура и функциональное назначение основных модулей программной системы поиска нечётких дубликатов

Определим последовательность действий при анализе текстовой информации на предмет наличия заимствований, которая является концептуальным базисом для выделения модулей в разрабатываемой программной системе:

1. Канонизация текста
2. Выделение ключевых слов текста
3. Формирование списка аналогов для дальнейшего сравнения
4. Канонизация текстов-аналогов
5. Попарное сравнение исходного текста с текстами-аналогами
6. Формирование и представление результатов сравнения

На основании описанной последовательности действий при анализе текстовой информации был сформирован ряд функциональных блоков для программной системы поиска нечетких дубликатов в текстовых документах. Структурный состав основных модулей программной системы представлен на рисунке 1.

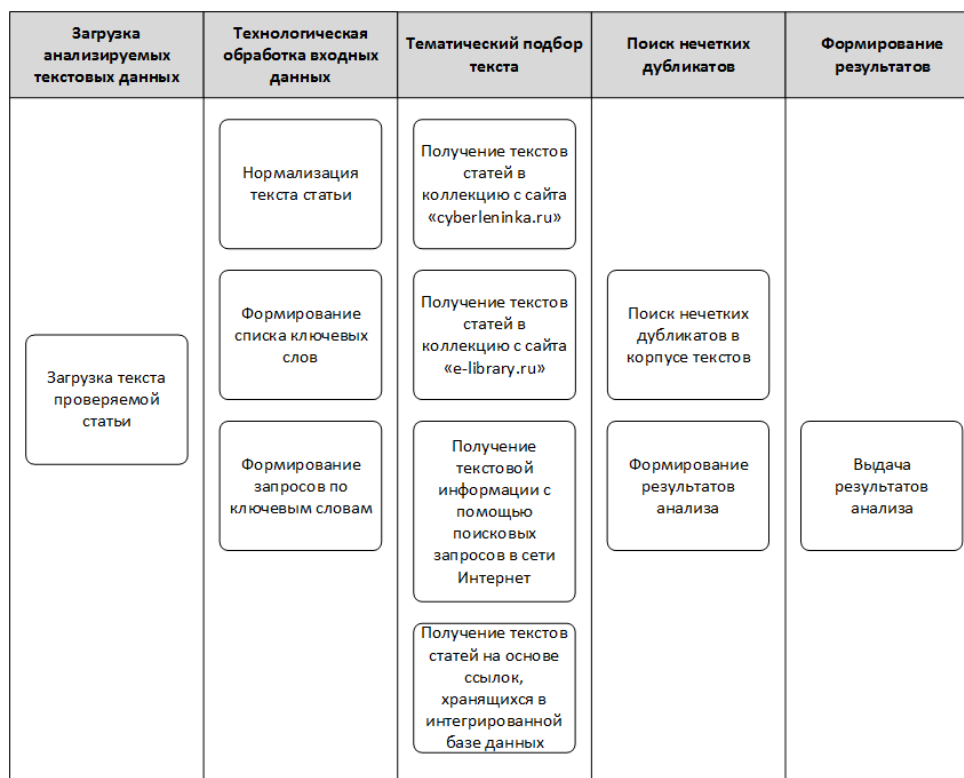


Рис.1. Структурный состав подсистем в программной системе

Функциональное назначение основных модулей:

1. Загрузка проверяемой статьи. Данный модуль позволяет произвести загрузку текстовой информации из файлов формата .doc, .docx, .rtf, .txt, .pdf в систему для последующего анализа.

2. Нормализация текста статьи. Данный модуль преобразует исходный документ во внутреннее представление текстовой информации. Процесс нормализации включает следующие функции: нахождение основы слова для заданного исходного слова (стемминг);

очистку текста от предлогов, союзов, знаков препинания, HTML-тегов, прочих символов; перевод всех символов к нижнему регистру. Блок-схема алгоритма нормализации текста приведена на рисунке 2.

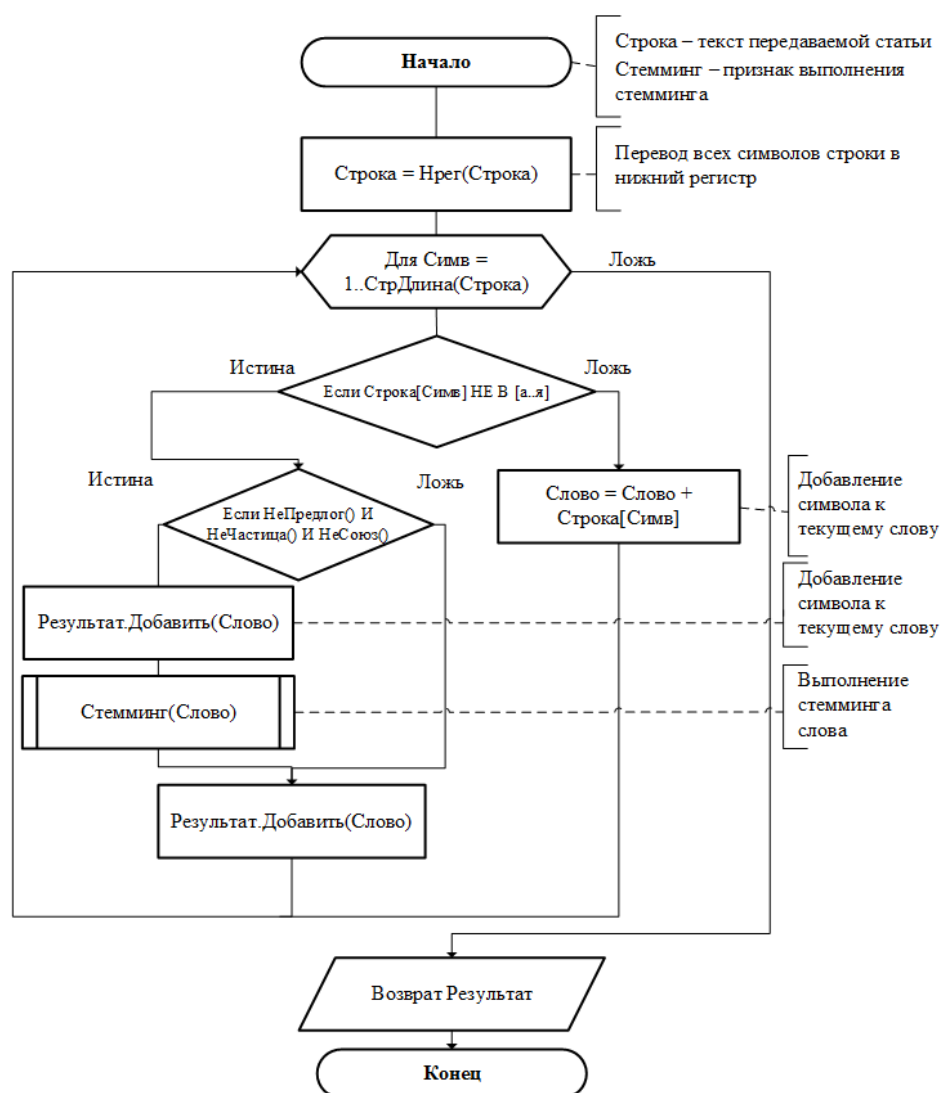


Рис.2. Блок-схема алгоритма нормализации текста

3. Формирование списка ключевых слов. Данный модуль на основе нормальной формы исходного документа при помощи статистических алгоритмов (мера TF) производит формирование списка ключевых слов.

4. Формирование запросов по ключевым словам. Данный модуль производит формирование поисковых запросов для подбора тематических статей на основе сформированного ранее списка ключевых слов.

5. Получение текстов статей с «cyberleninka.ru». Данный модуль осуществляет загрузку текстов статей с открытого информационного ресурса «cyberleninka.ru».

6. Получение текстов статей с «elibrary.ru». Данный модуль осуществляет загрузку текстов статей с открытого информационного ресурса «elibrary.ru».

7. Получение текстовой информации с помощью поисковых запросов в сети Интернет. Данный модуль осуществляет подбор текстовой информации и отбор по тематике в соответствии с заданными ключевыми словами.

8. Получение текстов статей на основе ссылок, хранящихся в информационной базе. Данный модуль осуществляет подбор статей из списка хранящихся в интегрированной базе

на основе списка ключевых слов. Таким образом, после выполнения модулей 3-8 формируется коллекция текстовых документов для последующего анализа.

9. Поиск нечетких дубликатов в корпусах текстов. Данный модуль осуществляет поиск нечетких дубликатов одним из предложенных методов: метод шинглов (супершинглов или мегашинглов), I-Match, метод опорных слов.

10. Формирование результатов анализа. Данный модуль формирует результирующий документ (экранную форму), в котором визуальным (цветовыми зонами) выделены элементы заимствований в тексте.

11. Выдача результатов анализа. Данный модуль позволяет отобразить результаты анализа исходного текста методами поиска нечетких дубликатов на предмет заимствования в экранной форме, и сохранить результаты анализа в текстовый документ.

2. Интерфейс программной системы поиска нечётких дубликатов

Для реализации диалогового взаимодействия пользователя с программной системой, был разработан интерфейс программной системы, содержащий в себе все необходимые объекты для реализации тонкой настройки механизма анализа текстовой информации и работы с результатами проверки.

Общий вид главной формы приложения приведен на рисунке 3. Программная система реализована на платформе 1С, что позволит интегрировать ее в приложение 1С:Университет.

Рис.3. Основная форма приложения

Элементы основной формы приложения разделены на три основных блока:

- блок пользовательских настроек;
- блок ввода данных;
- блок анализа результатов.

Помимо основной формы приложения была создана форма для добавления элементов в интегрированную базу данных.

Рис.4. Настройки метода шинглов

Блок пользовательских настроек позволяет пользователю вносить изменения в параметры анализа текста, тем самым варьируя степень углубленности анализа текстов и уровень быстродействия программной системы. На рисунке 4 представлена вкладка настройки параметров анализа методом шинглов. На вкладку вынесены такие параметры, как минимальный процент совпадения, определяющий нижнюю границу совпадения при сравнении шинглов текстов, длина шингла – параметр, определяющий длину единицы сравнения текстов, метод сравнения – будет ли производиться полное сравнение текстов обычным методом шинглов, или же краткое на основе мегашинглов, а также цвет выделения заимствований, идентифицированных с помощью метода шинглов.

Рис.5. Настройки метода I-Match

На рисунке 5 отображена вкладка изменения параметров метода I-Match. Для данного метода определены три степени глубины обхода – полнотекстовый обход статьи, обход по абзацам и по предложениям. В зависимости от выбранного метода обхода построение L-множеств и расчет сигнатур будет выполняться для всего текста, отдельных абзацев или предложений соответственно. Данный параметр позволяет пользователю выбрать требуемую ему степень детализации результатов анализа, а также сократить затрачиваемое время.

Рис.6. Настройка метода опорных слов

На рисунке 6 отображена вкладка изменения параметров для метода опорных слов. Для данного метода реализована возможность ввода порогового значения отбора, которое является нижней границей частот для слов, формирующих сигнатуру документа.

Панель выбора методов анализа (рисунок 7) позволяет пользователю выбрать один или несколько применяемых методов для анализа текста, а также использование стемминга при нормализации текста. В случае, если ни один из методов не будет выбран, анализ будет проведен с применением метода шинглов, с использованием метода сравнения мегашинглами, длиной шингла в 3 слова.

Проверять шинглами: ☒ Проверять I-Match: ☒ Проверять опорными словами: ☒ Использовать стемминг: ☐

Рис.7. Панель выбора методов анализа

Блок проверяемого текста (рисунок 8) используется для ввода исходного текста, по окончании анализа фрагменты текста в блоке выделяются цветом, соответствующим методу, которым было обнаружено заимствование. Так же, при наведении на фрагмент заимствования отображается наименование источника заимствования. При нажатии на фрагмент заимствования в блоке «Текст источника» будет отображен фрагмент статьи, из которой производилось заимствование.

Текст статьи:.

В лингвистике термин текст используется в широком значении, включая и образцы устной речи. восприятие текста изучается в рамках лингвистики текста и психолингвистики. так, например, и. р. гальперин определяет текст следующим образом: «это письменное сообщение, объективированное в виде письменного документа, состоящее из ряда высказываний, Займствование из: Текст1ыми типами лексической, грамматической и логической связи, имеющее определённый моральный характер, прагматическую установку и соответственно литературно обработанное»

Рис.8. Блок ввода проверяемого текста

Блок источников заимствования (рисунок 9) реализует навигацию по статьям, являющимся аналогами проверяемого текста. Для перехода между источниками используется выпадающий список в верхней части блока.

Источники заимствования: Текст1

Текст источника: Текст2

В лингвистике терм...
Восприятие текста...
определяет текст следующим образом: «это письменное сообщение,

Рис.9. Блок источников заимствования

Панель вывода результатов анализа (рисунок 10) содержит процентное содержания заимствований в тексте проверяемой статьи, выявленное используемыми методами анализа.

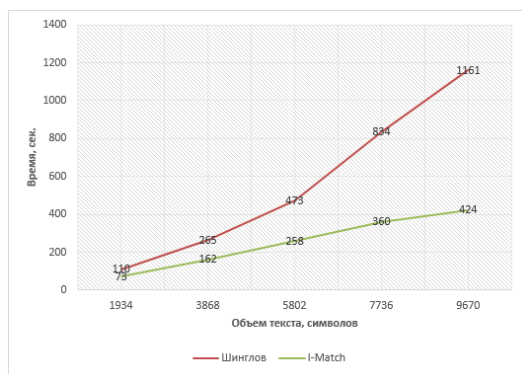
Сравнение I-Match: 0,00 Сравнение шинглами: 0,00 Сравнение опорным словами: 0,00

Общий результат: 0,00

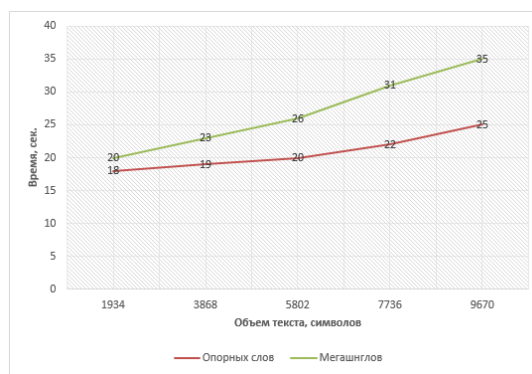
Рис.10. Панель вывода результатов анализа

3. Исследование зависимости скорости обработки данных по алгоритмам от объемов анализируемого текста

Проведено исследование зависимости скорости обработки данных от объемов анализируемого текста. Результаты исследований приведены в таблице 1 и на рисунке 11. Время в таблице 1 приведено в секундах, объемы данных в символах.



А)



Б)

Рис.11. Исследование зависимости скорости обработки данных от объемов данных методами: А) Шинглов и I-Match; Б) Опорных слов и Мегашинглов

Таблица 1 – Исследование зависимости скорости обработки данных различными методами от объемов данных

Метод	1934	3868	5802	7736	9670
Шинглов	110	265	473	834	1161
Мегашинглов	20	23	26	31	35
I-Match	73	162	258	360	424
Опорных слов	18	19	20	22	25

Исследования показали, что методы Шинглов и I-Match дают более точные результаты, но при этом время обработки данных существенно зависит от объемов обрабатываемого текста. Методы «опорных слов» и Мегашинглов дают хорошие результаты при существенно меньших затратах времени на обработку. Таким образом, для быстрой проверки текстов на наличие заимствований рекомендуется использовать методы «опорных слов» и Мегашинглов, при необходимости глубокой проверки следует подключать методы Шинглов и I-Match.

Заключение

Перспективами дальнейшего развития рассмотренной программной системы являются возможности добавления дополнительных алгоритмов поиска нечетких дубликатов, оптимизация существующих методов за счет рефакторинга программного кода. Так же система может быть расширена функциональными модулями анализа текста, представленного на иностранных языках.

Исследовательская задача может получить дальнейшее развитие в направлении проверки эффективности использования различных хеш-функций и метрик оценки близости текстов с точки зрения времени обработки текстовых документов.

Библиографический список:

1. Шевченко Д.Д., Машенко Е.Н. Исследование актуальных методов обнаружения плагиата в текстовых документах //Сборник статей студенческой научно-технической конференции «В мире компьютерных технологий», г. Севастополь, 4-8 апреля 2016г. – Севастополь: Изд-во СевГУ, 2016.– С.88-91
2. Плагиат [Электронный ресурс]// Научно-техническая библиотека Томского Политехнического университета – Электрон. дан. – URL: <http://www.lib.tpu.ru/plagiatus.html> (дата обращения: 05.03.2016).
3. Дубликат [Электронный ресурс] — Режим доступа: [http:// ru.wikipedia.org/wiki/Дубликат](http://ru.wikipedia.org/wiki/Дубликат) — Загл. с экрана. (дата обращения: 16.03.2013).
4. Квашина Ю. А. Методы поиска дубликатов скомпонованных текстов научной стилистики // ТАПП . 2013. №1 (11). URL: <http://cyberleninka.ru/article/n/metody-poiska-dublikatov-skomponovannyh-tekstov-nauchnoy-stilistiki> (дата обращения: 01.03.2016).
5. Кнут Д.Э. Искусство программирования. Сортировка и поиск. 2-е изд. – М.: Вильямс, 2007. – Т. 3. – 824 с.
6. Макконел Дж. Основы современных алгоритмов. 2-е изд. – М.: Техносфера, 2004. – 368 с.
7. Карахтанов Д.С. Использование алгоритмов нечеткого поиска при решении задач обработки массивов данных в интересах кредитных организаций // Аудит и финансовый анализ. – 2010. – № 2. URL: www.auditfin.com/2010/2/toc.asp (дата обращения: 15.05.2012).

УДК 681.3

А.Г. Шлемко студент 4-го курса

С.Н. Фисун, канд. техн. наук, доцент

Севастопольский государственный университет

ИССЛЕДОВАНИЕ МЕТОДОВ АВТОМАТИЗАЦИИ ПОСТРОЕНИЯ ЛЕКСИЧЕСКИХ И СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ

Аннотация: В данной статье проводится исследование быстродействия лексического и синтаксического анализаторов при разработке компиляторов для ЭВМ. Рассматриваются методы построения блоков лексического и синтаксического анализатора. Основное внимание уделяется автоматной модели при разработке лексического анализатора и методу нисходящего синтаксического анализа LL(1). Цель данной работы: сравнить эффективность построения и работоспособности анализаторов, реализованных разными способами.

Ключевые слова: лексический анализатор, управляющий автомат, генератор лексических анализаторов Lex, синтаксический анализатор, метод LL(1), рекурсивный спуск, исследование, сравнение.

Annotation: In this article, the speed of lexical and syntactic analyzers is investigated in the development of compilers for computers. Methods for constructing blocks of a lexical and syntactic analyzer are considered. The focus is on the automaton model in the development of the lexical analyzer and the LL (1) downlink parsing method. The purpose of this work: a comparison of the efficiency of construction and the performance of analyzers, implemented in different ways.

Keywords: lexical analyzer, control automaton, lexical analyzer generator Lex, syntactic analyzer, LL (1) method, recursive descent, research, comparison.

Лексический анализатор – программа, которая считывает исходный текст программы и распознает в нём лексемы заданного языка. В результате формируется перечень всех найденных лексем в исходной программе. Главная проблема, которая встречается при разработке лексического анализатора, является то, что невозможно однозначно определить границы лексем, поскольку в тексте исходной программы лексемы никак не ограничены.

В общем случае задача лексического анализа заключается в следующем:

- Определение границ лексем, которые в тексте исходной программы явно не указаны;
- Выполнение действий для сохранения информации об обнаруженной лексеме;
- Вывод сообщения об ошибке, если лексема неверна.

С теоретической точки зрения все функции лексического анализатора могут выполняться на этапе синтаксического анализа. Однако существует несколько причин, по которым лексический анализатор включают практически во все компиляторы:

- Сокращение объёма информации, обрабатываемой на этапе синтаксического анализа;
- Некоторые задачи на этапе лексического анализа могут быть решены более простыми методами, чем на этапе синтаксического анализа;
- Возможность обработки текста программы параллельно с его подготовкой;
- Отделение синтаксического анализатора от работы непосредственно с текстом исходной программы.

В общем случае можно выделить 3 основных метода реализации лексического анализатора.

1. Использование генератора лексических анализаторов, такого как Lex, для создания лексического анализатора по спецификациям, основанным на регулярных выражениях. В этом случае генератор предоставляет функции для чтения и буферизации ввода.

2. Написание лексического анализатора на подходящем языке программирования с использованием возможностей ввода-вывода этого языка для чтения входной информации.

3. Написание лексического анализатора на языке ассемблера и явное управление процессом чтения входной информации.

Первый способ самый легко реализуемый из всех перечисленных, так как порядок определения лексем мы задаем при помощи регулярных выражений и специального языка Lex, который достаточно просто понять. Однако диаграмма переходов получается избыточной, а регулярными выражениями не всегда просто определить возможные наборы лексем языка, что в итоге дает нам небольшую эффективность блока лексического анализатора.

Третий способ, наоборот, дает нам самый эффективный лексический анализатор, но процесс создания требует отличного знания ассемблера. Стоит отметить, что из-за отсутствия наглядного интерфейса страдает восприятие пользователя в процессе отладки.

Исходя из всего выше сказанного, был выбран второй способ, который является «золотой серединой» в контексте данного исследования, а в качестве языка высокого уровня был выбран Java.

В большинстве компиляторов лексический и синтаксический анализаторы – это взаимосвязанные части. Возможны два принципиально различных метода взаимосвязи лексического и синтаксического анализа:

- Последовательный;
- Параллельный.

Синтаксический анализатор (синтаксический разбор) — это часть компилятора, которая отвечает за выявление основных синтаксических конструкций входного языка. В задачу синтаксического анализа входит: найти и выделить основные синтаксические конструкции в тексте входной программы, установить тип и проверить правильность каждой синтаксической конструкции и, наконец, представить синтаксические конструкции в виде, удобном для дальнейшей генерации текста результирующей программы. Можно выделить два типа алгоритмов синтаксического анализа:

1) Нисходящий разбор - правила грамматики раскрываются, начиная со стартового символа, до получения требуемой последовательности лексем. Наибольшее распространение получили метод рекурсивного спуска и LL(1).

2) Восходящий разбор - правила восстанавливаются из правых частей, начиная с лексем и кончая стартовым символом. Наибольшее распространение получили методы LR(1) и предшествования [1, 2].

Существует также метод рекурсивного спуска, однако он предполагает полную реализацию конечного для конкретной грамматики. Из этого следует, что такой метод нельзя отнести к автоматизированному процессу построения конечных автоматов, а входные таблицы в общем случае строятся вручную. Преимуществом данного метода является то, что если имеем дело с константной грамматикой, то полученный автомат будет нагруженным и, как следствие, более производительным. Плата за скорость выполнения – отсутствие возможности модификации исходной грамматики (придется разрабатывать новый автомат).

Метод рекурсивного спуска реализуется следующим образом: для каждого нетерминала грамматики создается своя процедура, носящая его имя; ее задача - начиная с указанного места исходной цепочки найти подцепочку, которая выводится из этого нетерминала. Если такую подцепочку считать не удастся, то процедура завершает свою работу вызовом процедуры обработки ошибки, которая выдает сообщение о том, что цепочка не принадлежит языку, и останавливает разбор. Если подцепочку удалось найти, то работа процедуры считается нормально завершенной и осуществляется возврат в точку вызова. Тело каждой такой процедуры пишется непосредственно по правилам вывода соответствующего нетерминала: для правой части каждого правила осуществляется поиск подцепочки, выводимой из этой правой части. При этом терминалы распознаются самой процедурой, а нетерминалы соответствуют вызовам процедур, носящих их имена.

Такой подход не предполагает построение управляющей таблицы для исходной грамматики, что является достаточно трудоемкой задачей в общих случаях. Также очевидно, что трудоемкость разрабатываемого таким образом автомата напрямую зависит от размерности грамматики. Данный метод может оказаться проще при разработке и в теории полученный автомат должен обрабатывать входные цепочки быстрее, чем LL(1) анализатор. Однако стоит отметить, что такой подход не универсальный, ведь для каждой новой грамматики необходимо разрабатывать отдельный модуль.

Существует другой подход: разрабатывается модуль реализующий автомат с магазинной памятью, а исходными данными для первого являются грамматика и

управляющая таблица. Такой модуль в общих случаях считается универсальным, а построение управляющей таблицы – алгоритмически разрешимая задача. На входную грамматику накладываются ограничения LL-грамматики: в множестве правил не должно присутствовать прямой левой рекурсии и левой факторизации. Устранение левой факторизации алгоритмически невозможно.

Возможны случаи объединения лексического и синтаксического анализаторов при проектировании компилятора, так как спецификации лексем можно внести при помощи регулярных выражений внутри описываемой грамматики языка, но есть несколько серьезных доводов в пользу отделения лексического анализа от синтаксического:

- Значительная часть времени компиляции тратится на сканирование литер. Следовательно, выделение позволяет сконцентрироваться на сокращении времени.
- Синтаксис символов можно описать в рамках очень простых грамматик. Если отделить сканирование от синтаксического распознавания, то можно разработать эффективную технику разбора, которая наилучшим образом учитывает особенности таких грамматик.
- Развитие языков высокого уровня требует четкого понимания как лексических, так и синтаксических свойств языка. Разделение этих двух свойств позволит нам исследовать их независимо друг от друга.

Теперь подойдем к сути данной работы: для одной и той же входной цепочки заданной грамматики проверим быстроедействие двух вариантов автоматов.

Исходная грамматика: LL-грамматика PL-02;

Длина исходной цепочки: примерно 90 символов;

Исходная цепочка:

```
BU=V; IYTWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDWYDU=VE$
```

Для грамматики PL-02 таблицы будут выглядеть следующим образом (см. рисунки 1 и 2).

```

<оператор>
<идентификатор>
<выражение>
<условие>
CALL
BEGIN
END
IF
THEN
;

<op>→ u = v | call u | begin <cop> end | if y then <op> |
while y do <op>
<cop>→ <cop> ; <op> | <op>

```

	U	=	V	C	B	E	I	Y	T	W	D	;	\$
<OP>	1	E	E	2	3	6	4	E	E	5	E	E	6
<COP>	7	E	E	7	7	7	7	E	E	7	E	7	E
<X>	E	E	E	E	E	9	E	E	E	E	E	8	E

Рис. 1. Пример управляющей таблицы синтаксического анализатора

1	<OP>:U=V
2	<OP>:CU
3	<OP>:B<COP>E
4	<OP>:IYT<OP>
5	<OP>:WYD<OP>
6	<OP>:e
7	<COP>:<OP><X>
8	<X>:;<OP><X>
9	<X>:e

Рис. 2. Пример списка правил синтаксического анализатора

Для исходных данных повторим опыт 10 раз для каждого из анализаторов, а затем сравним полученные замеры времени выполнения обработки входной цепочки. Измерение будем проводить в наносекундах.

Таблица 1. Результаты анализа входной цепочки

№ опыта	Рекурсивный спуск, нс	LL(1), нс	dX, %
1	0,43	8,55	95
2	0,68	1,76	61
3	0,70	1,17	40
4	0,67	0,99	32
5	0,74	1,09	32
6	0,65	1,37	53
7	0,75	0,83	10
8	0,55	1,48	63
9	0,84	0,88	5
10	0,71	1,01	30

Из последнего столбца таблицы 1 видно, что быстродействие метода LL(1) меньше рекурсивного спуска. Разница варьируется в пределах от 5% до 95%. Среднее значение показателя: 42%, однако данная величина может измениться при увеличении количества повторения эксперимента. Стоит отметить, что результаты, полученные на других ЭВМ, могут отличаться в зависимости от производительности самой ЭВМ, а также состояния ее нагруженности. Однако в целом, метод LL(1) в любом случае будет более трудоемким, чем рекурсивный спуск. Это можно объяснить тем, что при построении выходного стека происходит постоянная обработка списка правил и управляющей таблицы, а также формирование рабочего стека, который в процессе обработки может существенно превышать размер выходного стека. В то же время автомат с рекурсивным спуском всего лишь вызывает заранее определенные для данной грамматики методы в правильном порядке, что исключает работу с дополнительными структурами данных, но делает автомат жестко привязанным к конкретной грамматике.

Библиографический список

1. Ахо А. Компиляторы: принципы, технологии и инструменты/ А. Ахо, Р. Сети, Д. Ульман – Пер. с англ. М.: Вильямс, 2003. – 768 с.
2. Волкова И.А. Формальные языки и грамматики. Элементы теории трансляции./ И.А. Волкова, Т.В. Руденко – М.: Диалог-МГУ, 1999. – 62с.



Сборник статей всероссийской студенческой научно-технической конференции
Издано в авторской редакции

Ответственность за содержание статей несут авторы

Ответственный за выпуск: О.В.Ченгарь

Для создания электронного издания использовано:
Microsoft Word 2010, Adobe Acrobat Pro
Подписано к использованию 6.04.2018 г.
Уч.-изд. л. 20,91. Объем данных 7,6 Мб

ФГАОУ ВО «СевГУ», кафедра ИТиКС
299053, Севастополь, ул. Университетская, 33.
Тел.: (8692) 43-50-08
E-mail: itiks@sevsu.ru
