

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и  
информационных технологий

РАЗРАБОТКА КАРТОГРАФИЧЕСКОГО ЭЛЕМЕНТА ОТОБРАЖЕНИЯ  
НОВОСТНЫХ ДАННЫХ  
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 421 группы  
направления 09.03.01 — Информатика и вычислительная техника  
факультета КНиИТ  
Горшковой Екатерины Алексеевны

Научный руководитель

доцент, к. ф.-м. н., доцент

\_\_\_\_\_

Л. Б. Тяпаев

Заведующий кафедрой

к. ф.-м. н., доцент

\_\_\_\_\_

Л. Б. Тяпаев

Саратов 2017

## ВВЕДЕНИЕ

**Актуальность темы.** В последние годы области применения геоинформационных технологий постоянно расширяются. Ежедневно огромным количеством людей осуществляется поиск организации и её контактных данных, маршрута движения на автомобиле, пешком или на общественном транспорте и т.п.

За счёт развития телекоммуникационных технологий, и, вследствие этого, роста пропускной способности сетей и вычислительной мощности ЭВМ, постоянно увеличивается количество пользователей сетей. В результате этого геоинформационные технологии стремительно и успешно развиваются в Интернет, где они существуют в виде картографических сервисов, с помощью которых можно лишь просмотреть географические данные через сеть и выполнить несложные манипуляции с ними, геопорталов, предоставляющих сервисы для визуализации, редактирования и анализа геоданных и картографических веб-приложений, позволяющих создавать интерфейсы нужной сложности и связывать их с базами данных. С их помощью пользователи Интернет получают возможность не только взаимодействовать с уже существующими картами, но и добавлять к ним свой функционал, будь то элементы управления, географические объекты или плагины, упрощающие работу с картами.

Актуальность определила выбор **цели** данной работы: построение графического элемента интерфейса для новостного сервиса, позволяющего интерактивно выводить новости в виде географической информации и осуществлять фильтрацию вывода по историческим данным и по типам данных.

**Объект исследования** — Картографический веб-сервер Leaflet.

**Предмет исследования** — Основные возможности картографического веб-сервера Leaflet, позволяющие отображать географическую информацию на новостном сервисе.

Для достижения поставленной цели в работе необходимо решить следующие **задачи**:

1. Изучить основы web-картографии.

2. Рассмотреть наиболее распространённые web-картографические сервисы, описать их возможности, провести сравнительный обзор и выбрать один из них для выполнения цели работы.
3. Создать класс на языке программирования JavaScript, позволяющий добавлять картографическую подложку на веб-страницу, загружать и удалять с карты географические данные в виде маркеров.
4. Выполнить кластеризацию маркеров.
5. Реализовать поиск геопозиции пользователя с помощью инструментов Browser Object Model.
6. Осуществить возможность добавления географического маркера на карту пользователем.
7. Реализовать взаимодействие карты с временной шкалой и фильтрами по типам данных.

#### **Структура работы:**

Данная работа состоит из двух глав:

Глава 1 — Основы Web-картографии.

Глава 2 — Построение картографического элемента интерфейса с использованием JavaScript библиотеки Leaflet.

**Практическая значимость** — создан графический элемент интерфейса для новостного сервиса обладающий возможностью географического отображения информации и её фильтрации.

## Основное содержание работы

Выпускная квалификационная работа содержит две главы:

Первая глава **Основы Web-картографии** начинается с рассмотрения понятия web-картографии и её задач, видов картографических web-сервисов.

Было выяснено, что картографические web-сервисы бывают динамическими и кэшируемыми.

В случае динамических web-сервисов всякий раз при перемещении или масштабировании происходит обращение к данным и перерисовка изображения с учетом нового охвата. При этом пользователь получает только готовые растровые картинки — их рендеринг осуществляется сервером.

В кэшированных («тайловых») сервисах мозаика растровых изображений (тайлов) для каждого масштаба готова заранее, поэтому работа выполняется намного быстрее.

Обязательным и всегда доступным режимом отображения данных в тайловом сервисе является режим «Схема», в составе которого отображается картографическая подложка.

При наличии дополнительных данных к тайловому сервису могут быть подключены режимы «Спутник», «Планы» и т.п.

Помимо тайлов, на картах присутствуют слои, которые представляют собой логическую совокупность географических данных, могут быть разнообразными для различных карт и бывают нескольких типов:

1. слои географических объектов - точки, линии и полигоны;
2. чертежи;
3. элементы управления картой и слоями;
4. виртуально любые данные, у которых есть местоположения (погода, глубина, условия дорожного движения и т.д.).

Далее были рассмотрены ключевые организации, регулирующие деятельность разработчиков по созданию веб-картографических приложений, основные картографические стандарты и современные механизмы для создания веб-картографических приложений. Выяснилось, что последние делятся на:

- Виртуальные глобусы — простые и эффективные средства быстрого создания и публикации данных в Internet. Эта категория инструментов харак-

теризуется массовым распространением и быстрой доставкой данных пользователям.

- Пользовательские ГИС — большая и сложная категория, тесно связанная с web-картографией. Как правило, пользовательские ГИС, с одной стороны, играют роль клиентов, работающих с данными поставляемыми картографическими web-серверами, а с другой, в них осуществляется массовая подготовка и анализ данных перед публикацией их в web.

- Картографические web-серверы — целое семейство продуктов свободного характера, предназначенных для быстрой публикации пользовательских данных в web. Главное отличие подобных систем от виртуальных глобусов — полный контроль над программным обеспечением и самими данными, однако взамен приходится расплачиваться большей сложностью установки и настройки, часто требующей знаний языков программирования и основ администрирования.

Далее был проведён сравнительный обзор картографических веб-серверов, представленный на рисунке 1.

Для выполнения задач данной дипломной работы была выбрана JavaScript библиотека Leaflet, так как она обладает всем нужным функционалом, занимает мало места на диске, хорошо поддерживает мобильные устройства, имеет открытый исходный код и не ограничивается в использовании.

Название web-сервера	Размер библиотеки/приложения	Доступ к исходному коду	Поддержка мобильных платформ	Использование тайловых сервисов	Совместимость с браузерами	Возможность коммерческого использования
MapServer	2,48 Мб	Есть	Есть	OpenStreetMap, Спутник, Google Maps, Яндекс Карты и т.д.	Яндекс браузер, Google Chrome, Opera, Mozilla Firefox, IE, Android Browser	Есть
GeoServer	48 Мб	Есть	Есть	OpenStreetMap, Спутник, Google Maps, Яндекс Карты и т.д.	Яндекс браузер, Google Chrome, Opera, Mozilla Firefox, IE, Android Browser, Safari	Есть
Mapstraction	1,37 Мб	Есть	Нет	OpenStreetMap, Спутник, Google Maps, Яндекс Карты и т.д.	Яндекс браузер, Google Chrome, Opera, Mozilla Firefox, IE	Есть
API Яндекс Карт	Подключается только через html тег <script>	Нет	Есть	Только Яндекс Карты	Яндекс браузер, Google Chrome, Opera 12, Android Browser 4.0+, Mozilla Firefox, Safari, IE 8-10	Есть, но для определённого использования необходима платная лицензия
Google Maps API	Подключается только через html тег <script>	Нет	Есть	Только Google Maps	Яндекс браузер, Internet Explorer 10 и 11, Mozilla Firefox, Google Chrome, Safari, Chrome на Android 4.1+,	Есть, но для определённого использования необходима платная лицензия
OpenLayers	423 Кб	Есть	Есть	OpenStreetMap, Спутник, Google Maps, Яндекс Карты и т.д.	Яндекс браузер, Google Chrome, Mozilla Firefox, Safari, IE 9-, Android 4	Есть
Leaflet	123 Кб	Есть	Есть	OpenStreetMap, Спутник, Google Maps, Яндекс Карты и т.д.	Яндекс браузер, Google Chrome, Mozilla Firefox, Safari 5+, Opera 12+, IE 7-11, Safari for IOS 7+, Android browser 2.2+	Есть

Рисунок 1: Сравнение картографических web-серверов

Вторая глава **Построение картографического элемента интерфейса с использованием JavaScript библиотеки Leaflet** включает в себя описание взаимодействия с JavaScript классом ViewMap.

Добавление картографической подложки было выполнено созданием экземпляра класса ViewMap и вызовом его метода addMap. В качестве аргументов данный метод принимает:

- шаблон для загрузки тайлов в виде ссылки;
- атрибут в виде html кода, содержащий информацию об отображаемой картографической подложке, лицензиях и т.д.;
- координаты центра карты при её инициализации;
- начальный масштаб карты;
- масштаб приближения при перемещении карты на координаты положения пользователя;
- параметр, наличие которого определяет необходимость проверки геопозиции.

Определение геопозиции пользователя происходит с помощью средства ВОР (Browser Object Model) — объекта navigator. Если браузер пользователя поддерживает геолокацию, то карта автоматически перемещается на местоположение данного пользователя.

Для города Саратова было выполнено центрирование геолокации — если координаты положения пользователя расположены в пределах граничных координат города, то состояние карты устанавливается так, чтобы в пределах карты помещался весь город (рисунок 2).



Рисунок 2: Определение геопозиции жителя города Саратова

Далее были инициализированы маркеры и кластеры.

Кластер — это маркер, в который объединяются другие маркеры, расположенные на заданном друг от друга расстоянии. Кластеризация выполнена с помощью функционала библиотеки `markercluster.js`.

Также маркерам было добавлено свойство вывода балуна с информацией при нажатии (рисунок 3).

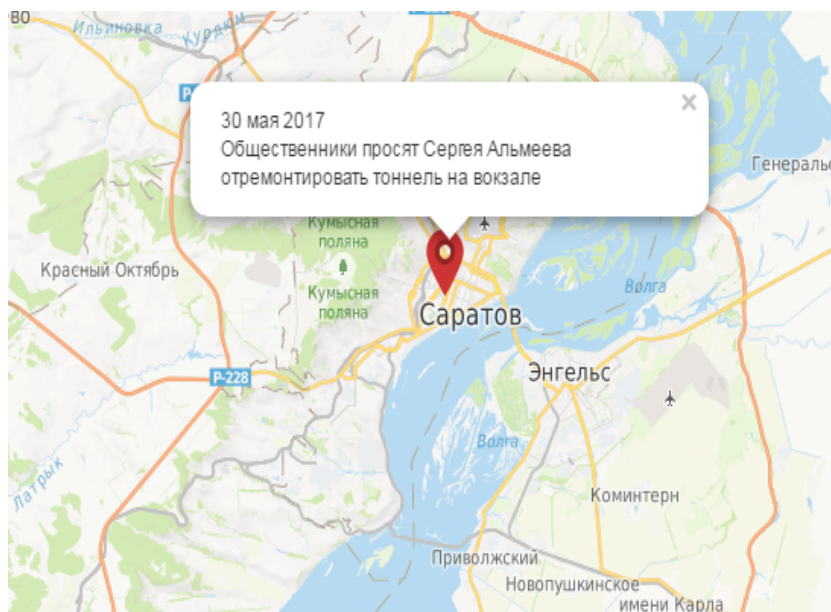


Рисунок 3: Добавление новости во всплывающем балуне

После этого была реализована возможность пользователем добавлять и удалять собственные маркеры по нажатию мыши на картографическую подложку. Для этого были созданы методы `onMapClick`, `addMarker` и `deleteMarker` класса `ViewMap`.

Также было описано, как загрузка, обновление и удаление географических данных, получаемых в виде массива, реализованы в методах `draw`, `redraw` и `clean` класса `ViewMap`.

Вторая глава оканчивается описанием взаимодействия карты с временной шкалой и фильтрами по типам данных, а также демонстрацией того, что класс `ViewMap` обладает возможностью добавления как одной карты с соответствующими ей панелями управления, так и нескольких (рисунок 4).



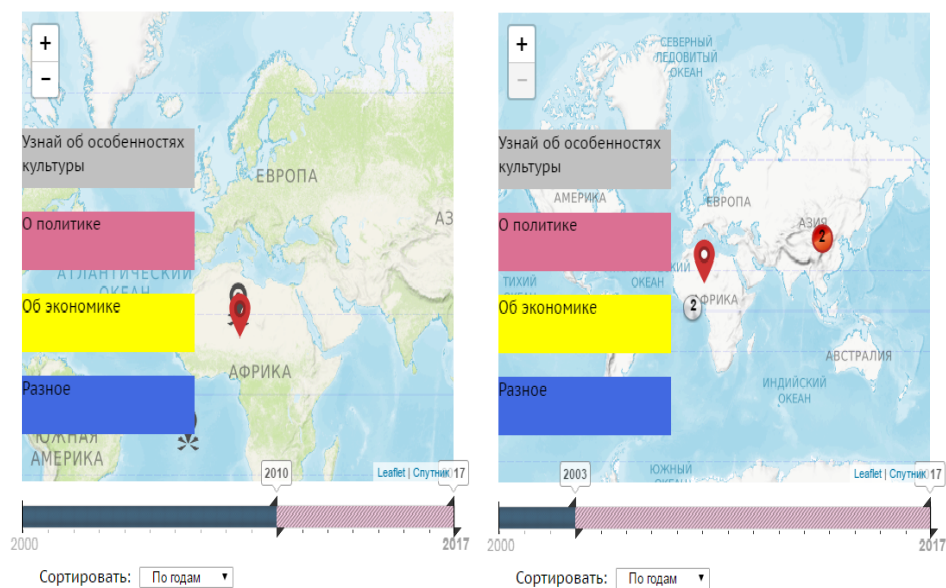


Рисунок 4: Добавление двух карт на веб-страницу

В приложениях выпускной квалификационной работы представлены исходные программные коды для построения графического элемента интерфейса для новостного сервиса, позволяющего интерактивно выводить новости в виде географической информации и осуществлять фильтрацию вывода по историческим данным и по типам данных

## ЗАКЛЮЧЕНИЕ

В ходе данной работы были подробно изучены основы web-картографии, рассмотрены наиболее распространённые web-картографические сервисы, описаны их возможности и проведён сравнительный обзор, в ходе которого для дальнейших задач была выбрана JavaScript библиотека Leaflet.

На основе Leaflet на языке JavaScript был создан класс ViewMap, позволяющий инициализировать на веб-странице графический элемент интерфейса для новостного сервиса, позволяющий интерактивно выводить новости в виде географической информации и осуществлять фильтрацию вывода по историческим данным и по типам данных.

Данный класс имеет следующие возможности:

1. Добавление произвольного количества карт на веб-страницу.
2. Использование любого тайлового или кэшируемого сервиса.
3. Задание любых координат, масштабов и позиционирования элементов управления.
4. Определение геопозиции пользователя.
5. Динамическая подгрузка новой географической информации и её удаление.
6. Добавление и удаление пользователями собственных маркеров.
7. Кластеризация маркеров.
8. Привязка к карте временной шкалы.
9. Добавление фильтров, осуществляющих сортировку по типам данных.